

Ατομική Διπλωματική Εργασία

**ΣΥΓΚΡΙΣΗ ΕΡΓΑΛΕΙΩΝ ΕΞΑΓΩΓΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΑΔΕΙΩΝ ΣΕ ΛΟΓΙΣΜΙΚΟ ΑΝΟΙΚΤΟΥ ΚΩΔΙΚΑ**

Ηλέκτρα Γεωργίου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2013

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

**ΣΥΓΚΡΙΣΗ ΕΡΓΑΛΕΙΩΝ ΕΞΑΓΩΓΗΣ ΠΛΗΡΟΦΟΡΙΑΣ ΑΔΕΙΩΝ ΣΕ ΛΟΓΙΣΜΙΚΟ
ΑΝΟΙΚΤΟΥ ΚΩΔΙΚΑ**

Ηλέκτρα Γεωργίου

Επιβλέπων Καθηγητής
Γεωργία Καπιτσάκη

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων
απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου
Κύπρου

Μάιος 2013

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπων καθηγήτρια μου Κυρία Γεωργία Καπιτσάκη για την πολύτιμη βοήθεια και στήριξη της. Επίσης θα ήθελα να ευχαριστήσω τον Κύριο Armiijn Hemel δημιουργό του εργαλείου BAT για τη σημαντική βοήθεια του σε μερικά προβλήματα που αντιμετώπισα. Τέλος ευχαριστώ όλους τους φίλους μου και την οικογένεια μου για την στήριξη τους σε όλα τα χρόνια των σπουδών μου.

Περίληψη

Η συμβατότητα μεταξύ αδειών σε λογισμικό ανοικτού κώδικα έχει πάρει μεγάλες διαστάσεις στη σημερινή εποχή αφού τα τελευταία χρόνια παρατηρείται μεγάλη ανάπτυξη στο λογισμικό αυτό. Στη διπλωματική αυτή έγινε μελέτη και σύγκριση εργαλείων που μπορούν να εξάγουν την πληροφορία άδειας ή και αδειών από λογισμικά ανοικτού κώδικα όπως βιβλιοθήκες και πηγαίο κώδικα. Συγκεκριμένα τα εργαλεία που μελετήθηκαν είναι το εργαλείο Fossology, Ninka, Ohcount, OSLC, ASLA, BAT και τέλος το What-License? Τα εργαλεία μετά από τα πειράματα στο σύνολο δεδομένων αξιολογήθηκαν βάσει κάποιων μετρικών.

Επίσης στα πλαίσια της παρούσας διπλωματικής υλοποιήθηκαν δύο προγράμματα. Το πρώτο πρόγραμμα Converter παίρνει σαν είσοδο αρχεία εξόδου από τα εργαλεία Fossology, Ninka και Ohcount και παράγει ως έξοδο ένα αρχείο που περιέχει τις πληροφορίες των αρχείων εισόδου σε μια γενική όμως μορφή. Το δεύτερο και κυρίως πρόγραμμα που υλοποιήθηκε είναι το Compatibility Checker το οποίο παίρνει σαν είσοδο το αρχείο εξόδου του προγράμματος Converter και ελέγχει για τυχόν παραβιάσεις στη συμβατότητα μεταξύ των αδειών που περιέχονται στο αρχείο εισόδου.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Πρόλογος	1
	1.2 Στόχος Διπλωματικής Εργασίας	2
	1.3 Μεθοδολογία που ακολουθήθηκε	3
	1.4 Δομή Διπλωματικής Εργασίας	3
Κεφάλαιο 2	Λογισμικό Ανοικτού Κώδικα.....	5
	2.1 Ορισμός λογισμικού ανοικτού κώδικα	5
	2.2 Πλεονεκτήματα χρήσης λογισμικού ανοικτού κώδικα	6
	2.3 Άδειες σε λογισμικό ανοικτού κώδικα	6
	2.4 Συμβατότητα μεταξύ αδειών	8
Κεφάλαιο 3	Περιγραφή Εργαλείων.....	10
	3.1 Εισαγωγή	10
	3.2 Fossology	11
	3.3 Ninka	12
	3.4 Ohcount	12
	3.5 OSLC	13
	3.6 ASLA	13
	3.7 BAT	14
	3.8 What-License?	15
Κεφάλαιο 4	Πειραματική Αξιολόγηση.....	16
	4.1 Εισαγωγή	16
	4.2 Επιλογή των εργαλείων	16
	4.3 Επιλογή δεδομένων εισόδου	17
	4.4 Επιλογή των μετρικών	23
	4.5 Αποτελέσματα πειραμάτων	26
	4.6 Σύγκριση	31

Κεφάλαιο 5	Υλοποίηση Εργαλείων	39
5.1	Εισαγωγή	39
5.2	Εργαλείο Μετατροπίας (Converter)	40
5.2.1	Κίνητρο για υλοποίηση	40
5.2.2	Περιγραφή Εργαλείου	45
5.2.3	Υλοποίηση Εργαλείου	49
5.3	Εργαλείο Ελεγκτής συμβατότητας (Compatibility Check)	50
5.3.1	Κίνητρο για υλοποίηση	50
5.3.2	Περιγραφή Εργαλείου	50
5.3.3	Υλοποίηση Εργαλείου	53
ΚΕΦΑΛΑΙΟ 6	Συμπεράσματα	54
6.1	Γενικά Συμπεράσματα	54
6.2	Μελλοντική Εργασία	56
Βιβλιογραφία		57
Παράρτημα Α		Α
	Πίνακας με στοιχεία για το σύνολο των δεδομένων εισόδου.....	A-1
	Πίνακας με τις άδειες του εργαλείου Fossology.....	A-2
	Πίνακας με όλες τις μετρικές που χρησιμοποιήθηκαν στη σύγκριση.....	A-3
	Πίνακας με τις υποστηριζόμενες άδειες του εργαλείου Converter.....	A-4
Παράρτημα Β		Β
	Κώδικας εργαλείου Μεταροπίας (Converter).....	B-1
	Κώδικας εργαλείου Ελεγκτής Συμβατότητας (Compatibility Check).....	B-2
Παράρτημα Γ		Γ
	Αρχείο readme εργαλείου Μεταροπίας (Converter).....	Γ-1
	Αρχείο readme εργαλείου Ελεγκτής Συμβατότητας (Compatibility Check).....	Γ-2

Κεφάλαιο 1

Εισαγωγή

1.1 Πρόλογος	1
1.2 Στόχος Διπλωματικής Εργασίας	2
1.3 Μεθοδολογία που ακολουθήθηκε	3
1.4 Δομή Διπλωματικής Εργασίας	3

1.1 Πρόλογος

Τον όρο λογισμικό ανοικτού κώδικα (open source software) τον ακούμε πολύ συχνά τα τελευταία χρόνια και με μια γρήγορη έρευνα στο διαδίκτυο μπορούμε να δούμε ότι υπάρχουν πάρα πολλά λογισμικά ανοικτού κώδικα διαθέσιμα προς το κοινό. Κάθε λογισμικό περιλαμβάνει μια τουλάχιστον άδεια λογισμικού ανοικτού κώδικα. Μια άδεια περιγράφει τα προνόμια που έχει ο χρήστης, αλλά και τις υποχρεώσεις του. Υπάρχουν διάφορες άδειες που μπορούν να χρησιμοποιηθούν για την αδειοδότηση του κάθε λογισμικού, μερικές πιο περιοριστικές και κάποιες άλλες λιγότερο. Η κάθε εταιρεία ή ο κάθε δημιουργός του λογισμικού ανάλογα με τα προνόμια που θέλει να δώσει στους χρήστες διαλέγει την κατάλληλη άδεια για να καλύπτει το λογισμικό του.

Το κύριο πρόβλημα με τις άδειες αυτές είναι ότι μπορεί να υπάρξουν παραβιάσεις μεταξύ τους αν περιέχονται για παράδειγμα δύο ή περισσότερες μη συμβατές άδειες σε ένα λογισμικό ή αν χρησιμοποιήσουμε λογισμικό ανοικτού κώδικα όπως βιβλιοθήκες ή άλλα μικρότερα συστήματα τρίτων που μπορεί να περιέχουν άδειες που να μην είναι συμβατές με τις υπόλοιπες άδειες στο τελικό μας σύστημα. Είναι σημαντικό λοιπόν να υπάρχουν εργαλεία τα οποία μπορούν να αναγνωρίσουν τις άδειες που περιέχονται μέσα στα διάφορα λογισμικά ανοικτού κώδικα είτε από βιβλιοθήκες, πηγαίο κώδικα και άλλα συστατικά διαφόρων παροχών λογισμικού, έτσι ώστε να μπορεί να βλέπει ο χρήστης του λογισμικού αν υπάρχουν άδειες οι οποίες δεν είναι συμβατές.

Η διαδικασία ανίχνευσης πιθανών παραβιάσεων μεταξύ των αναγνωρισμένων από τα εργαλεία αδειών μπορεί να γίνει όμως επίπονη διαδικασία αν υπάρχουν πολλές διαφορετικές άδειες μέσα σε ένα λογισμικό. Για το λόγο αυτό μερικά από τα εργαλεία που μελετήθηκαν στα πλαίσια της διπλωματικής προσφέρουν τη δυνατότητα ανίχνευσης παραβιάσεων της συμβατότητας μεταξύ των αδειών που αναγνωρίζουν ότι περιέχονται μέσα σε ένα λογισμικό, κάτι που είναι πολύ σημαντικό να γνωρίζει ο χρήστης του λογισμικού.

1.2 Στόχος Διπλωματικής Εργασίας

Σκοπός της παρούσας διπλωματικής αποτελεί η μελέτη, υλοποίηση, σύγκριση και πειραματική αξιολόγηση εργαλείων που εξάγουν την πληροφορία άδειας ή αδειών από συστήματα λογισμικού ανοικτού κώδικα. Τα εργαλεία αυτά εξάγουν την πληροφορία από βιβλιοθήκες, πηγαίο κώδικα και άλλα συστατικά διαφόρων παροχών λογισμικού. Το κάθε εργαλείο χρησιμοποιεί διαφορετικές τεχνικές για να πάρει την πληροφορία της πιθανής άδειας και στη συνέχεια να την ταυτοποιήσει με την κατάλληλη άδεια από τις υπάρχουσες. Επίσης οι δυνατότητες που προσφέρει κάθε εργαλείο είναι διαφορετικές, κάποια εργαλεία δίνουν στο χρήστη περισσότερες επιλογές και δυνατότητες από κάποια άλλα.

Επίσης στα πλαίσια της παρούσας διπλωματικής υλοποιήθηκαν δύο προγράμματα για την ανίχνευση παραβιάσεων στη συμβατότητα μεταξύ των αδειών. Το πρώτο πρόγραμμα Converter μετατρέπει αρχεία εξόδου συγκεκριμένων εργαλείων σε μια γενική μορφή έτσι ώστε να μπορούν να χρησιμοποιηθούν ως αρχεία εισόδου στο δεύτερο πρόγραμμα, το Compatibility Check που ανιχνεύει τυχόν παραβιάσεις μεταξύ αδειών. Τα δύο αυτά προγράμματα υλοποιήθηκαν με αφορμή του ότι κάποια εργαλεία δεν παρέχουν δυνατότητα ελέγχου της συμβατότητας μεταξύ των αδειών που αναγνωρίζουν. Έτσι για τρία από αυτά τα εργαλεία παίρνοντας τα αρχεία εξόδου τους, που περιέχουν τις άδειες που περιέχονται μέσα σε κάθε αρχείο του λογισμικού, και μετατρέποντας τα σε μια γενική μορφή με το πρόγραμμα Converter, γίνεται έλεγχος συμβατότητας στη συνέχεια με το πρόγραμμα Compatibility Check που υλοποιήθηκε.

1.3 Μεθοδολογία που ακολουθήθηκε

Για να επιτευχθούν οι στόχοι της Διπλωματικής μου Εργασίας ακολουθήθηκε η εξής μεθοδολογία:

Αρχικά μελετήθηκαν οι όροι λογισμικό ανοικτού κώδικα και άδειες σε λογισμικά ανοικτού κώδικα και στη συνέχεια διάφορα εργαλεία για αναγνώριση αδειών, μέσα από άρθρα [12-16] που προτάθηκαν από την επιβλέπουσα, Κυρία Γεωργία Καπιτσάκη. Στη συνέχεια επιλέχθηκαν από τα άρθρα που μελετήθηκαν και από έρευνα στο διαδίκτυο, τα εργαλεία για τα οποία θα γινόταν η σύγκριση. Για την εγκατάσταση των εργαλείων χρειάστηκε να ενσωματωθούν αρκετές βιβλιοθήκες κυρίως στο λειτουργικό σύστημα Ubuntu. Μετά την εγκατάσταση των εργαλείων πραγματοποιήθηκαν μικρές δοκιμές με πολύ μικρού μεγέθους αρχεία για να διαπιστωθεί η σωστή λειτουργία τους. Αφού ολοκληρώθηκαν οι μικρές δοκιμές επιλέχθηκαν τα λογισμικά ανοικτού κώδικα που θα έπαιρναν σαν είσοδο τα εργαλεία για τη φάση της πειραματικής αξιολόγησης. Τα λογισμικά που επιλέχθηκαν προέρχονται από αποθετήρια ανοικτού κώδικα όπως Maven και SourceForge. Για την πειραματική αξιολόγηση αφού επιλέχθηκαν τα δεδομένα εισόδου των εργαλείων στην συνέχεια καθορίστηκαν οι μετρικές στις οποίες θα αξιολογούνταν τα εργαλεία. Τέλος έγινε η πειραματική αξιολόγηση πάνω στα δεδομένα εισόδου και στις μετρικές και εξαχθήκαν συμπεράσματα από τα αποτελέσματα της σύγκρισης των εργαλείων.

1.4 Δομή Διπλωματικής Εργασίας

Η Διπλωματική Εργασία χωρίζεται σε 6 κεφάλαια. Στο επόμενο κεφάλαιο, Κεφάλαιο 2, αναφέρεται ο ορισμός του λογισμικού ανοικτού κώδικα και γίνεται μια περιγραφή για τη χρήση των διάφορων αδειών σε αυτό το λογισμικό καθώς επίσης και για η συμβατότητα μεταξύ των διάφορων αδειών. Στο Κεφάλαιο 3 περιγράφονται γενικά τα εργαλεία που μελετήθηκαν και αξιολογήθηκαν μέσα από την πειραματική αξιολόγηση. Στο Κεφάλαιο 4 περιγράφεται η πειραματική αξιολόγηση των εργαλείων. Συγκεκριμένα στο κεφάλαιο αυτό περιγράφεται με συντομία η πειραματική αξιολόγηση που διενεργήθηκε, αναφέρονται τα εργαλεία που επιλέχθηκαν για τη σύγκριση καθώς επίσης και τα κριτήρια των επιλογών αυτών, τα δεδομένα εισόδου που επιλέχθηκαν για την πειραματική αξιολόγηση καθώς επίσης και οι λόγοι που οδήγησαν στην επιλογή τους. Επίσης στο κεφάλαιο αυτό περιλαμβάνονται οι μετρικές για την αξιολόγηση, τα αποτελέσματα που εξάγαμε από τα πειράματα και τέλος περιγράφονται τα συμπεράσματα από τα πειράματα και γίνεται η σύγκριση μεταξύ των εργαλείων. Στο επόμενο κεφάλαιο, το Κεφάλαιο 5, περιγράφονται τα δύο εργαλεία που

υλοποιήθηκαν, το Converter και το Compatibility Check. Συγκεκριμένα στην αρχή του κεφαλαίου υπάρχει μια εισαγωγή για την υλοποίηση των δύο εργαλείων και στη συνέχεια σε δύο υποκεφάλαια παρουσιάζεται το κίνητρο για την υλοποίηση και μια περιγραφή για το κάθε εργαλείο. Στο 6^ο και τελευταίο κεφάλαιο παρουσιάζονται τα συμπεράσματα που προέκυψαν μέσα από τη παρούσα Διπλωματική Εργασία και στη συνέχεια περιγράφεται μελλοντική εργασία.

Κεφάλαιο 2

Λογισμικό ανοικτού κώδικα

2.1 Ορισμός λογισμικού ανοικτού κώδικα	5
2.2 Πλεονεκτήματα χρήσης λογισμικού ανοικτού κώδικα	6
2.3 Άδειες σε λογισμικό ανοικτού κώδικα	6
2.4 Συμβατότητα μεταξύ αδειών	8

2.1 Ορισμός λογισμικού ανοικτού κώδικα

Το λογισμικό ανοικτού κώδικα (open source software) είναι κάτι διαφορετικό από ότι είχε συνηθίσει και γνώριζε ο περισσότερος κόσμος αφού μέχρι πριν μερικές δεκαετίες η χρήση κάποιου λογισμικού παρέπεμπε αυτόματα στη σκέψη ότι υπάρχει κάποιο κόστος για τη χρήση του. Τα τελευταία χρόνια έχει εξαπλωθεί πάρα πολύ η δημιουργία και χρήση λογισμικών ανοικτού κώδικα. Όλο και περισσότερες εταιρείες και προγραμματιστές δημιουργούν λογισμικά τα οποία είναι διαθέσιμα προς το κοινό χωρίς κάποιο κόστος και τα οποία οι διάφοροι χρήστες τους μπορούν να τα επεξεργαστούν, να τα διανέμουν ή να κάνουν απλή χρήση τους.

Ο όρος λογισμικό ανοικτού κώδικα (open source software) χρησιμοποιείται για να περιγράψει το λογισμικό του οποίου ο πηγαίος κώδικας του είναι ανοικτός προς το κοινό. Έχοντας έτσι κάποιος τον πηγαίο κώδικα κάποιου λογισμικού έχει τη δυνατότητα να τον εξετάσει, να τον τροποποιήσει και να τον βελτιώσει αν επιθυμεί ή απλά να το χρησιμοποιήσει μόνο του ή και ως μέρος στην ανάπτυξη ενός άλλου λογισμικού συστήματος [1]. Το ότι τα λογισμικά αυτά είναι ανοικτά προς το κοινό χωρίς κάποιο κόστος ίσως να δημιουργεί την εντύπωση ότι είναι μικρού μεγέθους λογισμικά και όχι τόσο πολύπλοκα. Η εντύπωση αυτή είναι λανθασμένη και αυτό μπορεί κάποιος να το διαπιστώσει βλέποντας κάποια παραδείγματα αυτών των λογισμικών όπως ο Apache HTTP Server ο οποίος είναι ένας ολοκληρωμένος διακομιστής διαδικτύου (web server) [2]. Επίσης άλλο παράδειγμα λογισμικού ανοικτού κώδικα είναι το λειτουργικό σύστημα Linux το οποίο μπορεί κάποιος

να καταλάβει την πολυπλοκότητα ενός τέτοιου λογισμικού αφού μιλάμε για ολόκληρο λειτουργικό σύστημα [3]. Λογισμικά ανοικτού κώδικα υπάρχουν πάρα πολλά και καλύπτουν αρκετές αν όχι όλες τις ανάγκες των χρηστών. Τα λογισμικά αυτά παρέχονται στο κοινό μέσα από αποθετήρια (repositories) λογισμικών ανοικτού κώδικα (open source software repositories) όπως είναι το Google Code [4], Maven [5] και SourceForge [6].

2.2 Πλεονεκτήματα χρήσης λογισμικού ανοικτού κώδικα

Η χρήση λογισμικού ανοικτού κώδικα έχει εξαπλωθεί ραγδαία τα τελευταία χρόνια, καθώς προσφέρει σημαντικά πλεονεκτήματα έναντι του ιδιόκτητου/προστατευμένου λογισμικού (proprietary software). Το σημαντικότερο ίσως πλεονέκτημα για πολλούς χρήστες είναι ότι το λογισμικό ανοικτού κώδικα μπορεί να χρησιμοποιηθεί χωρίς κάποιο κόστος, έτσι δίνει τη δυνατότητα στον οποιονδήποτε χρειάζεται κάποιο συγκεκριμένο λογισμικό να το χρησιμοποιήσει χωρίς να χρειάζεται να ανησυχεί για το αν μπορεί να αντεπεξέλθει οικονομικά [7]. Επίσης από τη στιγμή που ο πηγαίος κώδικας κάποιου λογισμικού είναι διαθέσιμος σε κάποιο χρήστη αυτόματα σημαίνει ότι είναι διαθέσιμη και η γνώση του συγκεκριμένου λογισμικού, άρα με τη χρήση ενός λογισμικού ανοικτού κώδικα μπορεί κάποιος να μάθει και πώς λειτουργεί το λογισμικό αυτό [7,8]. Έχοντας λοιπόν κάποιος τη γνώση για ένα λογισμικό μπορεί επίσης να το τροποποιήσει και ακόμη να το βελτιώσει αν το επιθυμεί εφόσον κάτι τέτοιο επιτρέπεται από το δημιουργό του λογισμικού, όπως εξηγείται στην επόμενη ενότητα [7]. Τέλος η δυνατότητα αυτή της βελτίωσης ενός λογισμικού που έχουν οι χρήστες είναι πάρα πολύ σημαντική αφού ξέροντας πολλοί πως δουλεύει τότε μπορούν και πολλοί να διορθώσουν κάποιο λάθος που ανιχνεύθηκε στο λογισμικό πολύ πιο γρήγορα από το να διορθωθεί ένα λάθος σε ένα ιδιόκτητο λογισμικό που οι μόνοι που μπορούν να το διορθώσουν είναι οι προγραμματιστές του λογισμικού αυτού [8].

2.3 Άδειες σε λογισμικό ανοικτού κώδικα

Ο πηγαίος κώδικας ενός λογισμικού ανοικτού κώδικα είναι διαθέσιμος προς όλους χωρίς κάποιο κόστος. Εντούτοις ο δημιουργός κάθε λογισμικού ανοικτού κώδικα διαλέγει την κατάλληλη άδεια (license) ή ακόμη και άδειες για να καλύπτει το λογισμικό του. Οι άδειες αυτές, οι οποίες περιλαμβάνονται στον πηγαίο κώδικα συνήθως ή στην περιγραφή του λογισμικού, αναφέρουν στους χρήστες τα προνόμια αλλά και τις υποχρεώσεις τους ως προς το λογισμικό αυτό. Κάθε δημιουργός μπορεί να επιλέξει την άδεια που θέλει να καλύπτει το λογισμικό του ανάλογα με τα δικαιώματα που θέλει να έχει ο κάθε χρήστης του λογισμικού του όπως επίσης και τις υποχρεώσεις που θα πρέπει να τηρήσει κατά τη χρήση του.

Οι διάφορες άδειες που υπάρχουν για τα λογισμικά ανοικτού κώδικα χωρίζονται σε κατηγορίες ανάλογα με τα δικαιώματα που παρέχουν και τους περιορισμούς που επιβάλλουν. Οι κύριες κατηγορίες των αδειών είναι οι εξής [9]:

- Επιτρεπτικές Άδειες (Permissive Licenses): Επιτρέπουν στους δικαιούχους να αντιγράψουν, να διανέμουν και να τροποποιήσουν τον πηγαίο κώδικα του λογισμικού ζητώντας μόνο από τους δικαιούχους (licensees) του λογισμικού αυτού να αναγνωρίσουν ότι έχει χρησιμοποιηθεί λογισμικό ανοικτού κώδικα δίνοντας αναγνώριση στους προγραμματιστές. Επίσης οι επιτρεπτικές άδειες δεν επιβάλλουν περιορισμούς ως προς την αδειοδότηση και εκμετάλλευση οποιουδήποτε ιδιόκτητου λογισμικού που δημιουργήθηκε από τους δικαιούχους, έτσι ώστε να μπορούν να ενσωματώσουν τον ίδιο τον πηγαίο κώδικα, ή τροποποιήσεις του, σε ένα νέο προϊόν που δεν θα είναι ανοικτού κώδικα. Μερικά παραδείγματα τέτοιων αδειών είναι η BSD License και η MIT License.
- Μόνιμες Άδειες (Persistent Licenses): Επιτρέπουν στους δικαιούχους να αντιγράψουν και να διανείμουν τον πηγαίο κώδικα, εάν όμως είναι μέρος ολόκληρης ο κώδικας σε μια νέα εργασία, χωρίς να απαιτείται όμως η νέα εργασία να λάβει κάποια άδεια στο σύνολό της. Αν στον πηγαίο κώδικα δεν γίνουν κάποιες αλλαγές και μεταγλωττίζεται ή συνδέεται με κάποια νέα εργασία, τότε η νέα εργασία μπορεί να λάβει μια άδεια που να καλύπτει ένα νέο ιδιόκτητο λογισμικό. Όταν όμως ο πηγαίος κώδικας αλλάξει τότε η νέα εργασία θα πρέπει να λάβει μια άδεια, η οποία θα είναι η ίδια με αυτήν του πηγαίου κώδικα ή αυστηρότερη. Το πιο γνωστό παράδειγμα τέτοιου τύπου άδειας είναι η GNU Lesser General Public License (LGPL).
- Μόνιμες και Κληρονομήσιμες Άδειες (Persistent and Inheritable Licenses): Επιτρέπουν στους δικαιούχους να αντιγράψουν, να διανέμουν και να τροποποιήσουν τον πηγαίο κώδικα του λογισμικού, αλλά όλες οι εργασίες οι οποίες θα δημιουργηθούν από τους δικαιούχους και θα περιλαμβάνουν τον πηγαίο αυτό κώδικα θα πρέπει να λάβουν άδεια η οποία θα καλύπτει τους όρους που έχει επιβάλει ο δημιουργός του αρχικού πηγαίου κώδικα. Στόχος λοιπόν αυτού του τύπου αδειών είναι να δημιουργήσουν μια αλυσίδα από δικαιούχους οι οποίοι όλοι θα είναι νομικά καλυμμένοι από τη χρήση του πηγαίου κώδικα των υπολοίπων. Η πιο γνωστή άδεια αυτού του τύπου είναι η GNU General Public License (GPL).

Στον πιο κάτω πίνακα 2.1 παρουσιάζονται παραδείγματα αδειών για κάθε μια από τις κατηγορίες που αναφέρθηκαν πιο πάνω [9].

<u>Επιτρεπτικές Άδειες (Permissive Licenses)</u>	<u>Μόνιμες Άδειες (Persistent Licenses)</u>	<u>Μόνιμες και Κληρονομήσιμες Άδειες (Persistent and Inheritable Licenses)</u>
MIT License	GNU Lesser General Public License (LGPL)	GNU General Public License (GPL)
BSD License	Eclipse Public License (Eclipse)	European Union Public License (EURL)
Apache License	Mozilla Public License (MPL)	Sleepycat License
Zlib	Artistic License	AGPL (Affero GPL)
PHP	GFDL	Creative Commons Attribution- ShareAlike

Σχήμα 2.1 Πίνακας με παραδείγματα αδειών για κάθε κατηγορία

2.4 Συμβατότητα μεταξύ αδειών

Ο δημιουργός ενός έργου λογισμικού ανοικτού κώδικα θα πρέπει να αποφασίσει αρχικά τόσο τα δικαιώματα όσο και τις υποχρεώσεις που θα έχουν οι χρήστες του έργου του και έπειτα βάσει αυτών να καταλήξει στην επιλογή των αδειών που θέλει να δώσει στο έργο. Για κάθε δημιουργό υπάρχουν πολλές επιλογές όσον αφορά τις άδειες που επιθυμεί να καλύπτουν το έργο λογισμικού του. Κάθε άδεια περιλαμβάνει διαφορετικούς συνδυασμούς δικαιωμάτων και υποχρεώσεων για τους χρήστες του λογισμικού που καλύπτει, δίνοντας έτσι στους δημιουργούς την ευκαιρία να επιλέξουν την άδεια που πραγματικά θέλουν να καλύπτει το έργο χωρίς να πρέπει να συμβιβάζονται στην επιλογή τους. Ακόμη ένας δημιουργός μπορεί να επιλέξει περισσότερες από μία άδειες να καλύπτουν το έργο λογισμικού του λόγω του ότι

μια άδεια μόνο μπορεί να μη προσφέρει όλα τα δικαιώματα και τις υποχρεώσεις που επιθυμεί να έχουν οι χρήστες του έργου του. Έτσι λοιπόν σε ένα έργο λογισμικού ανοικτού κώδικα (open source software project) είναι πιθανό να περιέχονται περισσότερα από μια άδειες. Συνεπώς κάθε χρήστης του έργου αυτού έχει δικαιώματα και υποχρεώσεις που αντιστοιχούν σε πολλές άδειες, με αποτέλεσμα αν ο δημιουργός του έργου δεν είναι προσεκτικός να υπάρξει ασυμβατότητα μεταξύ των αδειών αυτών.

Για να είναι δύο ή περισσότερες άδειες συμβατές μεταξύ τους θα πρέπει τα δικαιώματα που δίνει η μια στο χρήστη να μην είναι απαγορευτικά από την άλλη άδεια. Δηλαδή δεν θα πρέπει η μια να επιτρέπει ενέργειες στο χρήστη, που στην άλλη άδεια να απαγορεύονται. Ένα παράδειγμα δύο τέτοιων αδειών είναι άδειες GPLv2 και GPLv3. Οι δύο αυτές άδειες δεν είναι συμβατές μεταξύ τους γιατί για παράδειγμα η άδεια GPLv3 έχει απαιτήσεις που δεν υπάρχουν στην GPLv2, όπως για παράδειγμα η παροχή πληροφοριών που αφορούν την εγκατάσταση ενός έργου [10]. Η άδεια GPLv3 δηλαδή, απαιτεί κάθε έργο λογισμικού το οποίο καλύπτει να παρέχει πληροφορίες και οδηγίες σχετικά με την εγκατάσταση του έργου αυτού από οποιονδήποτε χρήστη. Αντίθετα η άδεια GPLv2 δεν έχει αυτή την απαίτηση, άρα τα λογισμικά που καλύπτονται από την άδεια αυτή δεν είναι απαραίτητο να περιέχουν πληροφορίες που να αφορούν την εγκατάσταση του έργου λογισμικού. Από το παράδειγμα αυτό λοιπόν μπορούμε να δούμε ότι οι δύο άδειες αυτές δεν είναι συμβατές, αφού έχουν διαφορετικές απαιτήσεις, και αν συνδυάζονταν σε ένα έργο λογισμικού δεν θα υπήρχαν ξεκάθαρες απαιτήσεις αφού η μία άδεια αναιρεί κάποιες από τις απαιτήσεις της άλλης.

Αν οι άδειες που περιέχονται σε ένα έργο λογισμικού ανοικτού κώδικα δεν είναι συμβατές τότε ο πηγαίος κώδικας (source code) του έργου αυτού δεν μπορεί να χρησιμοποιηθεί για τη δημιουργία ενός νέου έργου λόγω των μη ξεκάθαρων απαιτήσεων και δικαιωμάτων του προς τους χρήστες. Δεν θα ήταν δυνατό δηλαδή να συνδυαστεί ο πηγαίος κώδικας ενός έργου με κώδικες άλλων έργων, κάτι που σημαίνει ότι αν ένα έργο λογισμικού περιέχει μη συμβατές άδειες ακυρώνεται αυτόματα και η χρήση του από τρίτους. Με την δυνατότητα χρήσης του πηγαίου κώδικα ενός έργου να μην είναι εφικτή, συνεπάγεται ότι το έργο αυτό δε βοηθά πλέον στην ανάπτυξη νέων έργων αλλά ούτε και στην επίλυση των προβλημάτων για τα οποία αρχικά υλοποιήθηκε. Έτσι ακυρώνεται ουσιαστικά η χρησιμότητα του έργου αυτού, κάτι που κανένας δημιουργός δεν θα ήθελε να συμβεί. Είναι λοιπόν σημαντικό για κάθε έργο λογισμικού ανοικτού κώδικα να λαμβάνεται υπόψη η συμβατότητα μεταξύ των υποψήφιων αδειών, έτσι ώστε να αποφεύγονται οι παραβιάσεις συμβατότητας μεταξύ τους και κατά συνέπεια να μπορεί ο πηγαίος κώδικας του έργου να χρησιμοποιηθεί χωρίς οποιαδήποτε προβλήματα.

Κεφάλαιο 3

Περιγραφή Εργαλείων

3.1 Εισαγωγή	10
3.2 Fossology	11
3.3 Ninka	12
3.4 Ohcount	12
3.5 OSLC	13
3.6 ASLA	13
3.7 BAT	14
3.8 What-License?	15

3.1 Εισαγωγή

Το κεφάλαιο αυτό περιγράφει με συντομία τα εργαλεία που μελετήθηκαν. Τα εργαλεία αυτά επιλέχθηκαν μετά από μελέτη σχετικών άρθρων της περιοχής των αδειών λογισμικού ανοικτού κώδικα [12-16] και έπειτα από έρευνα για επιπλέον εργαλεία που δεν αναφέρονταν στα άρθρα. Αφού επιλέχθηκαν κάποια από τα εργαλεία αυτά, έγινε η εγκατάστασή τους στα κατάλληλα λειτουργικά συστήματα, Linux και Windows, και πραγματοποιήθηκαν πειράματα για τη χρήση τους. Όλα τα εργαλεία που μελετήθηκαν από τα άρθρα συμπεριλήφθηκαν στην πειραματική αξιολόγηση εκτός από τα εργαλεία LIDESC και OSS Discovery. Το εργαλείο LIDESC δεν μπορούσε να συμπεριληφθεί στην πειραματική αξιολόγηση λόγω προβλημάτων που υπήρχαν στον κώδικα του εργαλείου που δεν καθιστούσαν δυνατή την εκτέλεσή του. Επίσης το εργαλείο OSS Discovery δεν προσφέρεται δωρεάν στο κοινό οπότε δεν μπορούσε να χρησιμοποιηθεί για να συμπεριληφθεί στα πειράματα.

Τα εργαλεία που μελετήθηκαν και συμπεριλήφθηκαν στη πειραματική αξιολόγηση και σύγκριση παρουσιάζονται συγκεντρωμένα στον πίνακα 3.1 μαζί με το λειτουργικό σύστημα στο οποίο μπορούν να χρησιμοποιηθούν και τη γλώσσα προγραμματισμού στην οποία υλοποιήθηκε. Στις επόμενες ενότητες για το κάθε εργαλείο περιγράφεται ο τρόπος με τον οποίο λειτουργεί, οι δυνατότητες που παρέχει στο χρήστη καθώς και λεπτομέρειες για την υλοποίησή του.

Εργαλεία	Λειτουργικό Σύστημα	Γλώσσα Προγραμματισμού
Fossology	Linux	C, PHP
Ninka	Linux	Perl
Ohcount	Linux, Mac	C
OSLC	Οποιοδήποτε	Java
ASLA	Linux	Java
BAT	Linux	Python
What-License?	Online	Html, Javascript

Σχήμα 3.1 Πίνακας εργαλείων που συμπεριλήφθηκαν στη πειραματική αξιολόγηση

3.2 Fossology

Το εργαλείο αυτό αναγνωρίζει άδειες χρησιμοποιώντας μια τεχνική η οποία καθορίζει κενά διαστήματα και λέξεις οι οποίες δεν έχουν δηλωθεί, στη μέση των προτύπων αναζήτησης και αναφέρει τα αποτελέσματα ταυτοποίησης σε ποσοστά [15]. Για να μπορέσει να αναγνωρίσει πιθανές άδειες που μπορεί να υπάρχουν, το εργαλείο αυτό ελέγχει όλα τα αρχεία, με όλους τους τύπους, που περιλαμβάνονται μέσα στο project αυτό.

Το εργαλείο Fossology παρέχει τη δυνατότητα ομαδοποίησης αδειών, έτσι ώστε η αναγνώριση κάποιων ομαδοποιημένων αδειών να μπορεί να γίνει εύκολα. Επίσης τα αποτελέσματα από την αναγνώριση αδειών αποθηκεύονται για μελλοντική χρήση και ο χρήστης έχει τη δυνατότητα να δει τα αποτελέσματα αυτά σε διαφορετικά πακέτα δεδομένων που μπορούν να γίνουν upload είτε στο δικό τους προσωπικό αποθετήριο repository είτε στο δημόσιο αποθετήριο (public repository) που προσφέρει το FOSSology. Η ιδέα της χρήσης του αποθετηρίου (repository) FOSSology είναι να έχουν οι χρήστες τη δυνατότητα να χρησιμοποιήσουν τα αποτελέσματα που προκύπτουν από τη χρήση του εργαλείου και μελλοντικά και να μπορούν να δουν και να μελετήσουν αποτελέσματα και από άλλους χρήστες μέσω του δημόσιου αποθετηρίου FOSSology (public repository) [17,18].

Το εργαλείο αυτό δεν παρέχει δυνατότητα ελέγχου συμβατότητας και εξαρτήσεων μεταξύ των αδειών που αναγνωρίζονται. Επίσης για γίνει προσθήκη μιας νέας άδειες στις ήδη γνωστές άδειες του εργαλείου θα πρέπει να γίνει ξανά η υλοποίηση του λογισμικού του εργαλείου έτσι ώστε να μπορεί να λαμβάνει υπόψη και τη νέα άδεια και επίσης να αλλάξουμε την ήδη υπάρχουσα βάση δεδομένων που περιέχει τα πρωτότυπα των αδειών [12]. Το λειτουργικό σύστημα στο οποίο μπορεί να λειτουργήσει το Fossology είναι Linux

και η γλώσσα προγραμματισμού στην οποία είναι υλοποιημένο είναι C και PHP για το γραφικό περιβάλλον.

3.3 Ninka

Το εργαλείο Ninka παίρνει οποιεσδήποτε καταστάσεις αδειών από το αρχείο εισόδου και μετά το χωρίζει σε προτάσεις κειμένου και ψάχνει για την κάθε πρόταση κειμένου, κάποιο κείμενο που να αντιστοιχεί σε αυτές. Στη συνέχεια οι προτάσεις που έχουν αντιστοιχηθεί με κάποιο από τα κείμενα που αποθηκευμένα στο εργαλείο, αναλύονται με σκοπό να βρεθεί αν υπάρχει κάποια άδεια [19]. Με πιο απλά λόγια το εργαλείο αυτό παίρνει κάθε πρόταση κειμένου που δημιουργεί από το αρχείο εισόδου και προσπαθεί να βρει αν αντιστοιχεί σε κάποια από τα προτάσεις κειμένου των διάφορων αδειών. Το εργαλείο παίρνει τις πρώτες 800 γραμμές από το αρχείο εισόδου και προσπαθεί μέσα από αυτές να δει αν περιέχεται κάποια άδεια στο αρχείο.

Το εργαλείο αυτό δεν παρέχει δυνατότητα ελέγχου συμβατότητας και εξαρτήσεων μεταξύ των αδειών που αναγνωρίζονται. Επίσης το εργαλείο Ninka χρειάζεται λειτουργικό σύστημα Linux για να μπορέσει να λειτουργήσει και είναι υλοποιημένο σε γλώσσα προγραμματισμού Perl.

3.4 Ohcount

Το εργαλείο Ohcount εκτός από την αναγνώριση αδειών, παρέχει τη δυνατότητα μέτρησης των γραμμών κώδικα και σχόλιων που υπάρχουν μέσα σε ένα source file. Το εργαλείο αυτό αποτελείται από δύο κύρια συστατικά: έναν ανιχνευτή ο οποίος καθορίζει την κύρια γλώσσα προγραμματισμού που χρησιμοποιείται στο κάθε αρχείο πηγαίου κώδικα (source files) των δεδομένων εισόδου και ένα πρόγραμμα ανάλυσης (parser) το οποίο χωρίζει γραμμή-γραμμή τα περιεχόμενα του αρχείου πηγαίου κώδικα [20]. Η αναγνώριση και ταυτοποίηση των αδειών που περιέχονται μέσα σε ένα αρχείο γίνεται με τη σύγκριση του κώδικα από το αρχείο εισόδου και του κειμένου των αδειών που υπάρχουν αποθηκευμένες στο εργαλείο. Η τεχνική αναγνώρισης των αδειών βασίζεται στη χρήση κανονικών εκφράσεων και η ταυτοποίηση ενός κειμένου με το κείμενο κάποιας άδειας γίνεται με τη χρήση ταυτοποίησης συμβολοσειρών [16].

Ως είσοδο το εργαλείο αυτό μπορεί να πάρει ένα μόνο αρχείο ή και ολόκληρο φάκελο, με την προϋπόθεση όμως ότι τα αρχεία που επιθυμούμε να εξετάσουμε είναι αρχεία πηγαίου κώδικα (source files). Οποιοδήποτε αρχείο με διαφορετική κατάληξη δεν εξετάζεται από το Ohcount.

Το εργαλείο δεν παρέχει έλεγχο συμβατότητας και εξαρτήσεων μεταξύ των αδειών. Τα υποστηριζόμενα λειτουργικά συστήματα για το εργαλείο είναι Linux και Mac και η γλώσσα προγραμματισμού στην οποία υλοποιήθηκε είναι η C.

3.5 OSLC

Η αναγνώριση αδειών στο εργαλείο αυτό γίνεται χρησιμοποιώντας μια τεχνική στην οποία αναφέρονται τα αποτελέσματα της ταυτοποίησης σε ποσοστά, δείχνοντας πόσο μοιάζει το κείμενο ενός αρχείου πηγαίου κώδικα (source file) με το κείμενο του προτύπου κάποιας άδειας [16].

Το εργαλείο αυτό προσφέρει ανίχνευση παραβιάσεων και εξαρτήσεων μεταξύ των αδειών μόνο σε επίπεδο κώδικα και βασίζεται στις γραμμές που καθορίζουν ποιες άδειες περιλαμβάνονται μέσα στο αρχείο πηγαίου κώδικα (source file). Η ανίχνευση των παραβιάσεων γίνεται με τη χρήση κανόνων μεταξύ των αδειών που έχει αποθηκευμένους το εργαλείο. Έτσι αν κάποιος κανόνας παραβιαστεί τότε το εργαλείο ανιχνεύει την παραβίαση αυτή και την εμφανίζει στο χρήστη [12,14].

Ως είσοδο το εργαλείο μπορεί να πάρει ολόκληρο φάκελο που περιλαμβάνει source αρχεία. Το εργαλείο OSLC είναι ανεξάρτητο πλατφόρμας οπότε δεν έχει πρόβλημα συμβατότητας με κάποιο λειτουργικό σύστημα. Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την υλοποίηση αυτού του εργαλείου είναι η Java.

3.6 ASLA

Το εργαλείο ASLA ψάχνει μέσα σε όλα τα αρχεία πηγαίου κώδικα (source files) και αναγνωρίζει τυχόν άδειες που υπάρχουν. Η αναγνώριση των αδειών γίνεται χρησιμοποιώντας πρότυπα αδειών τα οποία βρίσκονται σε μορφή κανονικών εκφράσεων [12].

Προσφέρει τη δυνατότητα ανίχνευσης παραβιάσεων και εξαρτήσεων ελέγχοντας τους κανόνες συμβατότητας για κάθε άδεια που περιέχονται μέσα στο εργαλείο. Η ανίχνευση παραβιάσεων και εξαρτήσεων μπορεί να εφαρμοστεί μόνο σε πακέτα που έχουν ήδη

μεταγλωττίζεται. Επίσης για να μπορέσει κάποιος χρήστης να προσθέσει ένα νέο πρότυπο μιας άδειας, θα πρέπει να δημιουργήσει το πρότυπο αυτό και να το αποθηκεύσει στο φάκελο με τις ήδη υπάρχουσες άδειες. Ακόμη ένας δεύτερος τρόπος που μπορεί κάποιος να προσθέσει μια νέα άδεια είναι επιλέγοντας το κείμενο από ένα αρχείο κώδικα, και αποθηκεύοντας το κείμενο αυτό ως ένα νέο πρότυπο άδειας. Το εργαλείο σε αυτή την περίπτωση μετατρέπει το κείμενο σε κανονική έκφραση και το αποθηκεύει για μελλοντική χρήση [12]. Το εργαλείο αυτό χρησιμοποιείται σε Linux λειτουργικό σύστημα και είναι υλοποιημένο σε γλώσσα προγραμματισμού Java.

3.7 BAT

Το εργαλείο BAT αναγνωρίζει κλωνοποίηση κώδικα (code cloning) μέσα σε ένα αρχείο διφυσυλλαβών (binary) ελέγχοντας το αρχείο για στοιχεία ότι ο κώδικας του αρχείου αυτού περιέχει πακέτα λογισμικού από τρίτους. Για να μπορέσει να εξάγει χρήσιμες πληροφορίες ελέγχει τα αρχεία εισόδου για δομές που είναι ήδη γνωστές και μπορεί να τις αναγνωρίσει και μετά εφαρμόζει ανίχνευση για κλωνοποίηση κώδικα στα αρχεία αποτελεσμάτων [13]. Επίσης προσφέρει ανίχνευση bootloaders, άνοιγμα συμπιεσμένων αρχείων και φακέλων (archives), αναζήτηση του κελυφους (kernel) Linux, αναγνώριση δυναμικά συνδεδεμένων βιβλιοθηκών (dynamically linked libraries) και αναφέρει τα αποτελέσματα σε μορφή Extensible Markup Language (XML) [16]. Για το διάβασμα του κώδικα στα binary αρχεία σε firmware μορφή και τη σύγκριση του με τον source κώδικα, το εργαλείο χρησιμοποιεί πίνακα συμβολοσειρών και συμβόλων.

Το εργαλείο αυτό δεν προσφέρει αναγνώριση εξαρτήσεων και παραβιάσεων. Τέλος χρησιμοποιείται μόνο σε λειτουργικό σύστημα Linux και είναι υλοποιημένο σε γλώσσα προγραμματισμού Python.

3.8 What-License?

Το What-License? είναι ένα online εργαλείο το οποίο ανιχνεύει άδειες οι οποίες μπορεί να περιέχονται μέσα σε κάποιο κομμάτι κειμένου. Για να χρησιμοποιήσουμε το εργαλείο αυτό το μόνο που χρειάζεται να κάνουμε είναι να αντιγράψουμε το κομμάτι κειμένου ή κώδικα που θέλουμε να ελέγξουμε για άδειες και να το επικολλήσουμε στο text box του εργαλείου αυτού. Στη συνέχεια αφού επιλέξουμε την εντολή για έλεγχο αδειών, το εργαλείο συγκρίνει το κείμενο που του δώσαμε σαν είσοδο με το κείμενο κάποιων αδειών που υπάρχουν στο opensource.org. Τέλος το εργαλείο εμφανίζει τα αποτελέσματα με τη μορφή μιας λίστας

κατάταξης που περιέχει τις άδειες που είναι οι πιο επικρατέστερες από τη σύγκριση, μαζί με τα κείμενα της κάθε άδειας [21].

Το εργαλείο δεν προσφέρει έλεγχο και ανίχνευση παραβιάσεων ή εξαρτήσεων και επειδή είναι online δεν χρειάζεται κάποιο συγκεκριμένο λειτουργικό σύστημα για τη χρήση του.

Κεφάλαιο 4

Πειραματική Αξιολόγηση

4.1 Εισαγωγή	16
4.2 Επιλογή των εργαλείων	16
4.3 Επιλογή των δεδομένων εισόδου	17
4.4 Επιλογή των μετρικών	23
4.5 Αποτελέσματα πειραμάτων	26
4.6 Σύγκριση	31

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο περιγράφεται η πειραματική αξιολόγηση των εργαλείων ανίχνευσης αδειών σε λογισμικό ανοικτού κώδικα. Συγκεκριμένα στο κεφάλαιο αυτό περιγράφονται αρχικά τα εργαλεία που επιλέχθηκαν για τη πειραματική αξιολόγηση, στη συνέχεια αναφέρεται πως έγινε η επιλογή των δεδομένων εισόδου και αναφέρονται τα δεδομένα αυτά. Μετά αναφέρονται οι μετρικές που επιλέχθηκαν για τη αξιολόγηση και σύγκριση των εργαλείων. Επίσης παραθέτονται στη συνέχεια τα αποτελέσματα που προέκυψαν από τα πειράματα και τέλος γίνεται η σύγκριση βάσει των αποτελεσμάτων των εργαλείων και αναφέρονται τα αποτελέσματα που προέκυψαν από τη διαδικασία των πειραμάτων.

Σκοπός της πειραματικής αξιολόγησης είναι η σύγκριση των εργαλείων και η εξαγωγή σημαντικών συμπερασμάτων σχετικά με λειτουργία του κάθε εργαλείου σε σχέση με τα υπόλοιπα. Η σύγκριση των εργαλείων γίνεται με βάσει τις μετρικές που έχουν καθοριστεί.

4.2 Επιλογή των Εργαλείων

Από τα εργαλεία που αναφέρθηκαν στο Κεφάλαιο 3 έχουν συμπεριληφθεί όλα στη πειραματική αξιολόγηση. Η εύρεση και επιλογή των εργαλείων έγινε με βάσει κάποια άρθρα [12-16] που είχαν ως θέμα το λογισμικού ανοικτού κώδικα και δικής μου έρευνας στο διαδίκτυο. Αφού έγινε η επιλογή των εργαλείων στη συνέχεια εντοπίστηκε το λογισμικό του

κάθε εργαλείου έτσι ώστε να γίνει η εγκατάσταση τους στον υπολογιστή μου. Λόγω του ότι τα περισσότερα εργαλεία δουλεύουν μόνο σε Linux λειτουργικό σύστημα, έγινε εγκατάσταση του λειτουργικού συστήματος Ubuntu 12.04.

Δεν αντιμετωπίστηκαν προβλήματα με την εγκατάσταση των εργαλείων εκτός από το εργαλείο BAT. Συγκεκριμένα το εργαλείο BAT για να μπορέσει να εφαρμόσει ανίχνευση αδειών σε ένα binary αρχείο χρειάζεται μια βάση δεδομένων όπου περιέχει τις άδειες ανοικτού λογισμικού. Όταν το πρόβλημα αυτό έγινε αντιληπτό υπήρξε επικοινωνία με το δημιουργό του εργαλείου, κύριο Armiijn Hemel, από τον οποίο πληροφορήθηκα ότι μπορούσε να μου παρέχει μια μικρή βάση δεδομένων με λίγες άδειες απλά για να μπορέσω να συμπεριλάβω το εργαλείο στην αξιολόγηση. Ο λόγος που δεν ήταν δυνατή η πρόσβαση στην κανονική βάση δεδομένων είναι ότι η βάση αυτή θα πρέπει να αγορασθεί για να μπορεί κάποιος να τη χρησιμοποιήσει.

4.3 Επιλογή των δεδομένων Εισόδου

Για την πειραματική αξιολόγηση των εργαλείων χρειάστηκαν κάποια δεδομένα που θα έπαιρναν σαν είσοδο τα εργαλεία. Τα δεδομένα αυτά έπρεπε να ήταν διάφορα projects ανοικτού κώδικα που να περιέχουν διάφορες άδειες, με σκοπό να δούμε αν τα εργαλεία που θέλουμε να συγκρίνουμε μπορούν να ανιχνεύσουν τις άδειες αυτές και να προσφέρουν χρήσιμες πληροφορίες για τα αποτελέσματα.

Τα δεδομένα εισόδου επιλέχθηκαν από διάφορες ιστοσελίδες-repositories λογισμικών ανοικτού κώδικα όπως SourceForge [6], Maven [5] και GoogleCode [4]. Για την επιλογή των δεδομένων λήφθηκαν υπόψη κάποια χαρακτηριστικά του κάθε project όπως γλώσσα /γλώσσες προγραμματισμού, μέγεθος, αριθμός γραμμών κώδικα και τέλος το είδος κάθε project. Έγινε προσπάθεια να επιλεχθούν δεδομένα με διαφορετικά χαρακτηριστικά για να εξαχθούν αποτελέσματα για όσες δυνατό περισσότερες διαφορετικές περιπτώσεις για κάθε εργαλείο.

Τα δεδομένα εισόδου είναι τα ίδια για όλα τα εργαλεία εκτός από το εργαλείο BAT και What-License?. Το εργαλείο BAT επειδή παίρνει σαν είσοδο μόνο binary αρχεία, δημιουργήθηκαν τα εκτελέσιμα αρχεία όσο το δυνατό περισσότερων projects. Λόγω της πολύ μικρής βάσης δεδομένων που υπάρχει για το BAT τα αποτελέσματα του δεν είναι ακριβής και τα αναμενόμενα. Για το λόγο αυτό δόθηκαν λίγα δεδομένα εισόδου στο BAT απλά για να δείξουμε τη λειτουργία του. Επίσης για το εργαλείο What-License? επιλέχθηκαν

κείμενα αδειών που περιέχονταν μέσα στα έργα λογισμικού του συνόλου δεδομένων λόγω του ότι το εργαλείο αυτό παίρνει σαν είσοδο μόνο κείμενο και όχι ολόκληρα έργα λογισμικού. Μετά την περιγραφή του συνόλου δεδομένων για τα υπόλοιπα εργαλεία που ακολουθεί, περιγράφονται στη συνέχεια αναλυτικά τα δεδομένα εισόδου των δύο αυτών εργαλείων.

Συνολικά τα δεδομένα εισόδου αποτελούνται από 100 projects ανοικτού κώδικα και το μέσο μέγεθος τους είναι 9.51MB.

Στον πίνακα 4.1 περιγράφονται κάποια γενικά στατιστικά στοιχεία σχετικά με τις γλώσσες προγραμματισμού που χρησιμοποιούνται στο σύνολο των δεδομένων εισόδου. Να σημειωθεί πως σε κάποια έργα χρησιμοποιούνται περισσότερες από μια γλώσσες προγραμματισμού.

Γλώσσα Προγραμματισμού	Χρήση της κάθε γλώσσας προγραμματισμού (Αριθμός έργων που τη χρησιμοποιούν)	Μέσος όρος χρήσης κάθε γλώσσας προγραμματισμού (Αριθμός έργων που τη χρησιμοποιούν/ αριθμός έργων * 100)
C	61	61%
C++	38	38%
HTML	37	37%
Python	28	28%
Javascript	21	21%
PHP	13	13%
Perl	11	11%
Java	8	8%
C#	4	4%
Assembler	4	4%
Visual Basic	2	2%
Fortran	2	2%
Pascal	1	1%
Ruby	1	1.07%

Σχήμα 4.1 Πίνακας στατιστικών στοιχείων για το σύνολο δεδομένων

Στον παρακάτω πίνακα 4.2 περιγράφονται μερικά από τα έργα λογισμικού (projects) των δεδομένων εισόδου και αναφέρονται συγκεκριμένα χαρακτηριστικά τους. Ολοκληρωμένος ο πίνακας με τα δεδομένα εισόδου βρίσκεται στο παράρτημα Α-1.

Έργο λογισμικού	Περιγραφή	Αριθμός γραμμών πηγαίου κώδικα	Γλώσσα προγραμματισμού	Πηγή	Μέγεθος (MB)
7zip	7zip is a file archiver with high compression ratio	165401	C, C++	SourceForge	5.1
Ajaxplorer-core-4.2.3	AjaXplorer is a software that can turn any web server into a powerfull file management system and an alternative to mainstream cloud storage providers.	209090	PHP, Javascript, HTML	SourceForge	14.7
qBittorrent	Multi-platform Bit Torrent client with Qt4 UI, Web UI and search engine	293238	C, C++, Python, Javascript	SourceForge	21.5
GNUCash	A personal and small-business finance manager	676429	C, C++, Python	SourceForge	72
Littlecmscolorengine	A color management library which implements fast transforms between ICC profiles	105610	C	SourceForge	7.7
MrBayes	A program for the Bayesian estimation of phylogeny	114356	C	SourceForge	6.3
A-note-source	A program that lets you to put notes on your desktop and those notes are uploaded in a database	27854	C, C++	SourceForge	6.4

Σχήμα 4.2 Πίνακας πληροφοριών για το σύνολο δεδομένων

Για το εργαλείο BAT έγινε προσπάθεια να δημιουργηθούν τα εκτελέσιμα αρχεία των έργων λογισμικού από το σύνολο δεδομένων. Συνολικά δόθηκαν σαν είσοδο 22 έργα λογισμικού τα οποία ήταν εφικτό να δημιουργηθούν τα εκτελέσιμα τους από τα 100 που περιείχε το σύνολο δεδομένων. Τα 22 έργα λογισμικού που χρησιμοποιήθηκαν ως είσοδο φαίνονται στον πίνακα 4.3.

Έργο λογισμικού	Περιγραφή	Αριθμός γραμμών πηγαίου κώδικα	Γλώσσα προγραμματισμού	Πηγή	Μέγεθος (MB)
7zip	7zip is a file archiver with high compression ratio	165401	C, C++	SourceForge	5.1
AMOS	collection of tools and class interfaces for the assembly of DNA reads	218000	, C++, Perl	SourceForge	8.9
A-note-source	A program that lets you to put notes on your desktop and those notes are uploaded in a database	27854	C, C++	SourceForge	6.4
AresGalaxy	Filesharing-Bittorrent p2p client connected to TCP supernode/leaf network and UDP DHT network	156870	Pascal	SourceForge	5.5
Artha~TheOpenThe saurus	A handy thesaurus based on WordNet with distinct features like global hotkey look-up, passive desktop notifications, regular expression based search, etc	46735	C	SourceForge	1.8
DvdFlick	A tool to convert various PC video formats to a DVD that can be played on pretty much any standalone DVD player	22200	VisualBasic	SourceForge	3.8
FAAD	the decoder of FAAC(Freeware Advanced Audio Coder	129734	C, C++	SourceForge	4.9
Flex	A tool for generating scanners programs which recognize lexical patterns in text	19480	C, C++	SourceForge	0.906
Gammu	A cellular manager for mobile phones/modems It contains libraries and functions for ringtones,logos,phonebook,S MS,etc	197149	C, C++, Python, PHP, Perl	SourceForge	41.5
GNUCash	A personal and small-business finance manager	676429	C, C++, Python	SourceForge	72

Grisbi	very functional personal financial management program with a lot of features	232706	C	SourceForge	13.3
Hunspell	A spell checker and morphological analyzer library and program designed for languages with rich morphology and complex compounding or character encoding	1459	C++	SourceForge	0.07
IOMeter	I/O performance Analysis Tool	58517	C++, C	SourceForge	6
JACOB	A JAVA-COM Bridge that allows you to call COM Automation components from Java It uses JNI to make native calls to the COM libraries	33307	HTML, Javascript	SourceForge	2.2
Mumble	A modular system for the rapid whole genome alignment of finished or draft sequence	363434	C++, C	SourceForge	17.5
MumMer	A MySQL database connector for Python programming	41778	C++, C	SourceForge	6.5
Nagios	visualization addon for the well known network managment system Nagios	178066	C, PHP, Perl	SourceForge	7.9
PonyProg;serialdev iceprogrammer	a Web Based Management tool created for Postfix	98524	C, C++	SourceForge	4.2
SmallDeviceCcomp iletsuite	compiler suite that supports a growing list of processors	532531	Assembler, C, HTML,	SourceForge	72.8
StreamRipper	records shoutcast-compatible streams	89200	C	SourceForge	3.9
UfRaw	A tool for reading and manipulating raw images from digital cameras	56342	C, C++	SourceForge	4.5
Xastir	Real-time tracking & messaging of stations via radio/internet APRS data streams, w/125 map formats supported	214783	C, Perl	SourceForge	11.3

Σχήμα 4.3 Πίνακας δεδομένων εισόδου του εργαλείου BAT

Για το εργαλείο What-License? χρησιμοποιήθηκαν κείμενα αδειών που περιέχονταν μέσα στα έργα λογισμικού του συνόλου δεδομένων. Τα κείμενα που επιλέχθηκαν είναι των κύριων αδειών που περιέχονταν μέσα στο σύνολο δεδομένων. Η επιλογή των αδειών αυτών έγινε για να διαπιστωθεί κατά πόσο το εργαλείο αυτό μπορεί να αναγνωρίσει τα κείμενα των αδειών που περιέχονταν μέσα στο σύνολο δεδομένων. Αφού η εξέταση κάθε αρχείου που περιέχεται μέσα στο σύνολο δεδομένων είναι αδύνατη με το εργαλείο αυτό, έγινε προσπάθεια να εξεταστεί με δεδομένα όσο το δυνατό πιο κοντά στο σύνολο δεδομένων και για αυτό χρησιμοποιήθηκαν τα κείμενα των αδειών που περιέχονται μέσα στο σύνολο δεδομένων. Συγκεκριμένα οι άδειες των οποίων τα κείμενα χρησιμοποιήθηκαν φαίνονται στον πίνακα 4.4.

<u>Κείμενο Άδειας</u>
GPL-2
GPL-3
LGPL-2.1
Apache-1
Apache-2
MPL-1.1
PHP-3
BSD-2
BSD-3
Artistic
Eclipse Public License
MIT License
Zlib/libpng

Σχήμα 4.4 Πίνακας δεδομένων εισόδου του εργαλείου What-license?

4.4 Επιλογή των μετρικών

Η πειραματική αξιολόγηση των εργαλείων πάνω στα δεδομένα εισόδου είχε ως στόχο τη σύγκριση των εργαλείων βάσει κάποιων μετρικών που έχουν καθοριστεί. Οι μετρικές αυτές έχουν ως στόχο να δείξουν ποιο από τα εργαλεία είναι το καλύτερο και το πιο κατάλληλο ανάλογα με το πια μετρική μας ενδιαφέρει περισσότερο.

Οι μετρικές που επιλέχθηκαν δείχνουν την ορθότητα των εργαλείων, το χρόνο εκτέλεσης τους, διάφορες λειτουργίες που θα πρέπει να προσφέρει και τέλος χαρακτηριστικά τα οποία έχει κάθε εργαλείο. Με τις μετρικές αυτές μπορεί κάποιος να πάρει όλες τις σημαντικές πληροφορίες για τη χρήση κάποιου από τα εργαλεία και να επιλέξει στη συνέχεια το εργαλείο το οποίο θεωρεί ότι είναι το κατάλληλο βάσει των μετρικών που θεωρεί ότι είναι οι πιο σημαντικές.

Οι μετρικές που έχουν επιλεγεί για τη σύγκριση των εργαλείων χωρίζονται σε δύο κατηγορίες, σε ποσοτικές και σε ποιοτικές. Οι ποσοτικές μετρικές αφορούν στοιχεία των εργαλείων τα οποία μετρούνται σε αριθμούς και στατιστικά στοιχεία ενώ οι ποιοτικές μετρικές αφορούν την ποιότητα εκτέλεσης του κάθε εργαλείου, δηλαδή τις δυνατότητες και τις προσφερόμενες λειτουργίες του.

Όλες οι μετρικές παρουσιάζονται συγκεντρωμένες στο παράρτημα Α-3. Πιο κάτω παρουσιάζονται οι μετρικές χωρισμένες στις δύο κατηγορίες που αναφέρθηκαν πιο πάνω.

Ποσοτικές Μετρικές

- Χρόνος Εκτέλεσης: Ο χρόνος που χρειάζεται ένα πρόγραμμα για να εκτελέσει μια συγκεκριμένη εργασία. Στη δική μας περίπτωση είναι ο χρόνος που χρειάζεται ένα εργαλείο να ψάξει τα αρχεία εισόδου, να ανιχνεύσει τυχόν άδειες που υπάρχουν και να παρουσιάσει τα αποτελέσματα στο χρήστη.

Η μετρική αυτή θα μετρηθεί σε δευτερόλεπτα (secs).

Τα πειράματα εκτελέστηκαν σε υπολογιστή με τα εξής χαρακτηριστικά:

Μνήμη RAM: 3 GB

Επεξεργαστής (CPU): Intel Pentium Dual CPU 2.16 GHz

- Σωστά Αποτελέσματα: Τα αποτελέσματα που προκύπτουν από την εκτέλεση ενός προγράμματος θα πρέπει να είναι σωστά. Τα εργαλεία θα πρέπει να ανιχνεύουν τις σωστές άδειες που υπάρχουν μέσα στα δεδομένα εισόδου και να μην δίνουν

λανθασμένα αποτελέσματα που πιθανό να δημιουργήσουν προβλήματα για το χρήστη.

Η μετρική αυτή χωρίζεται σε τρεις άλλες μετρικές οι οποίες δείχνουν την σχετικότητα των αποτελεσμάτων που παράγονται σε σχέση με τα επιθυμητά αποτελέσματα. Οι μετρικές αυτές χρησιμοποιούνται στην αναγνώριση προτύπων και στην ανάκτηση πληροφορίας (information retrieval) και είναι οι εξής:

- Ακρίβεια (Precision): [22] Ονομάζεται και θετική προγνωστική αξία (positive predictive value) και είναι το κλάσμα των στιγμιότυπων που ανακτήθηκαν και είναι σχετικά. Όταν ένας αλγόριθμος έχει υψηλή τιμή ακρίβειας τότε σημαίνει ότι επέστρεψε περισσότερα σχετικά αποτελέσματα από ότι άσχετα. Ο τύπος [22] για τον υπολογισμό της μετρικής αυτής έχει προσαρμοστεί για την παρούσα διπλωματική και φαίνεται στο σχήμα 4.5.

$$Precision = \frac{\text{Πλήθος σωστών αδειών που βρήκε το εργαλείο}}{\text{Συνολικό πλήθος αδειών που βρήκε το εργαλείο}}$$

Σχήμα 4.5 Τύπος υπολογισμού ακρίβειας (precision)

Τα δεδομένα που χρησιμοποιούνται από τον τύπο στο σχήμα 4.5 είναι:

- Πλήθος σωστών αδειών που βρήκε το εργαλείο: είναι ο αριθμός των αδειών που ανιχνεύθηκαν από το εργαλείο και περιέχονται όντως μέσα στα δεδομένα εισόδου.
 - Συνολικό πλήθος αδειών που βρήκε το εργαλείο: είναι ο συνολικός αριθμός των αδειών που ανιχνεύθηκαν από το εργαλείο.
- Ανάκληση (Recall): [22] Ονομάζεται και ευαισθησία (sensitivity) και είναι το κλάσμα των σχετικών στιγμιότυπων τα οποία έχουν ανακτηθεί. Όταν ένας αλγόριθμος έχει υψηλή τιμή ανάκλησης τότε σημαίνει ότι τα περισσότερα σχετικά αποτελέσματα περιέχονται μέσα αποτελέσματα που έχει επιστρέψει ο αλγόριθμος. Ο τύπος [22] για τον υπολογισμό της ανάκλησης φαίνεται στο σχήμα 4.6.

$$Recall = \frac{\text{Πλήθος σωστών αδειών που βρήκε το εργαλείο}}{\text{Πλήθος σωστών αδειών των δεδομένων εισόδου}}$$

Σχήμα 4.6 Τύπος υπολογισμού ανάκλησης (recall)

Τα δεδομένα που χρησιμοποιούνται από τον τύπο στο σχήμα 4.6 είναι:

- Πλήθος σωστών αδειών που βρήκε το εργαλείο: είναι ο αριθμός των αδειών που ανιχνεύθηκαν από το εργαλείο και περιέχονται όντως μέσα στα δεδομένα εισόδου.
- Πλήθος αδειών των δεδομένων εισόδου: είναι ο αριθμός των αδειών που περιέχονται μέσα στα δεδομένα εισόδου.

➤ F-Measure: [22] Η μετρική αυτή είναι ο αρμονικός μέσος των δύο προηγούμενων μετρικών, ακρίβειας και ανάκλησης. Με τη μετρική αυτή συνδυάζονται οι δύο προηγούμενες μετρικές και παρουσιάζεται μια σύνοψη των δύο μετρικών για το κάθε εργαλείο. Ο τύπος [22] για τον υπολογισμό της φαίνεται στο σχήμα 4.7.

$$F - Measure = 2 \times \frac{precision \times recall}{precision + recall}$$

Σχήμα 4.7 Τύπος υπολογισμού F-Measure

- Αριθμός υποστηριζόμενων προγραμματιστικών γλωσσών: Ο αριθμός των προγραμματιστικών γλωσσών που υποστηρίζονται από ένα εργαλείο. Κάθε project στα δεδομένα εισόδου μπορεί να είναι γραμμένο σε πολλές προγραμματιστικές γλώσσες και έτσι ένα εργαλείο θα πρέπει να είναι σε θέση να ελέγχει την πλειοψηφία, αν όχι όλα, τα αρχεία κώδικα τα οποία συμπεριλαμβάνονται σε κάθε project.
- Αριθμός υποστηριζόμενων αδειών: Ο αριθμός των αδειών που υποστηρίζει και μπορεί να αναγνωρίσει κάθε εργαλείο.

Ποιοτικές Μετρικές

- Στατιστικά Στοιχεία: Η δυνατότητα ενός εργαλείου να παρουσιάζει στατιστικά στοιχεία που αφορούν την ανάλυση ενός έργου λογισμικού. Για τα εργαλεία μερικά στατιστικά στοιχεία είναι:
 - Χρόνος που χρειάστηκε για την εκτέλεση του
 - Αριθμός και ποσοστό εμφανίσεων κάθε αναγνωρισμένης άδειας
 - Συνολικός αριθμός αρχείων που έχουν ελεγχθεί
- Αναφορά προβλημάτων: Όταν υπάρχει κάποιο πρόβλημα κατά την εκτέλεση ενός προγράμματος, το πρόγραμμα αυτό θα πρέπει να αναφέρει ποιο είναι το πρόβλημα που προέκυψε. Επίσης όταν κάποιο αρχείο περιέχει μια άδεια στην οποία δεν αντιστοιχεί πλήρως το κείμενο της αυθεντικής άδειας τότε το εργαλείο θα πρέπει να αναφέρει άδειες που μοιάζουν στο κείμενο αυτό ή θα πρέπει να βγάζει μήνυμα ότι δεν υπήρχε άδεια που να ταιριάζει στο κείμενο του αρχείου εισόδου.
- Παρουσίαση άδειας κάθε αρχείου: Για κάθε αρχείο που ελέγχεται από το εργαλείο θα πρέπει να παρουσιάζεται η άδεια που περιέχεται στο αρχείο αυτό και όχι μόνο η γενική άδεια του package.
- Ανίχνευση παραβιάσεων συμβατότητας μεταξύ των αδειών: Αν ένα project περιέχει περισσότερες από μια άδειες τότε είναι πιθανό να υπάρχουν παραβιάσεις στη συμβατότητα μεταξύ των αδειών αυτών. Ένα εργαλείο θα πρέπει να μπορεί να ανιχνεύει τις παραβιάσεις αυτές και να τις αναφέρει στο χρήστη.

4.5 Αποτελέσματα πειραμάτων

Για κάθε εργαλείο, από τα εργαλεία που επιλέχθηκαν για τη σύγκριση, δόθηκε σαν είσοδο όλα τα δεδομένα εισόδου και μέσα από τα πειράματα που διενεργήθηκαν, παράχθηκαν σημαντικά αποτελέσματα για την ορθότητα των εργαλείων και για το χρόνο εκτέλεσης τους. Για τα εργαλεία BAT και What-License έγιναν ξεχωριστά πειράματα, λόγω των περιορισμένων αποτελεσμάτων του BAT ως αποτέλεσμα της μικρής βάσης δεδομένων του και λόγω του ότι το εργαλείο What-License παίρνει σαν είσοδο μόνο ένα κομμάτι κειμένου άρα θα ήταν αδύνατο να εξετασθούν όλα τα αρχεία του συνόλου δεδομένων.

Ορθότητα Αποτελεσμάτων-Ανίχνευση Αδειών

Για την ορθότητα αποτελεσμάτων χρησιμοποιήθηκαν οι τρεις μετρικές precision, recall και f-measure που αναφέρθηκαν πιο πάνω. Για κάθε ένα από τα projects που περιέχονται μέσα στα δεδομένα εισόδου υπολογίζονταν οι τιμές για κάθε μια από τις μετρικές βάσει των αποτελεσμάτων του κάθε εργαλείου. Αφού υπολογίστηκαν οι τιμές για κάθε project για κάθε ένα από τα εργαλεία στη συνέχεια υπολογίστηκε ο μέσος όρος των τιμών για τις μετρικές αυτές.

Λόγω του ότι οι περισσότερες άδειες που περιέχονται μέσα στα δεδομένα εισόδου έχουν συγκεκριμένη έκδοση και ένα από τα εργαλεία έβγαζε ως αποτέλεσμα μόνο το όνομα της άδειας χωρίς συγκεκριμένη έκδοση τα αποτελέσματα χωρίζονται σε δύο περιπτώσεις. Στην πρώτη περίπτωση οι εκδόσεις των αδειών δεν παίζουν ρόλο, δηλαδή βλέπουμε μόνο να είναι σωστό το όνομα της άδειας, ενώ στη δεύτερη λαμβάνουμε υπόψη και την έκδοση των αδειών για να θεωρηθεί σωστό το αποτέλεσμα. Λόγω των δύο αυτών περιπτώσεων οι υπολογισμοί για την ορθότητα των αποτελεσμάτων του κάθε εργαλείου έγιναν δύο φορές για να δοθεί μια πιο καλή εικόνα της λειτουργίας των εργαλείων.

Πρώτη Περίπτωση-Χωρίς Εκδόσεις

Τα αποτελέσματα για κάθε εργαλείο για όλα τα δεδομένα εισόδου χωρίς να λαμβάνεται υπόψη την έκδοση των αδειών φαίνονται στον πίνακα 4.8.

	<u>Ohcount</u>	<u>Fossology</u>	<u>OSLC</u>	<u>ASLA</u>	<u>Ninka</u>
Ακρίβεια (Precision)	0.81	0.81	0.51	0.46	0.80
Ανάκληση (Recall)	0.77	0.92	0.61	0.37	0.75
F-measure	0.77	0.84	0.52	0.38	0.75

Σχήμα 4.8 Πίνακας αποτελεσμάτων χωρίς εκδόσεις

Δεύτερη Περίπτωση-Με Εκδόσεις

Τα αποτελέσματα για κάθε εργαλείο για όλα τα δεδομένα εισόδου λαμβάνοντας υπόψη την έκδοση των αδειών φαίνονται στον πίνακα 4.9.

	<u>Ohcount</u>	<u>Fossology</u>	<u>OSLC</u>	<u>ASLA</u>	<u>Ninka</u>
Ακρίβεια (Precision)	0.34	0.81	0.51	0.46	0.80
Ανάκληση (Recall)	0.33	0.92	0.61	0.37	0.75
F-measure	0.33	0.84	0.52	0.38	0.75

Σχήμα 4.9 Πίνακας αποτελεσμάτων με εκδόσεις

Από τους πιο πάνω πίνακες φαίνεται ότι οι τιμές των μετρικών μένουν οι ίδιες και για τις δύο περιπτώσεις για όλα τα εργαλεία εκτός από το Ohcount. Το συγκεκριμένο εργαλείο στα περισσότερα αποτελέσματα του όσον αφορά την άδεια GPL δεν έβγαζε την έκδοση της άδειας. Για το λόγο αυτό υπάρχει τόση μεγάλη μείωση στις τιμές για τη δεύτερη περίπτωση στην οποία λαμβάνεται υπόψη και η έκδοση της άδειας. Τα υπόλοιπα εργαλεία για όλες τις άδειες που έβγαζαν σαν αποτέλεσμα υπήρχε και η έκδοση της άδειας μαζί και γι' αυτό δε βλέπουμε διαφορά στις τιμές των μετρικών για τις δύο περιπτώσεις.

Εργαλείο BAT

Λόγω της πολύ μικρής βάση δεδομένων του, τα αποτελέσματα του εργαλείου BAT δεν είναι αντιπροσωπευτικά της κανονικής λειτουργίας του. Συγκεκριμένα από τα 22 έργα λογισμικού, το BAT κατάφερε να ανιχνεύσει τη σωστή άδεια που περιέχεται μόνο στα 5 ενώ στα υπόλοιπα έργα λογισμικού το BAT έδειξε ότι δεν περιέχουν κάποια άδεια. Επίσης και στα 5 αυτά λογισμικά το αποτέλεσμα ήταν το ίδιο, αφού αν και είχαν διαφορετική έκδοση άδειας, η άδεια του κάθε έργου άνηκε στην οικογένεια GNU GPL και GNU LGPL και έτσι το αποτέλεσμα του εργαλείου BAT ήταν GNU. Συγκεκριμένα για τα 5 αυτά έργα λογισμικού φαίνονται τα αποτελέσματα στον πιο κάτω πίνακα 4.10.

Έργο λογισμικού	Άδεια που περιέχει	Αποτέλεσμα BAT
7zip	GNU LGPLv2	GNU
Artha	GPL v2.0	GNU
FAAD	GPL v2.0	GNU
Grisbi	GPL v2.0 or later	GNU
Nagios	GPL v2.0	GNU

Σχήμα 4.10 Πίνακας αποτελεσμάτων BAT

Στον πιο κάτω πίνακα 4.11 φαίνονται τα αποτελέσματα του εργαλείου BAT με βάσει τις μετρικές ακρίβεια (precision), ανάκληση (recall) και f-measure. Λόγω του ότι το BAT στα αποτελέσματα εμφάνιζε μόνο την οικογένεια των αδειών που ανίχνευε ο υπολογισμός των τιμών των μετρικών για την ορθότητα του εργαλείου, υπολογίστηκαν βάσει του αν βρήκε σωστά την οικογένεια των αδειών.

	Ακρίβεια (Precision)	Ανάκληση (Recall)	F-measure
Οικογένεια αδειών	0.22	0.15	0.18

Σχήμα 4.11 Πίνακας αποτελεσμάτων BAT

Από τα αποτελέσματα του πιο πάνω πίνακα φαίνεται ότι το BAT λόγω του ότι περιέχει λίγες άδειες στη βάση που χρησιμοποιούμε κατάφερε να εντοπίσει τις άδειες αλλά στο αποτέλεσμα παρουσιάζει μόνο τη οικογένεια της άδειας χωρίς λεπτομέρειες για το ποια άδεια είναι ακριβώς και την έκδοση της. Αν υπολογίζαμε τις τιμές λαμβάνοντας υπόψη τις άδειες και τις εκδόσεις τους τότε τα αποτελέσματα θα ήταν 0 και στις 3 μετρικές αφού το BAT έβγαζε σαν αποτέλεσμα μόνο την οικογένεια κάποιας άδειας, άρα για καμία άδεια δεν θα είχε σωστό αποτέλεσμα αφού ποτέ δεν θα εμφάνιζε το όνομα και την έκδοση μιας άδειας. Οι τιμές και των τριών μετρικών, αν και υπολογίστηκαν βάσει του αν βρήκε σωστά την οικογένεια της κάθε άδειας, είναι πολύ χαμηλές, γεγονός που σημαίνει ότι το BAT λόγω της μικρής βάσης δεδομένων του δεν κατάφερε να βρει τις περισσότερες άδειες.

Εργαλείο What-License?

Το εργαλείο αυτό αναφέρει τα αποτελέσματα του σε ποσοστά. Δηλαδή για κάθε κείμενο που ελέγχει αναφέρει τις πιθανές άδειες που περιέχει το κείμενο αυτό σε ποσοστό ταυτοποίησης. Για να διαπιστωθεί η ορθότητα των αποτελεσμάτων επιλέγονταν οι άδειες με τα πιο ψηλά ποσοστά ταυτοποίησης. Οι άδειες με ποσοστά μικρότερα από 15% αγνοούνταν όταν υπήρχαν ποσοστά μεγαλύτερα από 85%.

Όπως και για τα υπόλοιπα εργαλεία τα αποτελέσματα χωρίστηκαν σε 2 περιπτώσεις: λαμβάνοντας υπόψη την έκδοση της άδειας και μη λαμβάνοντας υπόψη την έκδοση. Τα αποτελέσματα φαίνονται στον πίνακα 4.12.

	Ακρίβεια (Precision)	Ανάκληση (Recall)	F-measure
Με εκδόσεις	0.26	0.30	0.27
Χωρίς εκδόσεις	0.30	0.46	0.34

Σχήμα 4.12 Πίνακας αποτελεσμάτων What-License?

Από τον πιο πάνω πίνακα φαίνεται ότι οι τιμές των μετρικών για το εργαλείο αυτό είναι αρκετά χαμηλές. Επίσης φαίνεται ότι λαμβάνοντας υπόψη τις εκδόσεις των αδειών τα ποσοστά στις 3 μετρικές μειώνονται. Αυτό συμβαίνει γιατί σε μερικές άδειες όπως η BSD, το εργαλείο δεν παρουσιάζει την έκδοση της άδειας, π.χ. BSD2 και BSD3 αλλά μόνο BSD.

Ορθότητα Αποτελεσμάτων-Ανίχνευση Παραβιάσεων Συμβατότητας

Τη δυνατότητα ανίχνευσης παραβιάσεων μεταξύ των αδειών που έχουν ανιχνευθεί την προσφέρουν μόνο τα εργαλεία OSLC και ASLA. Από τα έργα λογισμικού που περιέχονταν μέσα στο σύνολο δεδομένων και τα 2 εργαλεία κατόρθωσαν να ανιχνεύσουν σωστά τις παραβιάσεις μεταξύ των αδειών και να τις παρουσιάσουν στο χρήστη. Για κάθε σύνολο παραβιάσεων που ανίχνευσε το εργαλείο γινόταν στη συνέχεια έλεγχος χειροκίνητα μέσω του γράφου συμβατότητας [11] για να διαπιστωθεί κατά πόσο οι παραβιάσεις που ανιχνεύθηκαν ήταν σωστές. Επίσης για τις άδειες που ανιχνεύονταν και δεν υπήρχαν κάποιες παραβιάσεις γινόταν έλεγχος αν όντως οι άδειες αυτές είναι συμβατές μεταξύ τους και πάλι μέσω του γράφου συμβατότητας. Και για τα 2 εργαλεία, μετά τους ελέγχους που έγιναν για την ορθότητα των αποτελεσμάτων για την ανίχνευση παραβιάσεων, αποδείχθηκε ότι οι παραβιάσεις που ανιχνεύθηκαν ήταν ορθές.

Χρόνος Εκτέλεσης

Για κάθε εργαλείο υπολογιζόταν ο χρόνος εκτέλεσης του για κάθε ένα από τα project από τα δεδομένα εισόδου. Στη συνέχεια υπολογίστηκε ο χρόνος εκτέλεσης σε δευτερόλεπτα του κάθε εργαλείου ανά 1 MB. Συγκεκριμένα τα αποτελέσματα φαίνονται στον πίνακα 4.13. Στην επόμενη ενότητα ακολουθεί σχολιασμός και σύγκριση των αποτελεσμάτων αυτών.

	<u>Ohcount</u>	<u>Fossology</u>	<u>OSLC</u>	<u>ASLA</u>	<u>Ninka</u>	<u>BAT</u>
Χρόνος Εκτέλεσης (secs) per MB	2.35	13.99	1.35	2.87	19.26	3.32

Σχήμα 4.13 Πίνακας αποτελεσμάτων χρόνου εκτέλεσης

Εργαλείο BAT

Όπως και για τα υπόλοιπα εργαλεία υπολογιζόταν ο χρόνος εκτέλεσης για κάθε ένα από τα έργα λογισμικού που έπαιρνε σαν είσοδο το εργαλείο BAT και στη συνέχεια υπολογίστηκε ο μέσος χρόνος εκτέλεσης του σε δευτερόλεπτα ανά 1 MB. Συγκεκριμένα ο μέσος χρόνος εκτέλεσης του εργαλείου BAT είναι 3.32 δευτερόλεπτα ανά 1 MB.

Εργαλείο What-License?

Για όλα τα κείμενα που δόθηκαν σαν είσοδο στο εργαλείο οι χρόνοι ήταν σχεδόν οι ίδιοι με πολύ μικρή διαφορά για το κάθε κείμενο. Συγκεκριμένα ο μέσος χρόνος που χρειάστηκε το εργαλείο για να αναγνωρίσει τις άδειες σε ένα κείμενο είναι 4.34 δευτερόλεπτα.

4.6 Σύγκριση

Η σύγκριση των εργαλείων γίνεται βάσει των μετρικών που έχουν καθοριστεί στην ενότητα 4.4 και των αποτελεσμάτων που παράχθηκαν από τα πειράματα.

Χρόνος Εκτέλεσης

Για τη σύγκριση των εργαλείων ως προς το χρόνο εκτέλεσης τους θα χρησιμοποιηθεί ο πίνακας 4.13 που παρουσιάστηκε πιο πάνω, ο οποίος περιέχει το χρόνο εκτέλεσης κάθε εργαλείου για 1 MB. Από τον πίνακα φαίνεται ότι τον μικρότερο χρόνο εκτέλεσης έχει το εργαλείο OSLC με μόλις 1.35 δευτερόλεπτα. Στη συνέχεια έχουμε το εργαλείο Ohcount με 2.35 δευτερόλεπτα και μετά με πολύ μικρή διαφορά το εργαλείο ASLA με 2.87 δευτερόλεπτα. Τα υπόλοιπα εργαλεία έχουν αρκετή μεγάλη διαφορά από τα τρία γρηγορότερα αφού ακολουθεί το Fossology με 13.99 δευτερόλεπτα και στο τέλος έχουμε το εργαλείο Ninka με τον μεγαλύτερο χρόνο εκτέλεσης, 19.26 δευτερόλεπτα. Το BAT με χρόνο εκτέλεσης 3.32 δευτερόλεπτα κατατάσσεται τέταρτο στη σειρά αλλά λόγω του ότι ο υπολογισμός του μέσου χρόνου έγινε με λιγότερα αρχεία και η λειτουργία του δεν έφτανε τις πλήρεις δυνατότητες του λόγω της μικρής βάσης δεδομένων, προτιμήθηκε να αναφερθεί απλά ο χρόνος εκτέλεσης του χωρίς να γίνει σύγκριση με τα υπόλοιπα εργαλεία. Επίσης το εργαλείο What-License? Λόγω του ότι ελέγχει απλά ένα κομμάτι κειμένου και όχι ολόκληρα έργα λογισμικού δεν θα μπορούσε να συμπεριληφθεί στη σύγκριση με τα υπόλοιπα εργαλεία.

Η σύγκριση των εργαλείων ως προς την ορθότητα των αποτελεσμάτων τους θα γίνει με τη χρήση των τιμών των τριών μετρικών που έχουν παρουσιαστεί στους πίνακες 4.8 και 4.9. Η ανώτατη τιμή που μπορεί κάποιο εργαλείο να έχει στις τρεις αυτές μετρικές είναι το 1 και η χαμηλότερη 0. Όσο πιο ψηλές τιμές έχουν τα εργαλεία στις μετρικές αυτές τόσο πιο καλά και ορθά είναι τα αποτελέσματα που παράγουν. Συγκεκριμένα ψηλή τιμή ακρίβειας (precision) σημαίνει ότι τα σχετικά αποτελέσματα, δηλαδή οι άδειες, που επέστρεψε κάποιο εργαλείο είναι περισσότερα από ότι τα άσχετα αποτελέσματα που επέστρεψε. Η ψηλή τιμή ανάκλησης (recall) δηλώνει ότι τα περισσότερα σχετικά αποτελέσματα, δηλαδή οι περισσότερες σχετικές άδειες, περιέχονται μέσα αποτελέσματα που έχει επιστρέψει κάποιο εργαλείο. Τέλος αν κάποιο εργαλείο έχει ψηλό f-measure τόσο το καλύτερο αφού η μετρική αυτή είναι ένας συνδυασμός των δύο προηγούμενων.

Για την πρώτη περίπτωση στην οποία δεν λαμβάνεται υπόψη η έκδοση των αδειών στα αποτελέσματα, οι τιμές για το κάθε εργαλείο φαίνονται στον πίνακα 4.8. Από τον πίνακα φαίνεται ότι τις ψηλότερες τιμές, 0.81, 0.92, 0.84, και για τις τρεις μετρικές τις έχει το εργαλείο Fossology. Άρα αυτό σημαίνει ότι από όλα τα εργαλεία που δοκιμάσαμε το Fossology έχει την καλύτερη επίδοση ως προς την ορθότητα των αποτελεσμάτων. Μετά το Fossology τις ψηλότερες τιμές, 0.81, 0.77, 0.77, έχει το εργαλείο Ohcount και με πολύ μικρή διαφορά ακολουθεί το εργαλείο Ninka με τιμές 0.80, 0.75, 0.75. Στη συνέχεια είναι το OSLC με τιμές 0.51, 0.61, 0.52 και τέλος, με τις χαμηλότερες τιμές, 0.46, 0.37, 0.38, είναι το εργαλείο ASLA.

Για την δεύτερη περίπτωση στην οποία λαμβάνεται υπόψη η έκδοση των αδειών στα αποτελέσματα, οι τιμές για το κάθε εργαλείο φαίνονται στον πίνακα 4.9. Οι τιμές για όλα τα εργαλεία παραμένουν οι άδειες εκτός από το εργαλείο Ohcount. Από τον πίνακα φαίνεται ότι τις ψηλότερες τιμές και για τις τρεις μετρικές τις έχει, όπως στην πρώτη περίπτωση, το εργαλείο Fossology. Σε αυτή την περίπτωση αλλάζει η κατάταξη των υπόλοιπων εργαλείων αφού το Ohcount που ήταν πριν το δεύτερο καλύτερο έχει τώρα πολύ μειωμένες τιμές σε σχέση με την πρώτη περίπτωση. Συγκεκριμένα τώρα δεύτερο είναι το εργαλείο Ninka, στη συνέχεια ακολουθεί το εργαλείο OSLC, μετά βρίσκεται το εργαλείο ASLA και το Ohcount βρίσκεται τώρα στην τελευταία θέση με τιμές 0.34, 0.33, 0.33. Ο λόγος που το Ohcount παρουσίασε τόσο χαμηλές τιμές σε αυτή την περίπτωση είναι ότι στις περισσότερες περιπτώσεις δεν συμπεριλαμβάνεται στο αποτέλεσμα η έκδοση της άδειας αλλά μόνο το όνομα της. Συγκεκριμένα για την άδεια GPL σχεδόν πάντα δεν βγάζει τις διάφορες εκδόσεις

της άδειας εκτός αν πρόκειται για την έκδοση GPL_v3. Για τις υπόλοιπες εκδόσεις της συγκεκριμένης άδειας το αποτέλεσμα είναι μόνο GPL.

Εργαλείο BAT

Στα 5 έργα λογισμικού που παρουσίασε αποτέλεσμα το εργαλείο BAT, το αποτέλεσμα ήταν GNU. Και τα 5 έργα περιείχαν άδειες που ανήκουν στην οικογένεια GNU αλλά με διαφορετικές εκδόσεις. Άρα το εργαλείο βρήκε σωστά την οικογένεια των αδειών αλλά αδυνατούσε να παρουσιάσει την ακριβή έκδοση της άδειας.

Εργαλείο What-License?

Το εργαλείο αυτό είχε σχετικά χαμηλά ποσοστά στις μετρικές που καθορίζουν την ορθότητα των αποτελεσμάτων του. Αυτό συμβαίνει λόγω του ότι σε μερικές περιπτώσεις δεν αναγνωρίζει την σωστή άδεια όπως τις άδειες Artistic, Eclipse Public Domain, Mozilla Public License και zlib/libpng. Επίσης το εργαλείο αδυνατεί για μερικές άδειες όπως η BSD να παρουσιάσει στο αποτέλεσμα και την έκδοση της άδειας. Δηλαδή για είσοδο κειμένου της άδειας BSD2 ή BSD3 το αποτέλεσμα είναι BSD.

Στατιστικά Στοιχεία

Με την μετρική αυτή παρουσιάζεται η δυνατότητα των εργαλείων να παρουσιάζουν στατιστικά στοιχεία σε σχέση με την εκτέλεση της αναγνώρισης αδειών και ανίχνευσης παραβιάσεων συμβατότητας. Είναι σημαντικό ο χρήστης να μπορεί να βλέπει στατιστικά στοιχεία που αφορούν την ανάλυση ενός έργου λογισμικού γιατί έτσι είναι πιο εύκολο για αυτόν να καταλάβει τα αποτελέσματα που εξάγει κάθε εργαλείο, να καταλήξει σε συμπεράσματα και να συγκρίνει αν επιθυμεί στη συνέχεια τα διάφορα εργαλεία βάσει των στατιστικών αυτών. Με τη δυνατότητα παρουσίασης στατιστικών στοιχείων, ένα εργαλείο διευκολύνει το χρήστη πολύ στην κατανόηση των αποτελεσμάτων του, κάτι που θα ήταν πολύ χρονοβόρο αν έπρεπε ο χρήστης να παράξει μόνος του τα στατιστικά στοιχεία για ένα μεγάλο σύνολο δεδομένων και στη συνέχεια να τα μελετήσει για να τα κατανοήσει. Τα στατιστικά στοιχεία λοιπόν είναι μια σημαντική λειτουργία που θα έπρεπε να όλα τα εργαλεία να την προσφέρουν γιατί κάνουν τη χρήση του εργαλείου πιο εύκολη και πιο κατανοητή, που αυτός θα έπρεπε να ήταν και ένας από τους κύριους στόχους ενός εργαλείου. Από τα εργαλεία που συγκρίναμε μόνο μερικά παρέχουν τη δυνατότητα αυτή. Συγκεκριμένα

ο πίνακας 4.14 που ακολουθεί παρουσιάζει ποια από τα εργαλεία παρέχουν τη δυνατότητα παρουσίασης στατιστικών στοιχείων των αποτελεσμάτων.

<u>Εργαλεία</u>	<u>Στατιστικά στοιχεία αποτελεσμάτων</u>	
	YES	NO
Ohcount		√
Fossology	√	
OSLC	√	
ASLA	√	
Ninka		√
BAT		√
What-License?		√

Σχήμα 4.14 Πίνακας δυνατότητας εμφάνισης στατιστικών στοιχείων αποτελεσμάτων

Στον επόμενο πίνακα 4.15 φαίνονται τα στατιστικά στοιχεία που παρέχει το κάθε εργαλείο για τα αποτελέσματα που παράγονται.

<u>Εργαλεία</u>	<u>Στατιστικά στοιχεία αποτελεσμάτων</u>
Fossology	-Άδειες που αναγνωρίστηκαν -Συνολικός αριθμός αρχείων που περιέχουν την κάθε άδεια που ανιχνεύθηκε -Λίστα με τα αρχεία που περιέχουν κάθε άδεια που αναγνωρίστηκε
OSLC	-Συνολικός αριθμός αρχείων -Συνολικός αριθμός αρχείων που ελέγχθηκαν -Συνολικός αριθμός αδειών που ανιχνεύθηκαν -Συνολικός αριθμός παραβιάσεων που ανιχνεύθηκαν -Συνολικός αριθμός αρχείων που περιέχουν την κάθε άδεια που ανιχνεύθηκε
ASLA	-Αριθμός αρχείων πηγαίου κώδικα στο μονοπάτι εισόδου -Αριθμός αρχείων πηγαίου κώδικα που χρησιμοποιήθηκαν από το μονοπάτι εισόδου -Συνολικός αριθμός αρχείων πηγαίου κώδικα -Άδειες που αναγνωρίστηκαν -Άδειες που αναγνωρίστηκαν από αρχεία στο μονοπάτι εισόδου -Συνολικός αριθμός αρχείων που περιέχουν την κάθε άδεια που ανιχνεύθηκε

Σχήμα 4.15 Πίνακας στατιστικών στοιχείων αποτελεσμάτων

Αριθμός υποστηριζόμενων προγραμματιστικών γλωσσών

Η μετρική αυτή παρουσιάζει τον αριθμό των προγραμματιστικών γλωσσών που υποστηρίζει το κάθε εργαλείο. Όσο πιο μεγάλος είναι ο αριθμός αυτό τόσο περισσότερα έργα λογισμικού θα μπορεί να αναλύει το εργαλείο. Στον πίνακα 4.16 φαίνεται ο αριθμός των προγραμματιστικών γλωσσών που υποστηρίζει το κάθε εργαλείο.

<u>Εργαλεία</u>	<u>Αριθμός υποστηριζόμενων γλωσσών προγραμματισμού</u>
Ohcount	70+
Fossology	Όλοι οι τύποι αρχείων
OSLC	5
ASLA	8
Ninka	Όλοι οι τύποι αρχείων
BAT	5
What-License?	Κείμενο

Σχήμα 4.16 Πίνακας υποστηριζόμενων γλωσσών προγραμματισμού

Συγκεκριμένα για το εργαλείο OSLC οι γλώσσες προγραμματισμού και οι τύποι αρχείων που υποστηρίζονται είναι οι εξής: Java, PHP, C, C++, Linux kernel αρχεία πηγαίου κώδικα (source files), αρχεία κειμένου (text files) [23].

Για το εργαλείο ASLA οι γλώσσες προγραμματισμού που υποστηρίζονται είναι όλες οι γλώσσες οι οποίες μπορούν να μεταγλωτιστούν (compile) μέσω του μεταγλωτιστή (compiler) GCC. Οι γλώσσες αυτές είναι: C, C++, Objective-C, Objective-C++, Java, Fortran, Ada, και Go [24].

Τέλος για το εργαλείο BAT ELF, bFLT (for MMU-less embedded Linux systems, only sporadically used), Java .class files, Dalvik και PE (for Windows binaries).

Αναφορά προβλημάτων

Αν σε κάποιο αρχείο δεν περιέχεται κάποια άδεια τότε το εργαλείο θα πρέπει να εμφανίζει μήνυμα ότι το συγκεκριμένο αρχείο δεν περιέχει κάποια άδεια. Στον πίνακα 4.17 φαίνεται ποια από τα εργαλεία αναφέρουν αν κάποιο αρχείο δεν περιέχει κάποια άδεια.

<u>Εργαλεία</u>	<u>Αναφορά αν κάποιο αρχείο δεν περιέχει άδεια</u>	
	YES	NO
Ohcount		√
Fossology	√	
OSLC	√	
ASLA	√	
Ninka	√	
BAT	√	

Σχήμα 4.17 Πίνακας αναφοράς αν δεν περιέχεται κάποια άδεια

Από τον πίνακα πιο πάνω φαίνεται ότι όλα τα εργαλεία ενημερώνουν το χρήστη αν κάποιο αρχείο δεν περιέχει κάποια άδεια εκτός από το εργαλείο Ohcount. Το συγκεκριμένο εργαλείο εμφανίζει μόνο τα αρχεία που περιέχουν κάποια άδεια στα αποτελέσματα του. Επίσης το εργαλείο What-License? λόγω του ότι παίρνει σαν είσοδο μόνο ένα κομμάτι κειμένου και όχι ένα σύνολο από αρχεία δεν μπορούσε να συμπεριληφθεί στη σύγκριση για αυτή την μετρική.

Ένα εργαλείο αν δε βρει άδεια σε κάποιο κείμενο που να ταιριάζει απόλυτα με κάποιο από τα πρότυπα των αδειών του, τότε θα πρέπει να προτείνει άδειες που ίσως να ταιριάζουν με το συγκεκριμένο κείμενο. Στον πίνακα 4.18 φαίνεται ποια από τα εργαλεία προτείνουν άδειες που μοιάζουν με το κείμενο εισόδου.

<u>Εργαλεία</u>	<u>Προτάσεις πιθανών αδειών</u>	
	YES	NO
Ohcount		√
Fossology		√
OSLC	√	
ASLA		√
Ninka	√	
BAT		√
What-License?	√	

Σχήμα 4.18 Πίνακας προτάσεων πιθανών αδειών

Παρουσίαση άδειας κάθε αρχείου

Τα έργα λογισμικού που παίρνουν ως είσοδο τα εργαλεία αποτελούνται από πολλά αρχεία. Μετά από τον έλεγχο κάποιου έργου λογισμικού ένα εργαλείο θα πρέπει να παρουσιάζει για το κάθε αρχείο την άδεια που ανίχνευσε. Στον πιο κάτω πίνακα 4.19 φαίνεται ποια από τα εργαλεία παρουσιάζουν τις άδειες που περιέχονται σε κάθε αρχείο.

<u>Εργαλεία</u>	<u>Παρουσίαση άδειας κάθε αρχείου</u>	
	YES	NO
Ohcount	√	
Fossology	√	
OSLC	√	
ASLA	√	
Ninka	√	
BAT		√

Σχήμα 4.19 Πίνακας παρουσίασης άδειας κάθε αρχείου

Από τον πιο πάνω πίνακα 4.19 φαίνεται ότι όλα τα εργαλεία εκτός από το BAT παρουσιάζουν στα αποτελέσματα τους τις άδειες που περιέχονται σε κάθε αρχείο ενός έργου λογισμικού. Το εργαλείο BAT παρουσιάζει μόνο τις άδειες που ανιχνεύθηκαν γενικά στο έργο λογισμικού και όχι τις άδειες που περιέχονται σε κάθε αρχείο. Το εργαλείο What-License? λόγω του ότι παίρνει σαν είσοδο μόνο ένα κομμάτι κειμένου και όχι ένα σύνολο από αρχεία δεν μπορούσε να συμπεριληφθεί στη σύγκριση για αυτή την μετρική.

Αριθμός υποστηριζόμενων αδειών

Η μετρική αυτή παρουσιάζει τον αριθμό των αδειών που υποστηρίζει το κάθε εργαλείο. Όσο πιο μεγάλος είναι ο αριθμός αυτό τόσο περισσότερες άδειες υποστηρίζει το εργαλείο άρα τόσο περισσότερες άδειες μπορεί να ανιχνεύσει. Στον πίνακα 4.20 φαίνεται ο αριθμός των αδειών που υποστηρίζει το κάθε εργαλείο.

<u>Εργαλεία</u>	<u>Αριθμός υποστηριζόμενων αδειών</u>
Ohcount	71
Fossology	618
OSLC	44
ASLA	37
Ninka	112
BAT	17

Σχήμα 4.20 Πίνακας υποστηριζόμενων αδειών

Από τον πιο πάνω πίνακα 4.20 φαίνεται ότι τις περισσότερες υποστηριζόμενες άδειες τις έχει το εργαλείο Fossology. Ακολουθεί στη συνέχεια το εργαλείο Ninka, μετά το Ohcount, το OSLC, το ASLA και τέλος είναι το εργαλείο BAT. Το εργαλείο What-License? λόγω του ότι είναι ένα online εργαλείο και δεν υπάρχουν άρθρα για αυτό ή κάποιες άλλες πηγές δεν ήταν δυνατό να βρεθεί κάπου η πληροφορία για τις υποστηριζόμενες άδειες του.

Ανίχνευση παραβιάσεων συμβατότητας μεταξύ των αδειών

Η μετρική αυτή δείχνει τη δυνατότητα ανίχνευσης παραβιάσεων μεταξύ της συμβατότητας των αδειών που αναγνωρίζονται στα αποτελέσματα των εργαλείων. Στον πιο κάτω πίνακα 4.21 φαίνεται ποια από τα εργαλεία παρέχουν τη δυνατότητα αυτή στους χρήστες.

<u>Εργαλεία</u>	<u>Ανίχνευση παραβιάσεων συμβατότητας</u>	
	YES	NO
Ohcount		√
Fossology		√
OSLC	√	
ASLA	√	
Ninka		√
BAT		√
What-License?		√

Σχήμα 4.21 Πίνακας ανίχνευσης παραβιάσεων συμβατότητας

Κεφάλαιο 5

Υλοποίηση Εργαλείων

5.1 Εισαγωγή	39
5.2 Εργαλείο Μετατροπéας (Converter)	40
5.2.1 Κίνητρο για υλοποίηση	40
5.2.2 Περιγραφή Εργαλείου	45
5.2.3 Υλοποίηση Εργαλείου	49
5.3 Εργαλείο Ελεγκτής Συμβατότητας (Compatibility Check)	50
5.3.1 Κίνητρο για υλοποίηση	50
5.3.2 Περιγραφή Εργαλείου	50
5.3.3 Υλοποίηση Εργαλείου	53

5.1 Εισαγωγή

Στα πλαίσια της παρούσας Διπλωματικής Εργασίας, υλοποιήθηκαν τα προγράμματα Converter (Μετατροπéας) και Compatibility Check (Ελεγκτής Συμβατότητας). Μέρος της Διπλωματικής αυτής ήταν να μελετηθεί κατά πόσο τα αποτελέσματα, που παράγουν τα εργαλεία, σχετικά με τις άδειες που περιέχονται σε διάφορα έργα ανοικτού λογισμικού είναι ορθά χωρίς να υπάρχουν προβλήματα συμβατότητας μεταξύ των αδειών αυτών. Για να επιτευχθεί ο στόχος αυτός αποφασίστηκε η υλοποίηση των δύο αυτών εργαλείων. Στο κεφάλαιο αυτό περιγράφονται τα δύο αυτά εργαλεία καθώς επίσης και η διαδικασία υλοποίησής τους.

Το κυρίως πρόγραμμα που έπρεπε να υλοποιηθεί είναι το Compatibility Check (Ελεγκτής Συμβατότητας) το οποίο θα μπορούσε να ανιχνεύει παραβιάσεις συμβατότητας μεταξύ των αδειών που περιέχονται στο αρχείο εισόδου. Το πρόγραμμα αυτό εντοπίζει παραβιάσεις συμβατότητας για άδειες που ανιχνεύονται από τα εργαλεία Fossology, Ohcount και Ninka. Τα τρία αυτά εργαλεία επιλέχθηκαν λόγω του ότι δεν παρέχουν τη δυνατότητα ανίχνευσης παραβιάσεων. Επίσης το εργαλείο BAT αν και δεν προσφέρει ούτε αυτό δυνατότητα ανίχνευσης παραβιάσεων δεν ήταν δυνατό να συμπεριληφθεί στα εργαλεία για τα οποία θα γινόταν η υλοποίηση των δύο προγραμμάτων λόγω των πολύ περιορισμένων αποτελεσμάτων

του, που οφείλονται στη μικρή βάση δεδομένων που χρησιμοποιήθηκε αλλά και λόγω του ότι τα αποτελέσματα του εργαλείου μπορούσαν να διαβαστούν μόνο μέσω της διεπαφής (interface) bat gui που προσφέρει το ίδιο το εργαλείο. Αντιθέτως τα εργαλεία OSLC και ASLA επειδή προσφέρουν δυνατότητα ανίχνευσης παραβιάσεων μεταξύ των αδειών δεν συμπεριλήφθηκαν για την υλοποίηση των προγραμμάτων. Αν και το ASLA δεν παρείχε ανίχνευση συμβατότητας σε όλα τα έργα λογισμικού ανοικτού κώδικα του συνόλου δεδομένων, εντούτοις δεν συμπεριλήφθηκε στην υλοποίηση γιατί τα αποτελέσματα του εμφανίζονται μόνο με τη χρήση διεπαφής χρήστη του εργαλείου.

Λόγω του ότι το κάθε εργαλείο βγάζει ένα αρχείο εξόδου με τις άδειες που ανιχνεύει, με μια δική του μορφή, υλοποιήθηκε το πρόγραμμα Converter (Μετατροπέας) το οποίο μετατρέπει και τα 3 αρχεία εξόδου σε μια γενική μορφή. Τα αρχεία που παράγονται από το πρόγραμμα Converter στη συνέχεια τα παίρνει σαν είσοδο το κυρίως πρόγραμμα Compatibility Check.

5.2 Εργαλείο Μετατροπέας (Converter)

5.2.1 Κίνητρο για υλοποίηση

Το κίνητρο για την υλοποίηση του εργαλείου αυτού ήταν ότι τα αρχεία εξόδου των τριών εργαλείων που έπρεπε να εξεταστούν για παραβιάσεις συμβατότητας είναι το καθένα σε διαφορετική μορφή. Όταν λέμε διαφορετική μορφή εννοούμε ότι κάθε εργαλείο για παράδειγμα έχει με διαφορετική μορφή αποθηκευμένα τα ονόματα των αδειών που ανιχνεύει. Η μορφή των αδειών του κάθε εργαλείου φαίνονται στον πίνακα 5.1 ένα μέρος για το Fossology (ολοκληρωμένος ο πίνακας βρίσκεται στο παράρτημα Α-2), 5.2 για το Ohcount και 5.3 για το Ninka. Επίσης το κάθε αρχείο εξόδου έχει διαφορετική διάταξη των αποτελεσμάτων, για παράδειγμα κάποιο εργαλείο εμφανίζει πρώτα το όνομα του αρχείου και μετά την άδεια την οποία περιέχει το αρχείο αυτό ενώ άλλο εργαλείο τα εμφανίζει με την ανάποδη σειρά. Παραδείγματα των αρχείων εξόδου του κάθε εργαλείου παρουσιάζονται και περιγράφονται στην επόμενη ενότητα. Έτσι λοιπόν δεν θα ήταν αποτελεσματικό το πρόγραμμα Compatibility Check (Ελεγκτής συμβατότητας) να παίρνει σαν είσοδο τρία διαφορετικά αρχεία και να τα μετατρέπει για να κάνει την ανίχνευση παραβιάσεων. Για το λόγο αυτό λοιπόν έπρεπε να υλοποιηθεί το εργαλείο Converter (Μετατροπέας) το οποίο θα μετέτρεπε τα τρία αυτά διαφορετικά αρχεία εξόδου σε μια γενική μορφή έτσι ώστε να μπορεί να το παίρνει σαν είσοδο το εργαλείο Compatibility Check (Ελεγκτής συμβατότητας) για να ψάξει για παραβιάσεις συμβατότητας. Στο σχήμα 5.4 φαίνεται η γενική λειτουργία και σύνδεση των δυο εργαλείων.

Όνομα άδειας				
3DFX	CopyLeft[1]	Intel	Novell	See-doc(OTHER)
3DFX-PL	CopyLeft[2]	Intel-EULA	Novell-IP	See-file(COPYING)
ACAA	CPAL	Intel-only-FW	Novell/SUSE	See-file(LICENSE)
ACDL	CPAL_v1.0	Intel(RESTRICTED)	NPL	See-file(README)
ACDL	CPL	Interbase-PL	NPL_v1.0	SGI_v2.0
Adaptec-GPL	CPL_v0.5	Interlink-EULA	NPL_v1.1	SGI-Freeware
Adaptec(RESTRICTED)	CPL_v1.0	IOS	NPL_v1.1+	SGI-Proprietary
Adaptive	CPOL	IoSoft(COMMERCIAL)	NPL-EULA	SGI-style
Adaptive_v1.0	CUA_v1.0	IP-claim	NRL	SGML
Adobe	CUPS	IPA	Nvidia	SISSL
Adobe-AFM	CUPS-EULA	IPA-Font-EULA	Nvidia-EULA	SISSL_v1.1
Adobe-EULA	Cylink-ISC	IPTC	O'Reilly	Skype(Non-commercial)
Adobe-SCLA	Cypress-FW	IronDoc	O'Reilly-style	Sleepycat
Affero	D.E.Knuth	ISC	OASIS	Sleepycat(Non-commercial)
Affero_v1	DARPA	Jabber	OCL	Sleepycat(Oracle)
Affero_v1+	DARPA-Cougaar	Jabber_v1.0	OCL_v1.0	SNIA
Affero_v3	Debian-social-DFSG	Java-Multi-Corp	OCL-style	SNIA_v1.0
Affero_v3+	Debian-SPI	Java-WSDL-Policy	OCLC	SNIA_v1.1
AFL	Debian-SPI-style	Java-WSDL-Schema	OCLC_v1.0	SpikeSource
AFL_v1.0	DMTF	Java-WSDL-Spec	OCLC_v2.0	SUN
AFL_v1.1	DOCBOOK	Java-WSDL4J	ODL	Sun
AFL_v1.2	DOCBOOK-style	JPEG/netpbm	OMF	Sun-BCLA
AFL_v2.0	DSL	JPNIC	OMRON(2) : OMRON	Sun-BCLA_v1.5.0
AFL_v2.1	Dyade	KD-Tools-EULA	Ontopia	Sun-EULA
AFL_v3.0	EBT	KDE	Open-PL	Sun-IP
AFPL-Ghostscript	EBT-style	Keyspan-FW	Open-PL_v1.0	Sun-Java
Against-DRM	ECL	Knuth-style	Open-Publication	Sun-PL

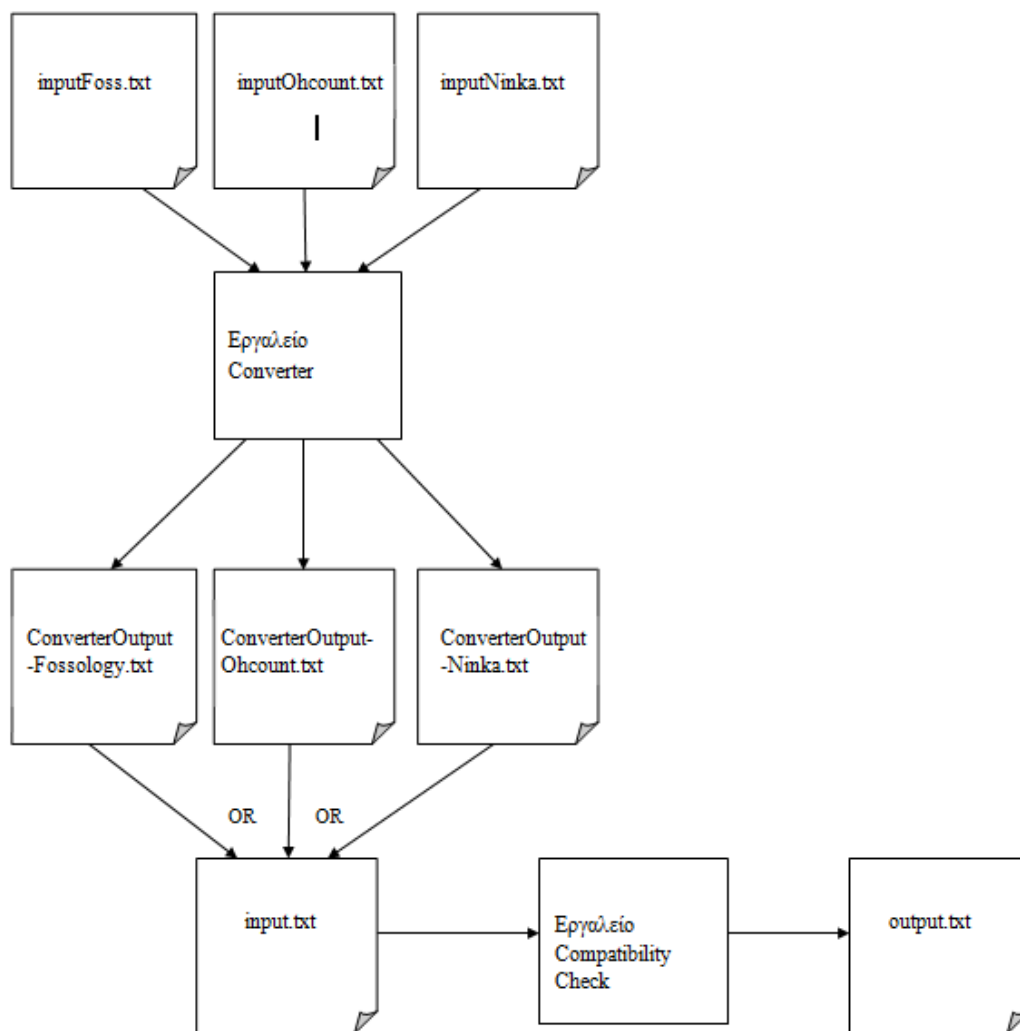
Σχήμα 5.1 Πίνακας αδειών εργαλείου Fossology

Όνομα άδειας		
academic	gpl3_or_later	python_license
adaptive	gpl3	python_software_foundation
affero	lgpl3	qt_public
apache	gpl	public_source
apache_2	lgpl	reciprocal_public
apple_open_source	historical	ricoh_source
artistic	i9_license	Sleepycat
attribution_assurance	ibm_public	sugarcrm_1_1_3
boost	intel_open_source	sun_industry_standards
bsd	jabber_open_source	sun_public
cecill	lucent_plan9	sybase_open_watcom
cecill_b	lucent_public	u_of_i_ncsa
cecill_c	mit	voidea_software
computer_associates_trusted	mitre	w3c
common_development_and_distribution	motosoto	wx_windows
common_public	mozilla_public_1	x_net
cua_office	mozilla_public_1_1	Zope
eu_datagrid	nasa_open	zlib_libpng
eclipse	naumen	apache_ish
educational	nethack	bsd_ish
eiffel	nokia_open_source	bsd_2clause_ish
eiffel_2	oclc_research	
entessa	open_group_test	
fair	open_software	
frameworkx	php_license	

Σχήμα 5.2 Πίνακας αδειών εργαλείου Ohcount

Όνομα άδειας		
AGPLv3+	GPLv3	MPLGPL2orLGPLv2_1
Apachev1.1	GPLv3+	MPL_LGPLsee
Apachev2	IBMV1	MPL-MIT-dual
artifex	intelBSDLicense	MPLv1_0
ArtisticLicensev1	InterACPILic	MPLv1_1
autoConfException	kerberos	MX4J
BeerWareVer42	LesserGPLnoVersion	MX4JLicensev1
BindMITX11Var	LesserGPLv2	NCSA
BisonException	LesserGPLv2+	NPLv1_0
boost	LesserGPLv2.1	NPLv1_1
boostV1	LesserGPLv2.1+	openssl
BSD1	LesserGPLv3	opensslvar1
BSD2	LesserGPLv3+	opensslvar2
BSD2AdvInsteadOfBinary	LGPLv2	opensslvar3
BSD2aie700	LGPLv2+	phpLicV3.01
BSD2EndorseInsteadOfBinary	LGPLv2_1	Postfix
BSD2SoftAndDoc	LGPLv2.1	Postgresql
BSD2var1	LGPLv2.1+	publicDomain
BSD2var2	LGPLv2_1orv3	QtGPLv2or3
BSD3	LGPLv2MISTAKE	Qtv1
BSD3NoWarranty	LGPLv2+MISTAKE	SameAsPerl
BSD4	LGPLv2orv3	SameTermsAs
BSD4NoEndor	LGPLv3	SeeFile
BSDCairoStyleWarr	LGPLv3+	sequenceLic
BSDdovecotStyle	LibGCLic	SimpleLic
BSDOnlyAdv	LibraryGPLv2	simpleLic
CDDLic	LibraryGPLv2+	simpleLic2
CDDLicV1	LinkException	simpleLicense1
CDDLv1orGPLv2	LinkExceptionBison	SimpleOnlyKeepCopyright
Cecill	LinkExceptionGPL	SleepyCat
ClassPathException	LinkExceptionLeGPL	SSLeay
CPLv0.5	LinkExceptionOpenSSL	Subversion
CPLv1	MITandGPL	subversion+
dovecotSeeCopying	MITCMU	subversionError
DoWhatTheFuckYouWantv2	MITCMUvar2	sunRPC
emacsLic	MITCMUvar3	SunSimpleLic
EPLv1	MITmodern	Svnkit
FreeType	MITold	svnkit+
GhostscriptGPL	MIToldMichiganVersion	tmate+
GPLnoVersion	MIToldwithoutSell	W3CLic
GPLv1	MIToldwithoutSellandNoDocumentation Requi	WxException
GPLv1+	MIToldwithoutSellCMUVariant	X11
GPLv1orArtistic	MITVariant	X11CMU
GPLv2	MITX11BSDvar	X11Festival
GPLv2+	MITX11noNotice	X11mit
GPLv2orLGPLv2.1	MITX11NoSellNoDocDocBSDvar	zendv2
GPLv2orv3	MITX11simple	ZLIB
GPLv2orv3qtException	MPL1_1andLGPLv2_1	ZLIBref

Σχήμα 5.3 Πίνακας αδειών εργαλείου Ninka



Σχήμα 5.4 Διάγραμμα λειτουργίας και σύνδεσης των δύο εργαλείων

Στο πιο πάνω σχήμα 5.4 περιγράφεται η λειτουργία των δύο εργαλείων καθώς επίσης και η σύνδεση των δύο εργαλείων. Όπως φαίνεται και στο σχήμα το εργαλείο Converter (Μετατροπέας) παίρνει σαν είσοδο τα τρία αρχεία εξόδου των εργαλείων, inputFoss.txt, inputOhcount.txt και inputNinka.txt και σαν έξοδο παράγονται τα τρία αυτά αρχεία σε γενική μορφή, ConverterOutput-Fossology.txt, ConverterOutput-Ohcount.txt και ConverterOutput-Ninka.txt. Στην συνέχεια επιλέγεται ένα από αυτά τα αρχεία για να δοθεί σαν είσοδος, input.txt, στο εργαλείο Compatibility Check (Ελεγκτής συμβατότητας) για να ανιχνευθούν τυχόν παραβιάσεις συμβατότητας στις άδειες που περιέχονται στο αρχείο αυτό. Τέλος αν ανιχνευθούν παραβιάσεις το εργαλείο Compatibility Check (Ελεγκτής συμβατότητας) παράγει σαν έξοδο το αρχείο output.txt που περιέχει τα αποτελέσματα του ελέγχου συμβατότητας των αδειών. Η λειτουργία των δυο εργαλείων περιγράφεται με μεγαλύτερη λεπτομέρεια στις επόμενες ενότητες.

5.2.2 Περιγραφή Εργαλείου Converter (Μετατροπέας)

Το εργαλείο Converter έχει σκοπό να μετατρέπει τα αρχεία εξόδου των εργαλείων Fossology, Ohcount και Ninka σε μια γενική μορφή και να παράγει σαν έξοδο το αρχείο αυτό με τη γενική μορφή. Στη συνέχεια το αρχείο εξόδου του εργαλείου αυτού θα δοθεί ως είσοδος στο εργαλείο Compatibility Check (Ελεγκτής συμβατότητας).

Τα τρία αρχεία εισόδου και εξόδου καταχωρούνται στο εργαλείο από το αρχείο Converter.properties και για το κάθε αρχείο εισόδου βγάζει και το αντίστοιχο αρχείο εξόδου με τη γενική μορφή. Ένα παράδειγμα του αρχείου Converter.properties φαίνεται πιο κάτω στο σχήμα 5.5.

Συγκεκριμένα για το εργαλείο Fossology με αρχείο εισόδου το inputFoss.txt παράγεται το αρχείο εξόδου ConverterOutput-Fossology.txt, για το εργαλείο Ohcount με αρχείο εισόδου το inputOhcount.txt παράγεται το αρχείο εξόδου ConverterOutput- Ohcount.txt και τέλος για το εργαλείο Ninka με αρχείο εισόδου το inputNinka.txt παράγεται το αρχείο εξόδου ConverterOutput- Ninka.txt.

```
inputNinka      = inputNinka.txt
inputOhcount    = inputOhcount.txt
inputFossology  = inputFoss.txt
outputNinka     = ConverterOutput-Ninka.txt
outputOhcount   = ConverterOutput-Ohcount.txt
outputFossology = ConverterOutput-Fossology.txt
```

Σχήμα 5.5 Παράδειγμα Αρχείου Converter.properties

Παραδείγματα των αρχείων εισόδου inputFoss.txt, inputOhcount.txt και inputNinka.txt παρουσιάζονται πιο κάτω στα σχήματα 5.6, 5.7 και 5.8 αντίστοιχα.

```
FileDlgOptionsHotKey.cpp contains license(s) GPL_v3+
FilePicture.cpp contains license(s) GPL_v3+
FileChildView.cpp contains license(s) GPL_v2+
FileDlgOptionsToolbar.h contains license(s) zlib/libpng
```

Σχήμα 5.6 Παράδειγμα Αρχείου inputFoss.txt

```
academic A Note.h
gpl DlgUserInformation.h
cecill DlgOptionsServer.cpp
cecill_b DlgDeleteInformation.h
gpl3 Translate.h
```

Σχήμα 5.7 Παράδειγμα Αρχείου inputOhcount.txt

```
./Source/_format_sources.bat;NONE
./Source/Translate.h;GPLv3+;;;0
./Source/Base64.cpp;GPLv3+;;;0
./Source/DlgOptionsProxy.cpp;GPLv3+;;;0
./Source/DlgLanguage.cpp;GPLv3+;;;0
```

Σχήμα 5.8 Παράδειγμα Αρχείου inputNinka.txt

Στα σχήματα 5.9, 5.10 και 5.11 φαίνονται τα αρχεία εξόδου που παράχθηκαν από το εργαλείο Converter (Μετατροπέας) με είσοδο τα αρχεία στα σχήματα 5.6, 5.7, 5.8. Η γενική μορφή των αρχείων εξόδου που παράγονται από το εργαλείο Converter (Μετατροπέας) είναι η εξής: <όνομα άδειας><κενό><όνομα αρχείου>

Για κάθε αρχείο που περιέχεται μέσα στο σύνολο δεδομένων παρουσιάζεται και το όνομα της άδειας που περιέχει. Δηλαδή στην αρχή κάθε γραμμής παρουσιάζεται το όνομα της άδειας και δίπλα του το όνομα του αρχείου που περιλαμβάνει την άδεια αυτή.

```
GPL_v3+ DlgOptionsHotKey.cpp
GPL_v3+ Picture.cpp
GPL_v2+ ChildView.cpp
zlib/libpng DlgOptionsToolbar.h
```

Σχήμα 5.9 Παράδειγμα Αρχείου ConverterOutput-Foss.txt

```
AFL A Note.h
GPL DlgUserInformation.h
CeCILL DlgOptionsServer.cpp
CeCILL-B DlgDeleteInformation.h
GPL_v3 Translate.h
```

Σχήμα 5.10 Παράδειγμα Αρχείου ConverterOutput-Ohcount.txt

```
GPL_v3+ Source/Translate.h
GPL_v3+ Source/Base64.cpp
GPL_v3+ Source/DlgOptionsProxy.cpp
GPL_v3+ Source/DlgLanguage.cpp
```

Σχήμα 5.11 Παράδειγμα Αρχείου ConverterOutput-Ninka.txt

Το εργαλείο Converter υποστηρίζει πολλές άδειες, δηλαδή μπορεί να αναγνωρίσει όλα τα ονόματα αδειών από τα αρχεία εξόδου των εργαλείων και να τα μετατρέψει στη γενική μορφή. Η λίστα με τις άδειες που υποστηρίζονται δημιουργήθηκε από την ένωση όλων των αδειών που μπορεί να αναγνωρίσει κάθε εργαλείο. Δηλαδή αν κάποια άδεια δεν υπάρχει στη λίστα κάποιου εργαλείου τότε η λίστα αυτή θα προστεθεί στις υποστηριζόμενες άδειες του Converter σε περίπτωση που το αρχείο εξόδου κάποιου εργαλείου περιέχει την άδεια αυτή.

Στον πίνακα 5.12 πιο κάτω παρουσιάζονται μερικές από τις άδειες που υποστηρίζει το εργαλείο Converter. Ολοκληρωμένος ο πίνακας με τις άδειες βρίσκεται στο παράρτημα A-2.

Οι άδειες του πίνακα που είναι σκιασμένες με γκριζό χρώμα είναι οι άδειες που περιέχονται μέσα στο γράφο συμβατότητας που χρησιμοποιεί το εργαλείο Compatibility Check.

Οι άδειες με κόκκινο χρώμα φόντου είναι οι άδειες που δεν περιλαμβάνονται στη λίστα αδειών του Fossology αλλά περιέχονται στη λίστα του εργαλείου Ninka.

Τέλος οι άδειες με μπλε χρώμα φόντου είναι οι άδειες που δεν περιλαμβάνονται στη λίστα αδειών του Fossology αλλά περιέχονται στη λίστα του εργαλείου Ohcount.

Όνομα Άδειας				
3DFX	Comtrol	InnerNet_v2.00	Not-Free!	SCSL_v3.0
3DFX-PL	Condor-PL	InnerNet-style	Not-OpenSource!	SCSL-TSA
ACAA	CopyLeft[1]	Intel	NotreDame	SCSL-TSA_v1.0
ACDL	CopyLeft[2]	Intel-EULA	NotreDame-style	See-doc(OTHER)
ACDL	CPAL	Intel-only-FW	Novell	See-file(COPYING)
Adaptec-GPL	CPAL_v1.0	Intel(RESTRICTED)	Novell-IP	See-file(LICENSE)
Adaptec(RESTRICTE D)	CPL	Interbase-PL	Novell/SUSE	See-file(README)
Adaptive	CPL_v0.5	Interlink-EULA	NPL	<i>sequenceLic</i>
Adaptive_v1.0	CPL_v1.0	IOS	NPL_v1.0	SGI_v2.0
Adobe	CPOL	IoSoft(COMMERCIA L)	NPL_v1.1	SGI-Freeware
Adobe-AFM	CUA_v1.0	IP-claim	NPL_v1.1+	SGI-Proprietary
Adobe-EULA	CUPS	IPA	NPL-EULA	SGI-style
Adobe-SCLA	CUPS-EULA	IPA-Font-EULA	NRL	SGML
Affero	Cylink-ISC	IPTC	Nvidia	<i>SimpleLic</i>
Affero_v1	Cypress-FW	IronDoc	Nvidia-EULA	<i>simpleLic</i>
Affero_v1+	D.E.Knuth	ISC	O'Reilly	<i>simpleLic2</i>
Affero_v3	DARPA	<i>i9_license</i>	O'Reilly-style	<i>simpleLicense1</i>
Affero_v3+	DARPA-Cougaar	Jabber	OASIS	<i>SimpleOnlyKeepCopyrig ht</i>
AFL	Debian-social-DFSG	Jabber_v1.0	OCL	SISSL
AFL_v1.0	Debian-SPI	Java-Multi-Corp	OCL_v1.0	SISSL_v1.1
AFL_v1.1	Debian-SPI-style	Java-WSDL-Policy	OCL-style	Skype(Non-commercial)

Πίνακας 5.12 Άδειες που υποστηρίζονται από το εργαλείο Converter

5.2.3 Υλοποίηση Εργαλείου Μετατροπής (Converter)

Το εργαλείο Converter αποτελείται από μία κλάση στην οποία είναι υλοποιημένο και όλο το εργαλείο. Η κλάση αυτή είναι η:

```
public class Converter {}
```

Το εργαλείο χωρίζεται σε διάφορες συναρτήσεις. Για το κάθε εργαλείο που το αρχείο εξόδου του δίνεται σαν είσοδο στο εργαλείο Converter υπάρχουν 2 συναρτήσεις. Η μια συνάρτηση διαβάζει το αρχείο και αποθηκεύει τις άδειες που υπάρχουν και τα αρχεία που τις περιέχουν και η δεύτερη συνάρτηση μετατρέπει τις άδειες αυτές σε γενική μορφή για να δημιουργηθεί στη συνέχεια το αρχείο εξόδου. Το εργαλείο Fossology δεν έχει συνάρτηση για μετατροπή των αδειών του σε γενική μορφή γιατί οι άδειες των υπόλοιπων εργαλείων μετατρέπονται στη μορφή των αδειών του Fossology επειδή είναι το εργαλείο που περιέχει τα περισσότερα πρότυπα αδειών.

Οι συναρτήσεις για διάβασμα του κάθε αρχείου εισόδου είναι οι εξής:

```
public static void Ninka() {}
```

Είναι η συνάρτηση που διαβάζει το αρχείο του εργαλείου Ninka.

```
public static void Fossology() {}
```

Είναι η συνάρτηση που διαβάζει το αρχείο του εργαλείου Fossology.

```
public static void Ohcount() {}
```

Είναι η συνάρτηση που διαβάζει το αρχείο του εργαλείου Ohcount.

Οι συναρτήσεις για μετατροπή των αδειών κάθε αρχείου εισόδου σε γενική μορφή είναι οι εξής:

```
public static void ConvertNinka(ArrayList<String> Licenses,  
ArrayList<String> Files)
```

Είναι η συνάρτηση που μετατρέπει τις άδειες του εργαλείου Ninka σε γενική μορφή.

```
public static void ConvertOhcount(ArrayList<String> Licenses,  
ArrayList<String> Files)
```

Είναι η συνάρτηση που μετατρέπει τις άδειες του εργαλείου Ohcount σε γενική μορφή.

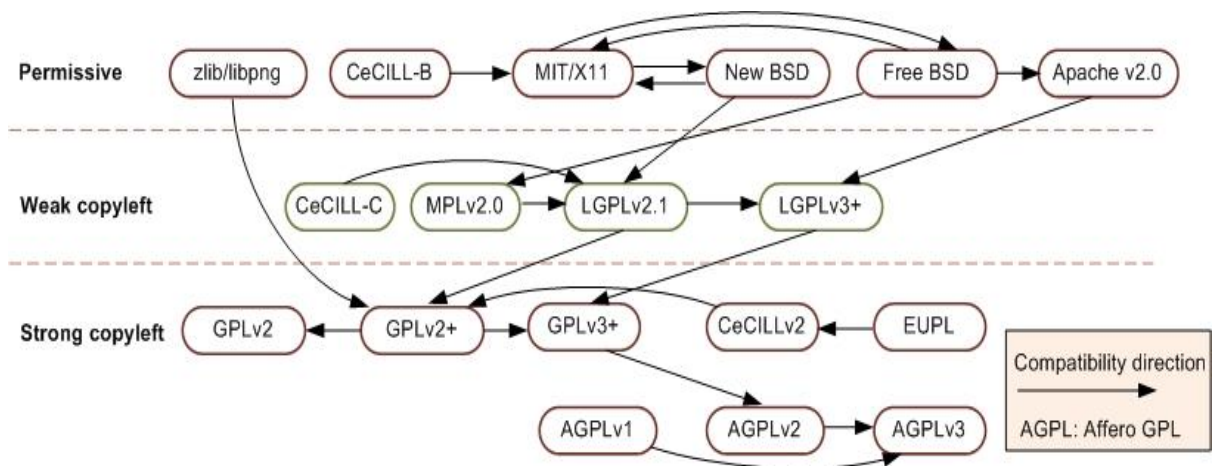
5.3 Εργαλείο Ελεγκτής συμβατότητας (Compatibility Check)

5.3.1 Κίνητρο για υλοποίηση

Τα περισσότερα από τα εργαλεία που έχουν μελετηθεί στην πειραματική αξιολόγηση δεν προσφέρουν τη δυνατότητα ανίχνευσης παραβιάσεων συμβατότητας ανάμεσα στις άδειες που έχουν αναγνωριστεί. Ο λόγος αυτός είναι που οδήγησε στην υλοποίηση του εργαλείου Compatibility Check. Το εργαλείο αυτό δίνει τη δυνατότητα στον χρήστη, αφού έχει αναγνωρίσει τις άδειες που περιέχονται σε κάποιο σύνολο δεδομένων με τη χρήση κάποιου από τα εργαλεία Fossology, Ohcount και Ninka, στην συνέχεια να πάρει τα αποτελέσματα αυτά και να δει αν υπάρχουν τυχόν παραβιάσεις συμβατότητας μεταξύ των αδειών που έχουν αναγνωριστεί. Είναι πολύ σημαντικό για κάποιον που χρησιμοποιεί ένα λογισμικό ανοικτού κώδικα, είτε για υλοποίηση ενός νέου λογισμικού είτε για απλή χρήση, να γνωρίζει αν οι άδειες που περιέχονται στο λογισμικό αυτό είναι συμβατές για να γνωρίζει τι προνόμια και δικαιώματα έχει πάνω στο λογισμικό αυτό. Θεωρήθηκε λοιπόν πολύ σημαντική η υλοποίηση του εργαλείου αυτού για να έχουν τη δυνατότητα οι χρήστες των τριών εργαλείων να ελέγχουν και τη συμβατότητα των αδειών του κάθε λογισμικού ανοικτού κώδικα που εξετάζουν.

5.3.2 Περιγραφή εργαλείου Ελεγκτής συμβατότητας (Compatibility Check)

Το εργαλείο Compatibility Check έχει ως στόχο την ανίχνευση παραβιάσεων συμβατότητας μεταξύ αδειών που περιέχονται στο αρχείο εισόδου. Ως είσοδο παίρνει τα αρχεία εξόδου που παράγονται από το εργαλείο Converter και ως έξοδο, αν ανιχνευτούν παραβιάσεις, δημιουργείται ένα αρχείο εξόδου που περιέχει τις άδειες που παραβιάζουν τη συμβατότητα και τα αρχεία που περιέχουν τις άδειες αυτές. Η συμβατότητα των αδειών προκύπτει από το γράφο συμβατότητας που φαίνεται στο σχήμα 5.13 [11]. Στο εργαλείο αυτό υλοποιήθηκε ο συγκεκριμένος γράφος και βάσει αυτού γίνεται ο έλεγχος για κάθε άδεια αν είναι συμβατή με τις υπόλοιπες άδειες που περιέχονται στο αρχείο εισόδου. Για τον έλεγχο συμβατότητας μεταξύ δύο αδειών κοιτάζουμε αν υπάρχει μονοπάτι στο γράφο μεταξύ των αδειών αυτών. Αν υπάρχει τότε είναι συμβατές, αλλιώς έχουμε πρόβλημα συμβατότητας.



Σχήμα 5.13 Γράφος συμβατότητας μεταξύ αδειών

Αν δεν ανιχνευτούν παραβιάσεις μεταξύ των αδειών τότε εμφανίζεται μήνυμα στο χρήστη ότι δεν υπήρξαν παραβιάσεις και δεν δημιουργείται αρχείο εξόδου. Ο καθορισμός του αρχείου εισόδου και εξόδου καθορίζεται στο αρχείο CompatibilityCheck.properties.

Στο σχήμα 5.14 φαίνεται ένα παράδειγμα του αρχείου CompatibilityCheck.properties.

```
input = input.txt
output = output.txt
```

Σχήμα 5.14 Παράδειγμα Αρχείου CompatibilityCheck.properties

Το αρχείο εισόδου που παίρνει το εργαλείο έχει την παρακάτω μορφή:

<όνομα άδειας><κενό><κενό><όνομα αρχείου>

Δηλαδή στην αρχή αναφέρεται το όνομα της άδειας, μετά δύο κενά και τέλος το όνομα του αρχείου που περιέχει την άδεια αυτή. Ένα παράδειγμα του αρχείου εισόδου παρουσιάζεται στο σχήμα 5.15 όπου περιέχει τις άδειες που περιέχει το κάθε αρχείο στο σύνολο των δεδομένων εισόδου.

```
GPL_v2 Source/DlgOptionsServer.cpp
GPL_v3+ Source/DlgDeleteInformation.h
GPL_v3+ Source/DlgLanguage.h
BSD3 Source/DlgOptionsFont.cpp
```

Σχήμα 5.15 Παράδειγμα Αρχείου Εισόδου για το εργαλείο CompatibilityCheck

Από το σχήμα 5.15 φαίνεται ότι οι άδειες που περιέχονται στο αρχείο εισόδου του εργαλείου CompatibilityCheck είναι οι GPL_v2, GPL_v3+ και BSD3. Το εργαλείο χρησιμοποιεί όπως αναφέραμε το γράφο συμβατότητας του σχήματος 5.13. Φαίνεται λοιπόν από το γράφο ότι δεν υπάρχουν ακμές που να δημιουργούν κάποιο μονοπάτι μεταξύ των αδειών GPL_v2 και GPL_v3+, αφού από την GPL_v2 δεν φεύγουν κάποιες ακμές για να δημιουργηθεί μονοπάτι προς την GPL_v3+ και από την GPL_v3+ οι ακμές πάνε μόνο προς τις άδειες AGPLv2 και μέσω αυτής στη συνέχεια στην AGPLv3. Άρα οι δύο αυτές άδειες δεν είναι συμβατές γιατί δεν υπάρχει μονοπάτι μεταξύ των δύο αδειών.

Για τις άδειες GPL_v2 και BSD3 (στο γράφο αναφέρεται ως New BSD) και GPL_v3+ και BSD3 φαίνεται από το γράφο ότι υπάρχει μονοπάτι μεταξύ των δύο αυτών περιπτώσεων των αδειών. Το μονοπάτι που δημιουργείται έχει ως εξής: αρχίζοντας από την άδεια New BSD, προχωρώντας στη συνέχεια στην άδεια LGPLv2.1 και μετά στην GPLv2+, μέσω της οποίας τέλος καταλήγουμε στις δύο άδειες GPL_v2 και GPL_v3+.

Από το γράφο λοιπόν φαίνεται ότι η μόνη παραβίαση στη συμβατότητα παρατηρείται στις άδειες GPL_v2 και GPL_v3+.

Ένα παράδειγμα του αρχείου εξόδου φαίνεται στο σχήμα 5.16 που περιέχει τις παραβιάσεις που ανιχνεύτηκαν, στο συγκεκριμένο παράδειγμα δηλαδή μεταξύ των αδειών GPL_v2 και GPL_v3+, και τα αρχεία που περιέχουν τις άδειες αυτές.

```
=====
===== Conflicts between licenses =====
=====

Conflict between license: GPL_v2 and license: GPL_v3+

=====
===== Files that contain the conflicted licenses =====
=====

The files that contain the GPL_v2 license are:
Source/DlgOptionsServer.cpp

The files that contain the GPL_v3+ license are:
Source/DlgDeleteInformation.h
Source/DlgLanguage.h
```

Σχήμα 5.16 Παράδειγμα Αρχείου Εξόδου για το εργαλείο CompatibilityCheck

5.3.3 Υλοποίηση Εργαλείου Ελεγκτής συμβατότητας (Compatibility Check)

Το εργαλείο αποτελείται από μια κλάση στην οποία είναι υλοποιημένο και όλο το εργαλείο. Η κλάση αυτή είναι η:

```
public class Compatibility {}
```

Αρχικά στη συνάρτηση main διαβάζεται το αρχείο εισόδου που περιέχει τις άδειες και τα αρχεία του συνόλου των δεδομένων εισόδου που περιέχουν τις άδειες αυτές. Οι άδειες και τα αρχεία αποθηκεύονται σε λίστες για να χρησιμοποιηθούν στη συνέχεια από τις υπόλοιπες συναρτήσεις. Έπειτα καλείται η συνάρτηση:

```
public static DirectedGraph<String, DefaultEdge> CreateGraph() {}
```

Η συνάρτηση αυτή δημιουργεί το γράφο συμβατότητας που χρησιμοποιείται από το εργαλείο για ανίχνευση των παραβιάσεων.

Μετά τη δημιουργία του γράφου καλείται η συνάρτηση:

```
public static void Compare(  
    ArrayList<String> Licenses, ArrayList<String> Files){}
```

Η συνάρτηση αυτή παίρνει σαν είσοδο τις λίστες με τις άδειες και τα αρχεία που περιέχονται στο αρχείο εισόδου και για κάθε συνδυασμό δύο αδειών ψάχνει να βρει αν υπάρχει μονοπάτι μεταξύ τους στο γράφο με τη χρήση της συνάρτησης:

```
path = DijkstraShortestPath.findPathBetween(graph, Licenses.get(i),  
Licenses.get(j));
```

Αν βρει μονοπάτι μεταξύ των δύο αδειών σημαίνει ότι είναι συμβατές, αν όχι τότε υπάρχει παραβίαση μεταξύ τους. Για κάθε συνδυασμό αδειών οι οποίες παραβιάζουν τη συμβατότητα του γράφου οι άδειες αυτές αποθηκεύονται σε μια λίστα με τις παραβιάσεις και τα αρχεία που τις περιέχουν σε μια άλλη λίστα με τα αρχεία. Τέλος αφού γίνει ο έλεγχος συμβατότητας για όλες τις άδειες του αρχείου εισόδου, οι άδειες με τις παραβιάσεις και τα αρχεία που τις περιέχουν τυπώνονται στο αρχείο εξόδου.

Κεφάλαιο 6

Συμπεράσματα

6.1 Γενικά Συμπεράσματα	54
6.2 Μελλοντική Εργασία	56

6.1 Γενικά Συμπεράσματα

Κάθε λογισμικό ανοικτού κώδικα καλύπτεται από μια άδεια η οποία καθορίζει τα δικαιώματα και τις υποχρεώσεις που έχει κάθε χρήστης του λογισμικού αυτού. Είναι λοιπόν πολύ σημαντικό οι χρήστες να γνωρίζουν την άδεια που περιέχεται μέσα στο έργο λογισμικού που θέλουν να χρησιμοποιήσουν. Λόγω του ότι όμως μερικά έργα λογισμικού δεν αναφέρουν ξεκάθαρα την άδεια η οποία τα καλύπτει ή δεν υπάρχει το σχετικό κείμενο της άδειας σε ξεχωριστό αρχείο, συχνά οι χρήστες δεν ξέρουν τι άδεια περιλαμβάνεται στο έργο αυτό. Για το λόγο αυτό είναι σημαντικό να υπάρχουν εργαλεία όπως αυτά που έχουν μελετηθεί και αξιολογηθεί στην παρούσα διπλωματική εργασία τα οποία μπορούν να εξάγουν την πληροφορία της άδειας μέσα από αρχεία πηγαίου κώδικα ή ακόμη και εκτελέσιμα αρχεία.

Από την πειραματική αξιολόγηση και στη συνέχεια τη σύγκριση των εργαλείων έχουν παραχθεί κάποια συμπεράσματα σχετικά με τη λειτουργία των εργαλείων. Αρχικά το εργαλείο με τα ορθότερα αποτελέσματα είναι το Fossology. Το μειονέκτημα όμως του εργαλείου αυτού είναι ότι επειδή ελέγχει όλα τα αρχεία που υπάρχουν μέσα σε ένα έργο λογισμικού, ανεξαρτήτως τύπου, ο χρόνος εκτέλεσης του είναι αρκετά μεγάλος και σε σύγκριση με τα υπόλοιπα εργαλεία είναι ο δεύτερος μεγαλύτερος. Άρα το εργαλείο αυτό είναι το καλύτερο για ανίχνευση αδειών αλλά είναι ένα από τα χειρότερα όσον αφορά το χρόνο εκτέλεσης. Μη λαμβάνοντας υπόψη τις εκδόσεις των αδειών στα αποτελέσματα, δεύτερο καλύτερο είναι το εργαλείο Ohcount το οποίο έχει τον δεύτερο μικρότερο χρόνο εκτέλεσης. Αν όμως οι εκδόσεις των αδειών λαμβάνονται υπόψη τότε έχουμε ως δεύτερο καλύτερο στα ορθά αποτελέσματα το εργαλείο Ninka το οποίο έχει τον μεγαλύτερο χρόνο εκτέλεσης από όλα τα εργαλεία. Τα υπόλοιπα εργαλεία είναι πολύ κοντά σε επιδόσεις σε σχέση και με τα αποτελέσματα αλλά και με το χρόνο εκτέλεσης. Παρατηρείται λοιπόν ότι τα

εργαλεία που προσφέρουν τα ορθότερα και πιο αξιόπιστα αποτελέσματα είναι τα εργαλεία με τα περισσότερα πρότυπα αδειών και τα εργαλεία αυτά έχουν επίσης μεγάλο χρόνο εκτέλεσης λόγω των πολλών αρχείων που ελέγχουν αλλά και των πολλών προτύπων που συγκρίνουν για να εξάγουν τα αποτελέσματα τους.

Εκτός από την ορθότητα και το χρόνο εκτέλεσης τα εργαλεία συγκρίθηκαν ως προς άλλες μετρικές στις οποίες τα 2 εργαλεία που αναφέρθηκαν πιο πάνω ως πιο αξιόπιστα υστερούν σε σχέση με μερικά από τα υπόλοιπα. Συγκεκριμένα τα εργαλεία Fossology και Ninka δεν προσφέρουν έλεγχο για παραβιάσεις στη συμβατότητα των αδειών που παρουσιάζονται στα αποτελέσματα, το οποίο είναι ένα σημαντικό μειονέκτημα σε σχέση με τα εργαλεία ASLA και OSLC τα οποία προσφέρουν τον έλεγχο αυτό. Επίσης το εργαλείο Ninka δεν προσφέρει καθόλου τη δυνατότητα παρουσίασης στατιστικών στοιχείων που αφορούν την ανάλυση ενώ το Fossology αν και παρουσιάζει κάποια στατιστικά εντούτοις υστερεί σε σχέση με τα εργαλεία ASLA και OSLC τα οποία παρουσιάζουν πολύ περισσότερα στατιστικά στο χρήστη. Τα στατιστικά στοιχεία που προσφέρει το κάθε εργαλείο έχουν παρουσιαστεί αναλυτικά στον πίνακα 4.15. Άρα μετά από αυτή τη σύγκριση και για άλλες μετρικές βλέπουμε ότι τα εργαλεία που είναι πιο αξιόπιστα παρέχουν λιγότερες δυνατότητες και πληροφορίες στο χρήστη.

Για το μειονέκτημα της μη παροχής ελέγχου συμβατότητας αδειών που αναφέρθηκε πιο πάνω, έγινε η υλοποίηση των 2 εργαλείων, Μετατροπέας (Converter) και Ελεγκτής Συμβατότητας (Compatibility Check), μέσω των οποίων προσφέρεται η δυνατότητα ανίχνευσης παραβιάσεων συμβατότητας για τα αποτελέσματα των εργαλείων Ninka, Fossology και Ohcount. Η υλοποίηση των 2 αυτών εργαλείων είναι αρκετά χρήσιμη αφού ο οποιοσδήποτε χρήστης των εργαλείων αυτών μπορεί αν θελήσει να ελέγξει για συμβατότητα των αδειών που περιέχονται σε ένα έργο λογισμικού ανοικτού κώδικα έτσι ώστε να αποφευχθούν τυχόν προβλήματα.

Μέσα από την παρούσα διπλωματική εξάχθηκαν σημαντικά αποτελέσματα και συμπεράσματα ως προς τη χρήση και λειτουργία των εργαλείων που μελετήθηκαν για ανίχνευση αδειών σε έργα λογισμικού ανοικτού κώδικα. Ακόμη υλοποιήθηκαν 2 προγράμματα τα οποία δίνουν στους χρήστες των εργαλείων Ninka, Fossology και Ohcount τη δυνατότητα ανίχνευσης παραβιάσεων, την οποία δεν είχαν πριν και μειονεκτούσαν σε σχέση με άλλα εργαλεία.

6.2 Μελλοντική Εργασία

Ως μελλοντικούς στόχους της συγκεκριμένη διπλωματικής εργασίας θα πρότεινα να γίνει αξιολόγηση περισσότερων εργαλείων ανίχνευσης αδειών σε έργα λογισμικού ανοικτού κώδικα. Μερικά εργαλεία δεν μπόρεσαν να συμπεριληφθούν στην έρευνα, όπως το LIDESC λόγω προβλημάτων που υπήρχαν στην εκτέλεση του εργαλείου και όπως το OSS Discovery το οποίο είναι ένα εργαλείο που δεν παρέχεται δωρεάν προς τους χρήστες.

Επίσης σχετικά με το πρόγραμμα Ελεγκτής Συμβατότητας (Compatibility Check) θα μπορούσαν να γίνουν κάποιες βελτιστοποιήσεις. Αρχικά θα μπορούσε να επεκταθεί ο γράφος συμβατότητας που χρησιμοποιήθηκε έτσι ώστε να μην περιέχει μόνο τις πιο γνωστές άδειες αλλά και άλλες άδειες που υποστηρίζονται από το εργαλείο Μετατροπέας (Converter). Με την εισαγωγή και άλλων αδειών στο γράφο πετυχαίνεται ανίχνευση παραβιάσεων μεταξύ περισσότερων αδειών που συνεπάγεται καλύτερα και πιο αξιόπιστα αποτελέσματα. Επίσης θα μπορούσε να απεικονιστεί οπτικά ο γράφος συμβατότητας για να μπορεί ο χρήστης να έχει άμεση γνώση του τι συμβαίνει στο πρόγραμμα και πως εξάγονται τα αποτελέσματα συμβατότητας.

Βιβλιογραφία

- [1] Wikipedia http://en.wikipedia.org/wiki/Open-source_software
- [2] Apache HTTP Server <http://httpd.apache.org/>
- [3] Linux <http://www.linux.org/>
- [4] Google Code <http://code.google.com/intl/el/>
- [5] Maven <http://mvnrepository.com/>
- [6] SourceForge <http://sourceforge.net/>
- [7] Πλεονεκτήματα ανοικτού λογισμικού
<http://open-source.gbdirect.co.uk/migration/benefit.html>
- [8] Πλεονεκτήματα ανοικτού λογισμικού
http://eu.conecta.it/paper/Advantages_open_source_soft.html
- [9] Andrew Charlesworth and Anna Home – Centre for IT & Law, University of Bristol Researchers, JISC Study to explore the legal and records management issues relating to the concept of the Lifelong Learner Record, Open Source Software Licensing: A Short FAQ http://www.jisc.ac.uk/uploaded_documents/Open_Source_FAQ.pdf
- [10] GPLv2 -GPLv3 Compatibility
<http://www.gnu.org/licenses/gplfaq.html#v2v3Compatibility>
- [11] D.A. Wheeler, “The Free-Libre / Open Source Software (FLOSS) License Slide,” <http://www.dwheeler.com/essays/floss-license-slide.pdf>, 2007.
- [12] Tuunamen, T., Koskinen, J. and Karkkaiken, T. Automated software license analysis. Automated Software Engineering.16, 3-4 (December 2009), 455-490.
- [13] Hemel, A., Trygve Kalleberg, K., Vermaas, R. and Dolstra, E. Finding software license violations through binary code clone detection. In Proceeding of the 8th working conference on Mining software repositories. ACM, New York, NY, (2011) 63-72.

- [14] Di Penta, M., German, D.M. and Antoniol, G. Identifying licensing of jar archives using a code-search approach. In Proceedings of the 7th IEEE Working Conference on Mining Software Repositories (May 2010), 151-160
- [15] Gobeille, R. The FOSSology project. In Proceedings of the 2008 International working conf. on Mining software repositories, ACM Press, New York, 2008, 47-50.
- [16] Ioannis E. Foukarakis, Georgia M. Kapitsaki, Nikolaos D. Tselikas Choosing licenses in free open source software, to appear in the 24th International Conference on Software Engineering and Knowledge Engineering (SEKE 2012), Redwood City, California, USA, July 1-3, 2012.
- [17] Fossology Project <http://www.fossology.org/projects/fossology/wiki>
- [18] Fossology Project http://www.linuxfoundation.org/sites/main/files/publications/lf_foss_compliance_fossology.pdf
- [19] D.M. German, Y. Manabe and K. Inoue. In Proceedings of the IEEE/ACM international Conference on Automated Software Engineering (ASE) 2010, pp: 437–446, A sentence-matching method for automatic license identification of source code files
- [20] Ohcount tool <http://www.ohloh.net/p/ohcount>
- [21] What-License? <http://www.what-license.com/>
- [22] Precision, Recall, F-mesure http://en.wikipedia.org/wiki/Precision_and_recall
- [23] OSLC readme
<http://www.soberit.hut.fi/T-76.4115/06-07/projects/groups/14/i2/readme.html>
- [24] ASLA readme <http://sourceforge.net/projects/asla/>

Παράρτημα Α

Α-1 Πίνακας με στοιχεία για το σύνολο των δεδομένων εισόδου

Έργο Λογισμικού	Περιγραφή	Αριθμός γραμών πηγαίου κώδικα	Γλώσσες προγραμματισμού	Πηγή	Μέγεθος (MB)
7zip	7zip is a file archiver with high compression ratio	165401	C, C++	SourceForge	5.1
Ajaxplorer-core-4.2.3	AjaXplorer is a software that can turn any web server into a powerfull file management system and an alternative to mainstream cloud storage providers.	209090	PHP, Javascript, HTML	SourceForge	14.7
qBittorrent	Multi-platform Bit Torrent client with Qt4 UI, Web UI and search engine	293238	C, C++, Python, Javascript	SourceForge	21.5
GNUCash	A personal and small-business finance manager	676429	C, C++, Python	SourceForge	72
Littlecmscolorengine	A color management library which implements fast transforms between ICC profiles	105610	C	SourceForge	7.7
MrBayes	A program for the Bayesian estimation of phylogeny	114356	C	SourceForge	6.3
A-note-source	A program that lets you to put notes on your desktop and those notes are uploaded in a database	27854	C, C++	SourceForge	6.4

GPUmat	A program which allows standard MATLAB code to run on GPUs	24967	C, C++	SourceForge	16.1
Q4wine	A qt4 GUI for WINE that helps in managing wine prefixes and installed applications	92788	C++	SourceForge	4.8
FreeImage	A library project to support popular graphics image formats	450238	C, C++, C#	SourceForge	18.4
JExcelApi	A java library which provides the ability to read, write and modify Excel spreadsheets	146114	C#, Java, HTML, Javascript	SourceForge	7.5
Logisim	An educational tool for designing and simulating digital logic circuits	143724	Java, HTML	SourceForge	13.3
gdata	The Google data Python client library which makes it easy to interact with Google services through the Google Data APIs	386029	Python, HTML	SourceForge	24.3
Mlpy	a Python module for Machine Learning built on top of NumPy/SciPy and of GSL	106558	Python, C	SourceForge	5.6
NumericalPython	Adds a fast and sophisticated array facility to Python	334130	Python, C, C++, Fortran	SourceForge	12.7
PythonFileFormat FileInterface	A python library for processing block structured binary files	134911	Python, C, C++, HTML, Javascript	SourceForge	7.9
VirtualDub	A program for video capture and processing	373694	Python, C, C++, Assembler	SourceForge	12.5

ffdshow	An advanced DirectShow filter and VFW codec which supports many audio and video formats	57097	Python, C, C++, Assembler, HTML	SourceForge	2
AMOS	collection of tools and class interfaces for the assembly of DNA reads	218000	, C++, Perl	SourceForge	8.9
Anomos	An anonymous, encrypted multi-P2P file sharing protocol, originally based on BitTorrent	17015	Python	SourceForge	2.1
AresGalaxy	Filesharing-Bittorrent p2p client connected to TCP supernode/leaf network and UDP DHT network	156870	Pascal	SourceForge	5.5
Artha~TheOpenThesaurus	A handy thesaurus based on WordNet with distinct features like global hotkey look-up, passive desktop notifications, regular expression based search, etc	46735	C	SourceForge	1.8
AWStats	A free powerful and featureful server logfile analyzer that shows you all your Web/Mail/FTP statistics including visits, unique visitors, pages, hits, rush hours, os, browsers, search engines, keywords, robots visits, broken links and more	56051	Perl, HTML ,Javascript	SourceForge	3.7
Boats	a race scenario drawing tool	21171	C,C++	SourceForge	0.907
Cerebrum	A Python/RDBMS framework upon which user administration systems can be built	69863	Python	SourceForge	3.3
Cheetah	A template engine and code generation tool, written in Python Web development is its principle use, but Cheetah is very flexible and is also being used to generate C++ game code, Java, sql, form emails and even Python code	25421	Python, C	SourceForge	0.974

ChronosXP	A planetary hours astrology program for Windows It runs from the system tray where it displays a glyph corresponding to the current planetary hour	33728	C#,HTML	SourceForge	3
Comical	A portable CBR and CBZ viewer	29972	C, C++	SourceForge	1.7
ChickenoftheVNC	A VNC client for Mac OS X A VNC client allows one to display and interact with a remote computer screen	20395	C, C++	SourceForge	1.7
CuteFlow	A webbased document circulation/workflow tool Users are able to define "documents" which are send step by step to every user in a list	89311	PHP, HTML, Javascript	SourceForge	8.4
Cutter	A Unit Testing Framework for C and C++	324985	C, C++, HTML, Ruby	SourceForge	16.2
DirectorySynchronizePro(DirSyncPro)	A configure tool to synchronize the content of a directory to another	732	HTML	SourceForge	4.5
Docutils	Utilities for general- and special-purpose documentation, including autodocumentation of Python modules	91926	Python, HTML, Javascript	SourceForge	5.3
DrPython	A highly customizable cross-platform ide to aid programming in Python	26826	Python	SourceForge	1.7
DvdFlick	A tool to convert various PC video formats to a DVD that can be played on pretty much any standalone DVD player	22200	VisualBasic	SourceForge	3.8
ExpatXMLParser	A stream oriented parser that requires setting handlers to deal with the structure that the parser discovers in the document	71118	C,C++	SourceForge	2.5

FAAD	the decoder of FAAC(Freeware Advanced Audio Coder	129734	C, C++	SourceForge	4.9
FckEditor	A browser-based WYSIWYG text editor, which brings to the Web common editing features found in desktop word processors	87193	HTML, Javascript, Python, PHP	SourceForge	4.2
Flex	A tool for generating scanners programs which recognize lexical patterns in text	19480	C, C++	SourceForge	0.906
Fraqtive	An open source, multi- platform generator of the Mandelbrot family fractals	77317	C, C++	SourceForge	5.8
Gammu	A cellular manager for mobile phones/modems It contains libraries and functions for ringtones,logos,phonebook, SMS,etc	197149	C, C++, Python, PHP, Perl	SourceForge	41.5
GMic	This project aims at defining a specialized script language dedicated to image processing, and provide various user interfaces to use it	293049	C, C++	SourceForge	15.1
GNS3	A graphical network simulator that allows you to design complex network topologies	366322	Python	SourceForge	23.5
GoldenDict	A feature-rich dictionary lookup program, supporting multiple dictionaries' formats, perfect article rendering with the complete markup, illustrations and other content retained, and allowing to type in words without any accents or correct case	85971	C, C++, HTML	SourceForge	8.1
Grisbi	very functional personal financial management program with a lot of features	232706	C	SourceForge	13.3

Hunspell	A spell checker and morphological analyzer library and program designed for languages with rich morphology and complex compounding or character encoding	1459	C++	SourceForge	0.07
Hydrogen	An advanced drum machine for GNU/Linux, Windows and Mac OS X	126105	C++, C, Python, HTML	SourceForge	13.9
Handbrake	An open-source, GPL-licensed, multiplatform, multithreaded video transcoder,	238686	C#, C, Python, C++	SourceForge	15.9
HyperSQLDatabaseEngine(HSQLDB)	a relational database engine written in Java, with a JDBC driver, conforming to ANSI SQL2008	481722	Java, html, javascript	SourceForge	24.5
IOMeter	I/O performance Analysis Tool	58517	C++, C	SourceForge	6
ITalc	A free classroom management software which enables teachers to view and control computers in their labs and interact with students	311387	C++, C	SourceForge	15.1
Icinga	is an enterprise grade open source monitoring system which keeps watch over networks and any conceivable network resource, notifies the user of errors and recoveries and generates performance data for reporting	333756	HTLM, C, Perl, Javascript	SourceForge	21
JACOB	A JAVA-COM Bridge that allows you to call COM Automation components from Java It uses JNI to make native calls to the COM libraries	33307	HTML, Javascript	SourceForge	: 2.2
JFreeChart	A free (LGPL) chart library for the Java(tm) platform	322822	HTML, Java	SourceForge	17

Joda-Time	A library of classes to replace the Java JDK Date and Time classes including formatting	146598	Java	SourceForge	9.4
JPDF-Tweak	A Swiss Army Knife GUI application for PDF documents combine, split, rotate, reorder (n-up, booklet), watermark, edit bookmarks/fileinfo/pagetransition, compress, encrypt, decrypt, sign, repair, edit attachments and more	306	: HTLM	SourceForge	2.4
Jsmooth	standard Windows executable files (exe) that smartly launch java applications	37579	Java, C, C++	SourceForge	5.1
Kdiff3	A graphical text difference analyzer for up to 3 input files, provides character-by-character analysis and a text merge tool with integrated editor	63761	C, C++,HTML	SourceForge	7
KGBArchiver	A compression tool with unbelievable high compression rate	75182	C, C++	SourceForge	14.5
LibUsb	library to enable user space application programs to communicate with USB devices	34881	C	SourceForge	1.3
LNGoogleCalSync	An easily deployable, web based bugtracker to aid product bug tracking It requires PHP, MySQL and a web server	4231	HTML, VisualBasic	SourceForge	2.9
MantisBt	Python implementation of Markdown	231869	PHP, HTML, Javascript	SourceForge	13.9
Markdown	A convenient unified display of the most relevant technical and tag data for video and audio files	4667	Python	SourceForge	0.243

MediaInfo	A program that resembles a traditional flash-card program but with an important twist it uses a sophisticated algorithm to schedule the best time for a card to come up for review	314891	C, C++	SourceForge	13
Mnemosyn	A Flex-based web application for searching, browsing, and playing your mp3 collection remotely	48224	Python, Javascript, HTML	SourceForge	3.4
Mp3dj	A system for multi-site booking of meeting rooms Rooms are grouped by building/area and shown in a side-by-side view	17910	Javascript, HTML	SourceForge	5.2
Mrbs	Low-latency, high-quality voice communication for gamers Includes game linking, so voice from other players comes from the direction of their characters, and has echo cancellation so the sound from your loudspeakers won't be audible to other players	62068	Javascript, HTML, PHP	SourceForge	4.6
Mumble	A modular system for the rapid whole genome alignment of finished or draft sequence	363434	C++, C	SourceForge	17.5
MumMer	A MySQL database connector for Python programming	41778	C++, C	SourceForge	6.5
MySQLforpython	powerful, enterprise-class host, service, application, and network monitoring program	7445	Python, C	SourceForge	0.289
Nagios	visualization addon for the well known network managment system Nagios	178066	C, PHP, Perl	SourceForge	7.9
Nagvis	provides tools and libraries relating to the Simple Network Management Protocol	94039	HTML, Javascript, PHP	SourceForge	7.6

Net-snmp	An automated library system written in PHP containing OPAC, circulation, cataloging, and staff administration functionality	541104	C, Perl, C++, Python	SourceForge	7.6
OpenBiblio	A Simple CSV Parser for Java	46815	PHP, HTML	SourceForge	3.2
OpenCSV	A web based point of sale system written in the PHP language	16606	Java, HTML, Javascript	SourceForge	0.907
OpenSourcePoint ofSale	An USB programmer for various devices, including PIC, dsPIC, AVR microcontrollers, and various kinds of serial eeproms	76542	PHP	SourceForge	2.8
OpenProgrammer	Wire file system	21478	C, C++	SourceForge	0.704
Owfs	Floating point library for PIC micro controllers	178418	C, C++, PHP, Python, Perl, HTML	SourceForge	6.2
PicFloat	A serial device programmer based on some simple and cheap interfaces for the PC and a user friendly GUI	15850	C, Assembler	SourceForge	0.845
PonyProg:serialdeviceprogrammer	a Web Based Management tool created for Postfix	98524	C, C++	SourceForge	4.2
PostFixAdmin	a satellite tracking software for observing purposes	68755	PHP, HTML, Perl	SourceForge	4.9
Previsat	a non-caching web proxy with advanced filtering capabilities for enhancing privacy, modifying web page data and HTTP headers, controlling access, and removing ads and other	44815	C, C++	SourceForge	4.1

	obnoxious Internet junk				
Privoxy	A distutils extension to create standalone windows programs from python scripts	95274	C, HTML, Perl	SourceForge	5.7
Py2Exe	Python bindings for Digia's Qt cross-platform application development framework	10990	Python, C	SourceForge	0.486
PyQt	UI for controlling / managing a sabnzdb instance from a Windows Mobile device	316638	HTML, C++, C	SourceForge	17.4
SABnzdb	A Geographic Information System (GIS) software with immense capabilities for geodata processing and analysis	69286	HTML, Javascript, Python	SourceForge	7.2
SagaGIS	The Swiss Army Knife of sound processing utilities	586426	C, C++	SourceForge	22.2
SciPy:ScientificLibraryforPython	a free open source, retargettable, optimizing ANSI C	859477	C, C++, Python, Fortran	SourceForge	35
SmallDeviceCompilerSuite	compiler suite that supports a growing list of processors	532531	C, HTML, Assembler	SourceForge	72.8
JellyMark	next-generation OpenGL and CPU benchmarking program	120654	C, C++	SourceForge	4.9
StreamRipper	records shoutcast-compatible streams	89200	C	SourceForge	3.9

UfRaw	A tool for reading and manipulating raw images from digital cameras	56342	C, C++	SourceForge	4.5
TightVNC	improved version of VNC, great free remote-desktop tool	160734	C, C++	SourceForge	7.6
WikidPad	A wiki-like notebook for storing your thoughts, ideas, todo lists, contacts, or anything else you can think of to write down	131894	Python	SourceForge	6.4
WebMin	A web-based system administration tool for Unix servers and services	91865	Perl, HTML, Javascript	SourceForge	16.2
Xastir	Real-time tracking & messaging of stations via radio/internet APRS data streams, w/125 map formats supported	214783	C, Perl	SourceForge	11.3
YaoqiangBPMNEditor	a graphical editor for business process diagrams , compliant with OMG specifications (BPMN 20)	20907	Java	SourceForge	6.9
ZintBarcodeGenerator	A barcode encoding library supporting over 50 symbologies	54897	C, C++	SourceForge	1.9
cocoon-licenses1.0.0-M1	A project with txt files that contain text of different licenses		Text	Maven	0.959
licenses-7-sources	A project with txt files that contain text of different licenses		Text	Maven	0.09

A-2 Πίνακας με τις άδειες του εργαλείου Fossology

3DFX	CopyLeft[1]	Intel	Novell	See-doc(OTHER)
3DFX-PL	CopyLeft[2]	Intel-EULA	Novell-IP	See-file(COPYING)
ACAA	CPAL	Intel-only-FW	Novell/SUSE	See-file(LICENSE)
ACDL	CPAL_v1.0	Intel(RESTRICTED)	NPL	See-file(README)
ACDL	CPL	Interbase-PL	NPL_v1.0	SGI_v2.0
Adaptec-GPL	CPL_v0.5	Interlink-EULA	NPL_v1.1	SGI-Freeware
Adaptec(RESTRICTED)	CPL_v1.0	IOS	NPL_v1.1+	SGI-Proprietary
Adaptive	CPOL	IoSoft(COMMERCIAL)	NPL-EULA	SGI-style
Adaptive_v1.0	CUA_v1.0	IP-claim	NRL	SGML
Adobe	CUPS	IPA	Nvidia	SISSL
Adobe-AFM	CUPS-EULA	IPA-Font-EULA	Nvidia-EULA	SISSL_v1.1
Adobe-EULA	Cylink-ISC	IPTC	O'Reilly	Skype(Non-commercial)
Adobe-SCLA	Cypress-FW	IronDoc	O'Reilly-style	Sleepycat
Affero	D.E.Knuth	ISC	OASIS	Sleepycat(Non-commercial)
Affero_v1	DARPA	Jabber	OCL	Sleepycat(Oracle)
Affero_v1+	DARPA-Cougaar	Jabber_v1.0	OCL_v1.0	SNIA
Affero_v3	Debian-social-DFSG	Java-Multi-Corp	OCL-style	SNIA_v1.0
Affero_v3+	Debian-SPI	Java-WSDL-Policy	OCLC	SNIA_v1.1
AFL	Debian-SPI-style	Java-WSDL-Schema	OCLC_v1.0	SpikeSource
AFL_v1.0	DMTF	Java-WSDL-Spec	OCLC_v2.0	SUN
AFL_v1.1	DOCBOOK	Java-WSDL4J	ODL	Sun
AFL_v1.2	DOCBOOK-style	JPEG/netpbm	OMF	Sun-BCLA
AFL_v2.0	DSL	JPNIC	OMRON(2) : OMRON	Sun-BCLA_v1.5.0
AFL_v2.1	Dyade	KD-Tools-EULA	Ontopia	Sun-EULA
AFL_v3.0	EBT	KDE	Open-PL	Sun-IP
AFPL-Ghostscript	EBT-style	Keyspan-FW	Open-PL_v1.0	Sun-Java
Against-DRM	ECL	Knuth-style	Open-Publication	Sun-PL

AGE-Logic	Eclipse	Lachman-Proprietary	Open-Publication_v1.0	Sun-PL_v1.0
Agere-EULA	Eclipse_v1.0	Larabie-EULA	Open-Publication-style	Sun-Proprietary
AGFA-EULA	eGenix	LDP	OpenGroup	Sun-RPC
AGFA(RESTRICTED)	Eiffel	LDP_v1A	OpenGroup-PL	Sun-SCA
Aladdin-Ghostscript	Eiffel_v1	LDP_v2.0	OpenGroup-Proprietary	Sun(Non-commercial)
Aladdin(Closed-Source!)	Eiffel_v2	Legato	OpenGroup-style	Sun(RESTRICTED)
Aladdin(RESTRICTED)	Entessa	LGPL	OpenGroupTest	Sun(tm)
Alfresco	Epinions	LGPL_v1	OpenLDAP	SunPro
Alfresco/FLOSS	Epson-EULA	LGPL_v1+	OpenLDAP_v1.2	Sybase_v1.1
Algorithmics	Epson-PL	LGPL_v2	OpenLDAP_v1.4	Tektronix
AMD	EU-DataGrid	LGPL_v2.1	OpenLDAP_v2.5	Tektronix-style
AMD-EULA	Fair	LGPL_v2.1+	OpenLDAP_v2.6	TeX-exception
AOL-EULA	Fair-style	LGPL_v2+	OpenLDAP_v2.7	Trident-EULA
Apache	Fedora	LGPL_v3	OpenLDAP_v2.8	TrollTech
Apache_v1.0	Fedora-CLA	LGPL_v3?	OpenLDAP-style	U-BC
Apache_v1.1	Flash2xml_v1.0	LGPL_v3+	OpenMarket	U-Cambridge
Apache_v2.0	Frameworkx	Link-exception	OpenSSL	U-Cambridge-style
Apache-style	Frameworkx_v1.0	Linux-HOWTO	Combined OpenSSL+SSLeay	U-Chicago
Apple	Free-Art_v1.0	LinuxDoc	Oracle-Dev	U-Columbia
Apple-EULA	Free-Art_v1.3	LPPL	Oracle-EULA	U-Del
Apple(FontForge)	Free-PL	LPPL_v1.0	OSD	U-Del-style
Apple(Sample)	Free-SW	LPPL_v1.0+	OSF	U-Edinburgh
APSL	Free-SW(run-COMMAND)	LPPL_v1.1	OSF-style	U-Edinburgh-style
APSL_v1.0	FreeBSD-Doc	LPPL_v1.1+	OSL	U-Mich-style
APSL_v1.1	Freetype	LPPL_v1.2	OSL_v1.0	U-Michigan
APSL_v1.2	Freetype-style	LPPL_v1.2+	OSL_v1.1	U-Monash
APSL_v2.0	FSF	LPPL_v1.3	OSL_v2.0	U-Utah
Aptana	Fujitsu	LPPL_v1.3+	OSL_v2.1	U-Wash-style
Aptana-PL_v1.0	Garmin-EULA	LPPL_v1.3a	OSL_v3.0	U-Wash(Free-Fork)

ARJ	Genivia(Commercial)	LPPL_v1.3a+	Paradigm	U-Washington
Arphic-Font-PL	GFDL	LPPL_v1.3b	Phorum	UCAR
Arphic-style	GFDL_v1.1	LPPL_v1.3b+	PHP	UCAR-style
Artifex	GFDL_v1.1+	LPPL_v1.3c	PHP_v2.0	UCWare-EULA
Artistic	GFDL_v1.2	LPPL_v1.3c+	PHP_v2.0.2	Unicode
Artistic_v2.0	GFDL_v1.2+	Lucent_v1.0	PHP_v3.0	UnitedLinux-EULA
Ascender-EULA	Ghostscript-GPL	Lucent_v1.02	PHP_v3.01	Unix-Intl
ATI-EULA	Ghostscript-GPL_v1.1	M-Plus-Project	PHP-style	USC
ATMEL-FW	GNU-copyleft	MacroMedia-RPSL	Piriform	USC-style
ATT	GNU-Ghostscript	Macrovision	Pixware-EULA	USC(Non-commercial)
ATT-Source	GNU-Manpages	Macrovision-EULA	Platform-Computing(RESTRICTED)	USL-Europe
ATT-Source_v1.0	GNU-style(EXECUTE)	Majordomo	Polyserve-CONFIDENTIAL	Vim
ATT-Source_v1.2d	GNU-style(interactive)	Majordomo_v1.1	Princeton	Vixie
ATT-style	Google	Mandriva	Princeton-style	VMware-EULA
ATT(Non-commercial)	GPDL	MellanoX	Proprietary!	Vovida
Attribution-Assurance	GPL	MetroLink	Public-Use	W3C
AVM	GPL_v1	MetroLink-nonfree	Public-Use_v1.0	W3C-IP
Baekmuk-Font	GPL_v1+	Mibble	Python	W3C-style
Baekmuk(Hwan)	GPL_v2	Mibble_v2.8	Python_v2	Wash-U-StLouis
BancTec	GPL_v2:v3	Microsoft	Python_v2.0.1	Wash-U-style
BEA	GPL_v2.1	MindTerm	Python_v2.0.2	WebM
Bellcore	GPL_v2.1+	MIT	Python_v2.1.1	Wintertree
Bellcore-style	GPL_v2+	MIT-style	Python_v2.1.3	WTI(Not-free)
BeerWareVer42	GPLv2+KDEupgradeClause	MITEM	Python_v2.2	wxWindows-LGPL
BindMITX11Var	GPL_v3	Mitre	Python_v2.2.3	X/Open
BH-Font	GPL_v3?	MitreCVW	Python_v2.2.7	X/Open-style
BH-Font-style	GPL_v3+	MitreCVW-style	Python_v2.3	X11
BISON	GPL-classpath-exception	Motorola	Python_v2.3.7	X11-style

Bitstream	GPL-exception	Motosoto_v0.9.1	Python_v2.4.4	Xerox
BitTorrent	GPL-Meta	MPEG3-decoder	Python-style	Xerox-style
BitTorrent_v1.0	GPL-Bison-exception	MPL	QPL	Ximian
BitTorrent_v1.1	GPL-or-LGPL	MPL_v1.0	QPL_v1.0	Ximian_v1.0
BIZNET	GPL(rms)	MPL_v1.1	Qt(Commercial)	XOPEN-EULA
BIZNET-style	gSOAP	MPL_v1.1+	QuarterDeck	Yahoo
Boost	gSOAP_v1.3	MPL_v2.0	Quest-EULA	Yahoo-EULA
Boost_v1.0	Hacktivismo	MPL-EULA_v1.1	RCSL	YaST(SuSE)
Boost-style	Hauppauge	MPL-EULA_v2.0	RCSL_v1.0	Zend_v2.0
BrainStorm-EULA	Helix/RealNetworks-EULA	MPL-EULA_v3.0	RCSL_v2.0	Zeus
Broadcom-EULA	Historical	MPL-style	RCSL_v3.0	zlib/libpng
BSD	HP	MPL/TPL	RealNetworks-EULA	ZoneAlarm-EULA
CCA-ND_v3.0	HP-Compaq	MPL/TPL_v1.0	RealNetworks-Unknown	Zope
CCA_v1.0	HP-DEC	MRL	RedHat	Zope-PL_v1.0
CCA_v2.5	HP-DEC-style	Ms-EULA	RedHat-EULA	Zope-PL_v2.0
CCA_v3.0	HP-EULA	Ms-indemnity	RedHat-specific	Zveno
CCA-SA	HP-Proprietary	Ms-IP	RedHat(Non-commercial)	
CCA-SA_v1.0	HP-style	Ms-LPL	Redland	
CCA-SA_v2.5	HP+IBM	Ms-LRL	Ricoh	
CCA-SA_v3.0	IBM	Ms-PL	Ricoh_v1.0	
CCPL	IBM-Courier	Ms-RL	Riverbank-EULA	
CCPL_v2.0	IBM-EULA	Ms-SSL	RPL	
CCPL_v2.5	IBM-JCL	MySQL_v0.3	RPL_v1.0	
CCPL_v3.0	IBM-PL	MySQL-style	RPL_v1.1	
CDDL	IBM-PL_v1.0	MySQL/FLOSS	RPL_v1.5	
CDDL_v1.0	IBM-PL_v2.0	NASA	RPSL	
CeCILL	IBM-reciprocal	NASA_v1.3	RPSL_v1.0	
CeCILL_v1	ID-EULA	Naumen	RPSL_v2.0	
CeCILL_v1.1	IDPL	nCipher	RPSL_v3.0	

CeCILL_v2	IDPL_v1.0	NESSUS-EULA	RSA-DNS	
CeCILL_v2.0	IEEE-Doc	NetBSD	RSA-Security	
CeCILL-B	IETF	Nethack	Ruby	
CeCILL-C	IETF-style	Netizen	Same-license-as	
Cisco	ImageMagick	Nokia	SciTech	
Cisco-style	ImageMagick-style	Nokia_v1.0a	SCO(commercial)	
CMU	ImageMagick(Apache)	Non-commercial!	SCSL	
CMU-style	InfoSeek	Not-Free!	SCSL_v2.3	
CompuServe	InnerNet	Not-OpenSource!	SCSL_v3.0	
Comtrol	InnerNet_v2.00	NotreDame	SCSL-TSA	
Condor-PL	InnerNet-style	NotreDame-style	SCSL-TSA_v1.0	

A-3 Πίνακας με τις μετρικές που χρησιμοποιήθηκαν στη σύγκριση

	Μετρικές	Μονάδα Μέτρησης
1	Χρόνος Εκτέλεσης	Δευτερόλεπτα (seconds)
2	Σωστά αποτελέσματα (ακρίβεια (precision), ανάκληση (recall), f-measure)	Ποσοστά (percentages)
3	Στατιστικά στοιχεία	Ναι/Όχι
4	Αριθμός υποστηριζόμενων προγραμματιστικών γλωσσών	Ακέραιος αριθμός
5	Αναφορά προβλημάτων	Ναι/Όχι
6	Παρουσίαση άδειας κάθε αρχείου	Ναι/Όχι
7	Αριθμός υποστηριζόμενων αδειών	Ακέραιος αριθμός
8	Ανίχνευση παραβιάσεων συμβατότητας μεταξύ των αδειών	Ναι/Όχι

A-4 Πίνακας με τις υποστηριζόμενες άδειες του εργαλείου Converter

Όνομα Άδειας				
3DFX	Comtrol	InnerNet_v2.00	Not-Free!	SCSL_v3.0
3DFX-PL	Condor-PL	InnerNet-style	Not-OpenSource!	SCSL-TSA
ACAA	CopyLeft[1]	Intel	NotreDame	SCSL-TSA_v1.0
ACDL	CopyLeft[2]	Intel-EULA	NotreDame-style	See-doc(OTHER)
ACDL	CPAL	Intel-only-FW	Novell	See-file(COPYING)
Adaptec-GPL	CPAL_v1.0	Intel(RESTRICTED)	Novell-IP	See-file(LICENSE)
Adaptec(RESTRICTED)	CPL	Interbase-PL	Novell/SUSE	See-file(README)
Adaptive	CPL_v0.5	Interlink-EULA	NPL	<i>sequenceLic</i>
Adaptive_v1.0	CPL_v1.0	IOS	NPL_v1.0	SGI_v2.0
Adobe	CPOL	IoSoft(COMMERCIAL)	NPL_v1.1	SGI-Freeware
Adobe-AFM	CUA_v1.0	IP-claim	NPL_v1.1+	SGI-Proprietary
Adobe-EULA	CUPS	IPA	NPL-EULA	SGI-style
Adobe-SCLA	CUPS-EULA	IPA-Font-EULA	NRL	SGML
Affero	Cylink-ISC	IPTC	Nvidia	<i>SimpleLic</i>
Affero_v1	Cypress-FW	IronDoc	Nvidia-EULA	<i>simpleLic</i>
Affero_v1+	D.E.Knuth	ISC	O'Reilly	<i>simpleLic2</i>
Affero_v3	DARPA	<i>i9_license</i>	O'Reilly-style	<i>simpleLicense1</i>
Affero_v3+	DARPA-Cougaar	Jabber	OASIS	<i>SimpleOnlyKeepCopy right</i>
AFL	Debian-social-DFSG	Jabber_v1.0	OCL	SISSL
AFL_v1.0	Debian-SPI	Java-Multi-Corp	OCL_v1.0	SISSL_v1.1
AFL_v1.1	Debian-SPI-style	Java-WSDL-Policy	OCL-style	Skype(Non-

				commercial)
AFL_v1.2	DMTF	Java-WSDL-Schema	OCLC	Sleepycat
AFL_v2.0	DOCBOOK	Java-WSDL-Spec	OCLC_v1.0	Sleepycat(Non-commercial)
AFL_v2.1	DOCBOOK-style	Java-WSDL4J	OCLC_v2.0	Sleepycat(Oracle)
AFL_v3.0	<i>dovecotSeeCopying</i>	JPEG/netpbm	ODL	SNIA
AFPL-Ghostscript	<i>DoWhatTheFuckYouWantv2</i>	JPNIC	OMF	SNIA_v1.0
Against-DRM	DSL	KD-Tools-EULA	OMRON(2) : OMRON	SNIA_v1.1
AGE-Logic	Dyade	KDE	Ontopia	SpikeSource
Agere-EULA	EBT	Keyspan-FW	Open-PL	<i>SSLeay</i>
AGFA-EULA	EBT-style	Knuth-style	Open-PL_v1.0	<i>subversion</i>
AGFA(RESTRICTED)	ECL	Lachman-Proprietary	Open-Publication	<i>subversion+</i>
Aladdin-Ghostscript	Eclipse	Larabie-EULA	Open-Publication_v1.0	<i>subversionError</i>
Aladdin(Closed-Source!)	Eclipse_v1.0	LDP	Open-Publication-style	SUN
Aladdin(RESTRICTED)	eGenix	LDP_v1A	OpenGroup	Sun
Alfresco	Eiffel	LDP_v2.0	OpenGroup-PL	Sun-BCLA
Alfresco/FLOSS	Eiffel_v1	Legato	OpenGroup-Proprietary	Sun-BCLA_v1.5.0
Algorithmics	Eiffel_v2	LGPL	OpenGroup-style	Sun-EULA
AMD	<i>emacsLic</i>	LGPL_v1	OpenGroupTest	Sun-IP
AMD-EULA	Entessa	LGPL_v1+	OpenLDAP	Sun-Java
AOL-EULA	Epinions	LGPL_v2	OpenLDAP_v1.2	Sun-PL
Apache	Epson-EULA	LGPL_v2.1	OpenLDAP_v1.4	Sun-PL_v1.0
Apache_v1.0	Epson-PL	LGPL_v2.1+	OpenLDAP_v2.5	Sun-Proprietary
Apache_v1.1	EU-DataGrid	LGPL_v2+	OpenLDAP_v2.6	Sun-RPC
Apache_v2.0	Fair	LGPL_v3	OpenLDAP_v2.7	Sun-SCA
Apache-style	Fair-style	LGPL_v3?	OpenLDAP_v2.8	Sun(Non-commercial)

Apple	Fedora	LGPL_v3+	OpenLDAP-style	Sun(RESTRICTED)
Apple-EULA	Fedora-CLA	<i>LibGCLic</i>	OpenMarket	Sun(tm)
Apple(FontForge)	Flash2xml_v1.0	Link-exception	OpenSSL	SunPro
Apple(Sample)	Frameworkx	Linux-HOWTO	Combined OpenSSL+SSLeay	<i>Svnkit</i>
APSL	Frameworkx_v1.0	LinuxDoc	Oracle-Dev	<i>svnkit+</i>
APSL_v1.0	Free-Art_v1.0	LPPL	Oracle-EULA	<i>tmate+</i>
APSL_v1.1	Free-Art_v1.3	LPPL_v1.0	OSD	Sybase_v1.1
APSL_v1.2	Free-PL	LPPL_v1.0+	OSF	Tektronix
APSL_v2.0	Free-SW	LPPL_v1.1	OSF-style	Tektronix-style
Aptana	Free-SW(run- COMMAND)	LPPL_v1.1+	OSL	TeX-exception
Aptana-PL_v1.0	FreeBSD-Doc	LPPL_v1.2	OSL_v1.0	Trident-EULA
ARJ	Freetype	LPPL_v1.2+	OSL_v1.1	TrollTech
Arphic-Font-PL	Freetype-style	LPPL_v1.3	OSL_v2.0	U-BC
Arphic-style	FSF	LPPL_v1.3+	OSL_v2.1	U-Cambridge
Artifex	Fujitsu	LPPL_v1.3a	OSL_v3.0	U-Cambridge-style
Artistic	Garmin-EULA	LPPL_v1.3a+	Paradigm	U-Chicago
Artistic_v2.0	Genivia(Commercial)	LPPL_v1.3b	Phorum	U-Columbia
Ascender-EULA	GFDL	LPPL_v1.3b+	PHP	U-Del
ATI-EULA	GFDL_v1.1	LPPL_v1.3c	PHP_v2.0	U-Del-style
ATMEL-FW	GFDL_v1.1+	LPPL_v1.3c+	PHP_v2.0.2	U-Edinburgh
ATT	GFDL_v1.2	Lucent_v1.0	PHP_v3.0	U-Edinburgh-style
ATT-Source	GFDL_v1.2+	Lucent_v1.02	PHP_v3.01	U-Mich-style
ATT-Source_v1.0	Ghostscript-GPL	M-Plus-Project	PHP-style	U-Michigan
ATT-Source_v1.2d	Ghostscript-GPL_v1.1	MacroMedia-RPSL	Piriform	U-Monash
ATT-style	GNU-copyleft	Macrovision	Pixware-EULA	<i>u_of_i_ncsa</i>
ATT(Non-commercial)	GNU-Ghostscript	Macrovision-EULA	Platform- Computing(RESTRICT	U-Utah

			ED)	
Attribution-Assurance	GNU-Manpages	Majordomo	Polyserve-CONFIDENTIAL	U-Wash-style
AVM	GNU-style(EXECUTE)	Majordomo_v1.1	<i>Postfix</i>	U-Wash(Free-Fork)
Baekmuk-Font	GNU-style(interactive)	Mandriva	<i>postgresql</i>	U-Washington
Baekmuk(Hwan)	Google	MellanoX	Princeton	UCAR
BancTec	GPL	MetroLink	Princeton-style	UCAR-style
BEA	GPL	MetroLink-nonfree	Proprietary!	UCWare-EULA
Bellcore	GPL_v1	Mibble	Public-Use	Unicode
Bellcore-style	GPL_v1+	Mibble_v2.8	Public-Use_v1.0	UnitedLinux-EULA
BeerWareVer42	GPL_v2	Microsoft	Python	Unix-Intl
BindMITX11Var	GPL_v2:v3	MindTerm	Python_v2	USC
BH-Font	GPL_v2.1	MIT	Python_v2.0.1	USC-style
BH-Font-style	GPL_v2.1+	MIT-style	Python_v2.0.2	USC(Non-commercial)
BISON	GPL_v2+	MITEM	Python_v2.1.1	USL-Europe
Bitstream	GPLv2+KDEupgradeClause	Mitre	Python_v2.1.3	Vim
BitTorrent	GPL_v3	MitreCVW	Python_v2.2	Vixie
BitTorrent_v1.0	GPL_v3?	MitreCVW-style	Python_v2.2.3	VMware-EULA
BitTorrent_v1.1	GPL_v3+	Motorola	Python_v2.2.7	Vovida
BIZNET	GPL-classpath-exception	Motosoto_v0.9.1	Python_v2.3	W3C
BIZNET-style	GPL-exception	MPEG3-decoder	Python_v2.3.7	W3C-IP
Boost	GPL-Meta	MPL	Python_v2.4.4	W3C-style
Boost_v1.0	GPL-Bison-exception	MPL_v1.0	Python-style	Wash-U-StLouis
Boost-style	GPL-or-LGPL	MPL_v1.1	QPL	Wash-U-style
BrainStorm-EULA	GPL(rms)	MPL_v1.1+	QPL_v1.0	WebM
Broadcom-EULA	gSOAP	MPL_v2.0	Qt(Commercial)	Wintertree

BSD	gSOAP_v1.3	MPL-EULA_v1.1	QuarterDeck	WTI(Not-free)
<i>BSD1</i>	Hacktivismo	MPL-EULA_v2.0	Quest-EULA	wxWindows-LGPL
<i>BSD2</i>	Hauppauge	MPL-EULA_v3.0	RCSL	X/Open
<i>BSD3</i>	Helix/RealNetworks-EULA	MPL-style	RCSL_v1.0	X/Open-style
<i>BSD4</i>	Historical	MPL/TPL	RCSL_v2.0	X11
CCA-ND_v3.0	HP	MPL/TPL_v1.0	RCSL_v3.0	X11-style
CCA_v1.0	HP-Compaq	MRL	RealNetworks-EULA	Xerox
CCA_v2.5	HP-DEC	Ms-EULA	RealNetworks-Unknown	Xerox-style
CCA_v3.0	HP-DEC-style	Ms-indemnity	RedHat	Ximian
CCA-SA	HP-EULA	Ms-IP	RedHat-EULA	Ximian_v1.0
CCA-SA_v1.0	HP-Proprietary	Ms-LPL	RedHat-specific	x_net
CCA-SA_v2.5	HP-style	Ms-LRL	RedHat(Non-commercial)	XOPEN-EULA
CCA-SA_v3.0	HP+IBM	Ms-PL	Redland	Yahoo
CCPL	IBM	Ms-RL	Ricoh	Yahoo-EULA
CCPL_v2.0	IBM-Courier	Ms-SSL	Ricoh_v1.0	YaST(SuSE)
CCPL_v2.5	IBM-EULA	MySQL_v0.3	Riverbank-EULA	Zend_v2.0
CCPL_v3.0	IBM-JCL	MySQL-style	RPL	Zeus
CDDL	IBM-PL	MySQL/FLOSS	RPL_v1.0	zlib/libpng
CDDL_v1.0	IBM-PL_v1.0	<i>MX4J</i>	RPL_v1.1	ZoneAlarm-EULA
CeCILL	IBM-PL_v2.0	<i>MX4JLicensev1</i>	RPL_v1.5	Zope
CeCILL_v1	IBM-reciprocal	NASA	RPSL	Zope-PL_v1.0
CeCILL_v1.1	ID-EULA	NASA_v1.3	RPSL_v1.0	Zope-PL_v2.0
CeCILL_v2	IDPL	Naumen	RPSL_v2.0	Zveno
CeCILL_v2.0	IDPL_v1.0	nCipher	RPSL_v3.0	
CeCILL-B	IEEE-Doc	<i>NCSA</i>	RSA-DNS	
CeCILL-C	IETF	NESSUS-EULA	RSA-Security	

Cisco	IETF-style	NetBSD	Ruby	
Cisco-style	ImageMagick	Nethack	Same-license-as	
CMU	ImageMagick-style	Netizen	SciTech	
CMU-style	ImageMagick(Apache)	Nokia	SCO(commercial)	
CompuServe	InfoSeek	Nokia_v1.0a	SCSL	
<i>computer_associates_trusted</i>	InnerNet	Non-commercial!	SCSL_v2.3	

Παράρτημα Β

B-1 Κώδικας εργαλείου Μετατροπείας (Converter)

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.DataInputStream;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Properties;

public class Converter {
    /* create a properties object */
    static Properties props = new Properties();

    public static void Ninka() throws IOException {
        ArrayList<String> Licenses = new ArrayList<String>();// list that

        // contains the

        // licenses of

        // each file
        ArrayList<String> Files = new ArrayList<String>();// parallel list that

        // contains the name

        // of the file
        String[] temp = null;
        String temp2 = null;
```

```

try {
    /******* reading the file *****/
    /* get inputNinka property of Converter.properties file */
    FileInputStream fstream = new FileInputStream(
        props.getProperty("inputNinka"));
    DataInputStream in = new DataInputStream(fstream);
    BufferedReader br = new BufferedReader(new
InputStreamReader(in));
    String strLine;
    // Read File Line By Line
    while ((strLine = br.readLine()) != null) {
        temp = strLine.split(";");

        /* for each license check if it's already in the list */
        if (!(temp[1].equals("ERROR") || temp[1].equals("NONE") ||
temp[1]

                                .startsWith("UNMATCHED"))) {
            Licenses.add(temp[1]);
            temp2 = temp[0].replace("./", "");
            Files.add(temp2);
        }
        temp2 = "";
        // flag=0;

    }
    // Close the input stream
    in.close();
} catch (Exception e) { // Catch exception if any
    System.err.println("Error: " + e.getMessage());
}
/* call the function that converts the input file of Ninka */
ConvertNinka(Licenses, Files);

}

```

```

public static void Ohcount() throws IOException {

```

```

ArrayList<String> Licenses = new ArrayList<String>();// list that

// contains the

// licenses of

// each file
ArrayList<String> Files = new ArrayList<String>();// parallel list that

// contains the name

// of the file
String[] temp = null;
String temp2 = "";

/***** reading the file *****/
/* get inputOhcount property of Converter.properties file */
FileInputStream fstream = new FileInputStream(
    props.getProperty("inputOhcount"));
DataInputStream in = new DataInputStream(fstream);
BufferedReader br = new BufferedReader(new InputStreamReader(in));
String strLine;
// Read File Line By Line
while ((strLine = br.readLine()) != null) {
    temp = strLine.split(" ");
    /* for each license check if it's already in the list */

    if (!(temp[0].equals("Bad"))) {
        Licenses.add(temp[0]);
        for (int i = 1; i < temp.length; i++)
            temp2 = temp2 + " " + temp[i];
        Files.add(temp2);
    }
    temp2 = "";
}

```

```

        // flag=0;

    }
    // Close the input stream
    in.close();
    /* call the function that converts the input file of Ohcount */
    ConvertOhcount(Licenses, Files);
}

/*****
 * replace the names of ohcount licenses with fossology names
 *
 * @throws IOException
 *****/
public static void ConvertOhcount(ArrayList<String> Licenses,
                                   ArrayList<String> Files) throws IOException {
    ArrayList<String> Licenses1 = new ArrayList<String>(); // list that

    // contains the

    // licenses of

    // each file
    ArrayList<String> Files1 = new ArrayList<String>(); // parallel list that

    // contains the name

    // of the file

    /******replace the license names with the general format license
names*****/
    for (int i = 0; i < Licenses.size(); i++) {
        if (Licenses.get(i).equals("academic")) {
            Licenses1.add("AFL");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("adaptive")) {

```

```

        Licenses1.add("Adaptive");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("affero")) {
        Licenses1.add("Affero");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("apache")) {
        Licenses1.add("Apache");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("apache_2")) {
        Licenses1.add("Apache_v2.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("apple_open_source")) {
        Licenses1.add("APSL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("artistic")) {
        Licenses1.add("Artistic");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("attribution_assurance")) {
        Licenses1.add("Attribution-Assurance");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("boost")) {
        Licenses1.add("Boost");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("bsd")) {
        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("cecill")) {
        Licenses1.add("CeCILL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("cecill_b")) {
        Licenses1.add("CeCILL-B");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("cecill_c")) {
        Licenses1.add("CeCILL-C");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("computer_associates_trusted")) {

```

```

        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals(
        "common_development_and_distribution")) {
        Licenses1.add("CDDL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("common_public")) {
        Licenses1.add("CPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("cua_office")) {
        Licenses1.add("CUA_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("eu_datagrid")) {
        Licenses1.add("EU-DataGrid");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("eclipse")) {
        Licenses1.add("Eclipse");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("educational")) {
        Licenses1.add("ECL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("eiffel")) {
        Licenses1.add("Eiffel");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("eiffel_2")) {
        Licenses1.add("Eiffel_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("entessa")) {
        Licenses1.add("Entessa");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("fair")) {
        Licenses1.add("Fair");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("frameworx")) {
        Licenses1.add("Frameworx");
        Files1.add(Files.get(i));
    }

```

```

} else if (Licenses.get(i).equals("gpl3_or_later")) {
    Licenses1.add("GPL_v3+");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("gpl3")) {
    Licenses1.add("GPL_v3");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("lgpl3")) {
    Licenses1.add("LGPL_v3");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("gpl")) {
    Licenses1.add("GPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("lgpl")) {
    Licenses1.add("LGPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("historical")) {
    Licenses1.add("Historical");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("i9_license")) {
    Licenses1.add("i9_license");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("ibm_public")) {
    Licenses1.add("IBM-PL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("intel_open_source")) {
    Licenses1.add("Intel");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("jabber_open_source")) {
    Licenses1.add("Jabber");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("lucent_plan9")) {
    Licenses1.add("Lucent_v1.0");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("lucent_public")) {
    Licenses1.add("Lucent_v1.0");
    Files1.add(Files.get(i));

```

```

} else if (Licenses.get(i).equals("mit")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("mitre")) {
    Licenses1.add("Mitre");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("motosoto")) {
    Licenses1.add("Motosoto_v0.9.1");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("mozilla_public_1")) {
    Licenses1.add("MPL_v1.0");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("mozilla_public_1_1")) {
    Licenses1.add("MPL_v1.1");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("nasa_open")) {
    Licenses1.add("NASA");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("naumen")) {
    Licenses1.add("Naumen");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("nethack")) {
    Licenses1.add("Nethack");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("nokia_open_source")) {
    Licenses1.add("Nokia");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("oclc_research")) {
    Licenses1.add("OCLC");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("open_group_test")) {
    Licenses1.add("OpenGroupTest");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("open_software")) {
    Licenses1.add("OSL");
    Files1.add(Files.get(i));

```

```

} else if (Licenses.get(i).equals("php_license")) {
    Licenses1.add("PHP");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("python_license")) {
    Licenses1.add("Python");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("python_software_foundation")) {
    Licenses1.add("Python_v2");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("qt_public")) {
    Licenses1.add("QPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("realnetworks_public_source")) {
    Licenses1.add("RPSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("reciprocal_public")) {
    Licenses1.add("RPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("ricoh_source")) {
    Licenses1.add("Ricoh");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sleepycat")) {
    Licenses1.add("Sleepycat");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sugarcrm_1_1_3")) {
    Licenses1.add("SugarCRM(attribution)");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sun_industry_standards")) {
    Licenses1.add("SISSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sun_public")) {
    Licenses1.add("Sun-PL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sybase_open_watcom")) {
    Licenses1.add("Sybase");
    Files1.add(Files.get(i));

```

```

    } else if (Licenses.get(i).equals("u_of_i_ncsa")) {
        Licenses1.add("u_of_i_ncsa");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("vovida_software")) {
        Licenses1.add("Vovida");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("w3c")) {
        Licenses1.add("W3C");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("wx_windows")) {
        Licenses1.add("wxWindows-LGPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("x_net")) {
        Licenses1.add("x_net");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("zope")) {
        Licenses1.add("Zope");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("zlib_libpng")) {
        Licenses1.add("zlib/libpng");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("apache_ish")) {
        Licenses1.add("Apache");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("bsd_ish")) {
        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("bsd_2clause_ish")) {
        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    }
}

/***** print Files and Licenses to the output file *****/

// Create file

/* get outputOhcount property of Converter.properties file */

```

```

        FileWriter fstream1 = new FileWriter(props.getProperty("outputOhcount"));
        BufferedWriter out = new BufferedWriter(fstream1);
        for (int i = 0; i < Licenses1.size(); i++) {
            out.write(Licenses1.get(i) + " ");
            out.write(Files1.get(i));
            if (i != Licenses.size() - 1)
                out.newLine();
        }
        out.close();
        System.out.println("Ohcount output file was created!");
    }

    /**
     * replace the names of ninka licenses with fossology names
     *
     * @throws IOException
     */
    public static void ConvertNinka(ArrayList<String> Licenses,
                                   ArrayList<String> Files) throws IOException {
        ArrayList<String> Licenses1 = new ArrayList<String>(); // list that

        // contains the

        // licenses of

        // each file
        ArrayList<String> Files1 = new ArrayList<String>(); // parallel list that

        // contains the name

        // of the file

        /**replace the license names with the general format license
names***/
        for (int i = 0; i < Licenses.size(); i++) {
            if (Licenses.get(i).equals("AGPLv3+")) {

```

```

        Licenses1.add("Affero_v3+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("Apachev1.1")) {
        Licenses1.add("Apache_v1.1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("Apachev2")) {
        Licenses1.add("Apache_v2.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("artifex")) {
        Licenses1.add("Artifex");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("ArtisticLicensev1")) {
        Licenses1.add("Artistic");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("autoConfException")) {
        Licenses1.add("GPL-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BeerWareVer42")) {
        Licenses1.add("BeerWareVer42");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BindMITX11Var")) {
        Licenses1.add("BindMITX11Var");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BisonException")) {
        Licenses1.add("BISON");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("boost")) {
        Licenses1.add("Boost");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("boostV1")) {
        Licenses1.add("Boost_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD1")) {
        Licenses1.add("BSD1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2")) {

```

```

        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2AdvInsteadOfBinary")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2aic700")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2EndorseInsteadOfBinary")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2SoftAndDoc")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2var1")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD2var2")) {
        Licenses1.add("BSD2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD3")) {
        Licenses1.add("BSD3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD3NoWarranty")) {
        Licenses1.add("BSD3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD4")) {
        Licenses1.add("BSD4");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSD4NoEndor")) {
        Licenses1.add("BSD4");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSDCairoStyleWarr")) {
        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSDdovecotStyle")) {

```

```

        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("BSDOnlyAdv")) {
        Licenses1.add("BSD");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("CDDLic")) {
        Licenses1.add("CDDL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("CDDLicV1")) {
        Licenses1.add("CDDL_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("CDDLv1orGPLv2")) {
        Licenses1.add("CDDL_v1.0");
        Files1.add(Files.get(i));
        Licenses1.add("GPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("Cecill")) {
        Licenses1.add("Cecill");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("ClassPathException")) {
        Licenses1.add("GPL-classpath-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("CPLv0.5")) {
        Licenses1.add("CPL_v0.5");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("CPLv1")) {
        Licenses1.add("CPL_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("dovecotSeeCopying")) {
        Licenses1.add("dovecot");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("DoWhatTheFuckYouWantv2")) {
        Licenses1.add("DoWhatTheFuckYouWantv2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("emacsLic")) {
        Licenses1.add("emacsLic");

```

```

        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("EPLv1")) {
        Licenses1.add("Eclipse_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("FreeType")) {
        Licenses1.add("FreeType");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GhostscriptGPL")) {
        Licenses1.add("Ghostscript-GPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLnoVersion")) {
        Licenses1.add("GPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv1")) {
        Licenses1.add("GPL_v1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv1+")) {
        Licenses1.add("GPL_v1+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv1orArtistic")) {
        Licenses1.add("GPL_v1");
        Files1.add(Files.get(i));
        Licenses1.add("Artistic");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv2")) {
        Licenses1.add("GPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv2+")) {
        Licenses1.add("GPL_v2+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv2orLGPLv2.1")) {
        Licenses1.add("GPL_v2");
        Files1.add(Files.get(i));
        Licenses1.add("LGPL_v2.1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv2orv3")) {

```

```

        Licenses1.add("GPL_v2:v3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv2orv3qtException")) {
        Licenses1.add("GPL-classpath-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv3")) {
        Licenses1.add("GPL_v3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("GPLv3+")) {
        Licenses1.add("GPL_v3+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("IBMv1")) {
        Licenses1.add("IBM-PL_v1.0");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("intelBSDLicense")) {
        Licenses1.add("Intel");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("InterACPILic")) {
        Licenses1.add("Interbase-PL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("kerberos")) {
        Licenses1.add("MIT");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLnoVersion")) {
        Licenses1.add("LGPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv2")) {
        Licenses1.add("LGPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv2+")) {
        Licenses1.add("LGPL_v2+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv2.1")) {
        Licenses1.add("LGPL_v2.1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv2.1+")) {

```

```

        Licenses1.add("LGPL_v2.1+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv3")) {
        Licenses1.add("LGPL_v3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LesserGPLv3+")) {
        Licenses1.add("LGPL_v3+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2")) {
        Licenses1.add("LGPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2+")) {
        Licenses1.add("LGPL_v2+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2.1")) {
        Licenses1.add("LGPL_v2.1");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2.1+")) {
        Licenses1.add("LGPL_v2.1+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv3")) {
        Licenses1.add("LGPL_v3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv3+")) {
        Licenses1.add("LGPL_v3+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2MISTAKE")) {
        Licenses1.add("LGPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2+MISTAKE")) {
        Licenses1.add("LGPL_v3");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LGPLv2orv3")) {
        Licenses1.add("LGPL_v3");
        Files1.add(Files.get(i));
        Licenses1.add("LGPL_v2");
    }

```

```

        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LibGCJLic")) {
        Licenses1.add("LibGCJLic");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LibraryGPLv2")) {
        Licenses1.add("GPL_v2");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LibraryGPLv2+")) {
        Licenses1.add("GPL_v2+");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LinkException")) {
        Licenses1.add("Link-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LinkExceptionBison")) {
        Licenses1.add("Link-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LinkExceptionGPL")) {
        Licenses1.add("Link-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("inkExceptionLeGPL")) {
        Licenses1.add("Link-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("LinkExceptionOpenSSL")) {
        Licenses1.add("Link-exception");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("MITandGPL")) {
        Licenses1.add("MIT");
        Files1.add(Files.get(i));
        Licenses1.add("GPL");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("MITCMU")) {
        Licenses1.add("MIT");
        Files1.add(Files.get(i));
    } else if (Licenses.get(i).equals("MITCMUvar2")) {
        Licenses1.add("MIT");
        Files1.add(Files.get(i));
    }

```

```

} else if (Licenses.get(i).equals("MITCMUvar3")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITmodern")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITold")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MIToldMichiganVersion")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MIToldwithoutSell")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals(
    "MIToldwithoutSellandNoDocumentationRequi")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MIToldwithoutSellCMUVariant")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITVariant")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITX11BSDvar")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
    Licenses1.add("X11");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITX11noNotice")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
    Licenses1.add("X11");
    Files1.add(Files.get(i));

```

```

    } else if (Licenses.get(i).equals("MITX11NoSellNoDocDocBSDvar"))
{
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
    Licenses1.add("X11");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MITX11simple")) {
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
    Licenses1.add("X11");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MPL1_1andLGPLv2_1")) {
    Licenses1.add("MPL_v1.1");
    Files1.add(Files.get(i));
    Licenses1.add("LGPL_v2.1");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MPLGPL2orLGPLv2_1")) {
    Licenses1.add("MPL_v1.1");
    Files1.add(Files.get(i));
    Licenses1.add("LGPL_v2.1");
    Files1.add(Files.get(i));
    Licenses1.add("GPL_v2");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MPL_LGPLsee")) {
    Licenses1.add("MPL");
    Files1.add(Files.get(i));
    Licenses1.add("LGPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MPL-MIT-dual")) {
    Licenses1.add("MPL");
    Files1.add(Files.get(i));
    Licenses1.add("MIT");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MPLv1_0")) {
    Licenses1.add("MPLv1_0");
    Files1.add(Files.get(i));

```

```

} else if (Licenses.get(i).equals("MPLv1_1")) {
    Licenses1.add("MPL_v1.1");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MX4J")) {
    Licenses1.add("MX4J");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("MX4JLicensev1")) {
    Licenses1.add("MX4J");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("NCSA")) {
    Licenses1.add("NCSA");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("NPLv1_0")) {
    Licenses1.add("NPL_v1.0");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("NPLv1_1")) {
    Licenses1.add("NPL_v1.1");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("openssl")) {
    Licenses1.add("OpenSSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("opensslvar1")) {
    Licenses1.add("OpenSSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("opensslvar1")) {
    Licenses1.add("OpenSSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("opensslvar1")) {
    Licenses1.add("OpenSSL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("phpLicV3.01")) {
    Licenses1.add("PHP_v3.01");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("Postfix")) {
    Licenses1.add("Postfix");
    Files1.add(Files.get(i));

```

```

} else if (Licenses.get(i).equals("postgresql")) {
    Licenses1.add("PostgreSQL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("publicDomain")) {
    Licenses1.add("Public-Use");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("QtGPLv2or3")) {
    Licenses1.add("QPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("QTv1")) {
    Licenses1.add("QPL_v1.0");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sequenceLic")) {
    Licenses1.add("Sequence");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SameTermsAs")) {
    Licenses1.add("Same-license-as");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SeeFile")) {
    Licenses1.add("See-file(README)");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SimpleLic")) {
    Licenses1.add("SimpleLic");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("simpleLic2")) {
    Licenses1.add("SimpleLic");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("simpleLic")) {
    Licenses1.add("SimpleLic");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SimpleOnlyKeepCopyright")) {
    Licenses1.add("SimpleLic");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SleepyCat")) {
    Licenses1.add("Sleepycat");
    Files1.add(Files.get(i));

```

```

} else if (Licenses.get(i).equals("SSLeay")) {
    Licenses1.add("SSLeay");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("subversion")) {
    Licenses1.add("Subversion");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("subversion+")) {
    Licenses1.add("Subversion");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("subversionError")) {
    Licenses1.add("Subversion");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("sunRPC")) {
    Licenses1.add("Sun-RPC");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("SunSimpleLic")) {
    Licenses1.add("Sun");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("svnkit")) {
    Licenses1.add("SvnKit");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("svnkit+")) {
    Licenses1.add("SvnKit");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("tmate+")) {
    Licenses1.add("Tmate+");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("W3CLic")) {
    Licenses1.add("W3C");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("WxException")) {
    Licenses1.add("wxWindows-LGPL");
    Files1.add(Files.get(i));
} else if (Licenses.get(i).equals("X11")) {
    Licenses1.add("X11");
    Files1.add(Files.get(i));

```

```

        } else if (Licenses.get(i).equals("X11CMU")) {
            Licenses1.add("X11");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("X11Festival")) {
            Licenses1.add("X11");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("X11mit")) {
            Licenses1.add("X11");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("zendv2")) {
            Licenses1.add("Zend_v2.0");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("ZLIB")) {
            Licenses1.add("zlib/libpng");
            Files1.add(Files.get(i));
        } else if (Licenses.get(i).equals("ZLIBref")) {
            Licenses1.add("zlib/libpng");
            Files1.add(Files.get(i));
        }
    }

    /***** print Files and Licenses to the output file *****/

    // Create file
    /* get outputNinka property of Converter.properties file */
    FileWriter fstream1 = new FileWriter(props.getProperty("outputNinka"));
    BufferedWriter out = new BufferedWriter(fstream1);
    for (int i = 0; i < Licenses1.size(); i++) {
        out.write(Licenses1.get(i) + " ");
        out.write(Files1.get(i));
        if (i != Licenses.size() - 1)
            out.newLine();
    }
    out.close();
    System.out.println("Ninka output file was created!");
}

```

```

public static void Fossology() throws IOException {
    ArrayList<String> Licenses = new ArrayList<String>();
    ArrayList<String> Files = new ArrayList<String>();
    String[] temp = null;
    String[] tempComma=null;
    int i=0;

    /******* reading the file *****/

    /* get inputFossology property of Converter.properties file */
    FileInputStream fstream = new FileInputStream(
        props.getProperty("inputFossology"));
    DataInputStream in = new DataInputStream(fstream);
    BufferedReader br = new BufferedReader(new InputStreamReader(in));
    String strLine;

    // Read File Line By Line
    while ((strLine = br.readLine()) != null) {
        temp = strLine.split(" ");
        /* for each license check if it's already in the list */
        if (temp[0].equals("File")) {
            if (!(temp[4].equals("No") || temp[4].equals("None"))) {
                /*check if a file has more than one licenses divided by
                ", " */

                if(temp[4].contains(", ")){
                    tempComma=temp[4].split(", ");
                    for(i=0;i<tempComma.length;i++){
                        /*add the licenses of the file to the list*/
                        Licenses.add(tempComma[i]);
                        /*add the name of the file to the list*/
                        Files.add(temp[1]);
                    }
                }
            }
        }
    }
}

```

```

        /*add the licenses of the file to the list*/
        Licenses.add(temp[4]);
        /*add the name of the file to the list*/
        Files.add(temp[1]);
    }
}
}

/***** print Files and Licenses to the output file *****/

// Create file
/* get ourputFossology property of Converter.properties file */
FileWriter fstream1 = new FileWriter(
    props.getProperty("outputFossology"));
BufferedWriter out = new BufferedWriter(fstream1);
for (i = 0; i < Licenses.size(); i++) {
    out.write(Licenses.get(i) + " ");
    out.write(Files.get(i));
    if (i != Licenses.size() - 1)
        out.newLine();
}
out.close();
System.out.println("Fossology output file was created!");
}

/**
 * @param args
 * @throws IOException
 */
public static void main(String[] args) throws IOException {
    // TODO Auto-generated method stub
    /* read the Converter.properties file to get the input and output files */
    props.load(new FileInputStream("Converter.properties"));

    Scanner in = new Scanner(System.in);

```

```

int choice=0;

/* call the function that reads the input file of Fossology */
while(choice!=4){
    System.out.println("=====Tools=====");
    System.out.println("1.Fossology");
    System.out.println("2.Ninka");
    System.out.println("3.Ohcount");
    System.out.println("4.Exit");
    System.out.println("Please select an input file to convert: ");

    choice = in.nextInt();
    if(choice==1)
        Fossology();
    /* call the function that reads the input file of Ninka */
    if(choice==2)
        Ninka();
    /* call the function that reads the input file of Ohcount */
    if(choice==3)
        Ohcount();

}

}

}

```

B-2 Κώδικας εργαλείου Ελεγκτής Συμβατότητας (Compatibility Check)

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.DataInputStream;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Properties;

/*JGraphT libraries*/
import org.jgrapht.DirectedGraph;
import org.jgrapht.graph.*;

/*Dijkstra library*/
import org.jgrapht.alg.DijkstraShortestPath;

public class Compatibility {

    /******* variables for each license *****/
    static final String zlib = "zlib/libpng";
    static final String cecillB = "CeCILL-B";
    static final String mit = "MIT";
    static final String x11 = "X11";
    static final String bsd2 = "BSD2";
    static final String freeBsd = "Free-BSD";
    static final String apache2 = "Apache_v2.0";
    static final String cecillC = "CeCILL-C";
    static final String mpl2 = "MPL_v2.0";
    static final String lgpl2_1 = "LGPL_v2.1";
    static final String lgpl3plus = "LGPL_v3+";
    static final String gpl2 = "GPL_v2";
    static final String gpl2plus = "GPL_v2+";
    static final String gpl3plus = "GPL_v3+";
```

```

static final String cecill2 = "CeCILL_v2";
static final String affero1 = "Affero_v1";
static final String affero2 = "Affero_v2";
static final String affero3 = "Affero_v3";
static final String eupl = "EUPL";

/* create a properties object */
static Properties props = new Properties();

static DirectedGraph<String, DefaultEdge> graph = new
DefaultDirectedGraph<String, DefaultEdge>(
    DefaultEdge.class);

/***** function which creates the graph *****/
public static DirectedGraph<String, DefaultEdge> CreateGraph() {

    /* create the directed graph */
    DirectedGraph<String, DefaultEdge> g = new DefaultDirectedGraph<String,
DefaultEdge>(
        DefaultEdge.class);

    /*** create the nodes of the graph ***/
    g.addVertex(zlib);
    g.addVertex(cecillB);
    g.addVertex(mit);
    g.addVertex(x11);
    g.addVertex(bsd2);
    g.addVertex(freeBsd);
    g.addVertex(apache2);
    g.addVertex(cecillC);
    g.addVertex(mpl2);
    g.addVertex(lgpl2_1);
    g.addVertex(lgpl3plus);
    g.addVertex(gpl2);
    g.addVertex(gpl2plus);
    g.addVertex(gpl3plus);

```

```

g.addVertex(cecill2);
g.addVertex(affero1);
g.addVertex(affero2);
g.addVertex(affero3);
g.addVertex(eupl);

/**** create the edges between the nodes of the graph ****/
g.addEdge(zlib, gpl2plus);
g.addEdge(cecillB, mit);
g.addEdge(cecillB, x11);
g.addEdge(mit, bsd2);
g.addEdge(x11, bsd2);
g.addEdge(mit, freeBsd);
g.addEdge(x11, freeBsd);
g.addEdge(bsd2, lgpl2_1);
g.addEdge(bsd2, mit);
g.addEdge(bsd2, x11);
g.addEdge(freeBsd, apache2);
g.addEdge(freeBsd, mpl2);
g.addEdge(freeBsd, mit);
g.addEdge(freeBsd, x11);
g.addEdge(apache2, lgpl3plus);
g.addEdge(lgpl3plus, gpl3plus);
g.addEdge(cecillC, lgpl2_1);
g.addEdge(mpl2, lgpl2_1);
g.addEdge(lgpl2_1, lgpl3plus);
g.addEdge(lgpl2_1, gpl2plus);
g.addEdge(gpl2plus, gpl2);
g.addEdge(gpl2plus, gpl3plus);
g.addEdge(cecillB, gpl2plus);
g.addEdge(eupl, cecillB);
g.addEdge(gpl3plus, affero2);
g.addEdge(affero1, affero3);
g.addEdge(affero2, affero3);

System.out.println("Graph created!");

```

```

        return g;
    }

/**** function which compares the compatibility between the input licenses *****/
public static void Compare(
    ArrayList<String> Licenses, ArrayList<String> Files)
    throws IOException {

    /* 2 parallel lists which contain the conflicted licenses */
    ArrayList<String> Conflicts1 = new ArrayList<String>();
    ArrayList<String> Conflicts2 = new ArrayList<String>();

    ArrayList<String> List1 = new ArrayList<String>(); // list which

        // contains the

        // conflicted

        // licenses(one time

        // each) for writing

        // the result in the

        // file

    int ConflictFlag = 0; // flag which changes to 1 if any conflicts were
                        // detected

    /*
    * check for each input license if there is a path between each one of
    * the rest licenses
    */
    for (int i = 0; i < Licenses.size() - 1; i++)
        for (int j = i + 1; j < Licenses.size(); j++)

```

```

/*
 * check if the two licenses exist already in the two parallel
 * lists(if they were already checked)
 */
if (!(Conflicts1.contains(Licenses.get(i))
      && Conflicts2.contains(Licenses.get(j)) ||

Conflicts2

      .contains(Licenses.get(i))
      && Conflicts1.contains(Licenses.get(j)))) {

/*
 * if the two licenses haven't been checked yet check if
the
 * licenses exist as nodes in the graph
 */
if (graph.containsVertex(Licenses.get(i))
    &&
graph.containsVertex(Licenses.get(j))
    &&
!Licenses.get(i).equals(Licenses.get(j))) {

/*
 * call Dijkstra function to find if there is a path
 * between the two nodes
 */
java.util.List<DefaultEdge> path =

DijkstraShortestPath

      .findPathBetween(graph,
Licenses.get(i),

      Licenses.get(j));

// Prints the shortest path from vertex
Licenses.get(i)

// to vertex Licenses.get(j). This path does
// NOT exist for our particular directed graph.

Hence

// the path is
B-32

```

```

// empty and the variable path must be null.
path =
DijkstraShortestPath.findPathBetween(graph,
                                Licenses.get(i), Licenses.get(j));

/*
 * call for a second time the Dijkstra function
but
 * change the order of the two nodes
 */
java.util.List<DefaultEdge> path2 =
DijkstraShortestPath
                                .findPathBetween(graph,
Licenses.get(j),
                                Licenses.get(i));

// Prints the shortest path from vertex
Licenses.get(j)
// to vertex Licenses.get(i). This path does
// NOT exist for our particular directed graph.
Hence
// the path is
// empty and the variable path must be null.

path2 =
DijkstraShortestPath.findPathBetween(graph,
                                Licenses.get(j), Licenses.get(i));

/*
 * check if both functions return null then there
is not
 * a path between the two nodes
 */
if (path == null && path2 == null) {
    Conflicts1.add(Licenses.get(i)); // add the
first

```

second

```
// license to

// Conflicts1 list
Conflicts2.add(Licenses.get(j)); // add the

// license to the

// parallel

// Conflicts2 list

    }
}

}

/* if conflicts were discovered */
if (!Conflicts1.isEmpty()) {
    System.out.println("Conflicts were detected!");
    ConflictFlag = 1;
} else {
    /* if no conflicts were discovered */
    System.out.println("No conflicts were detected!");
    ConflictFlag = 0;
}

/* if conflicts were detected then write the results to the output file */
/* get the output property of Compatibility.properties file */
if (ConflictFlag == 1) {
    FileWriter fstream = new FileWriter(props.getProperty("output")); //
    // for
    // writing the
    // results in
    // the output
```

filerwiter

```

// file
BufferedWriter out = new BufferedWriter(fstream);// bufferedwiter

// for writing

// the results

// in the output

// file
// print the result to the output file

out.write("=====
=====");

out.write("\n");
out.write("=====           Conflicts   between   licenses
=====");

out.write("\n");

out.write("=====
=====");

out.write("\n");
out.write("\n");

// print the conflicted licenses
for (int k = 0; k < Conflicts1.size(); k++) {
    out.write("Conflict between license: " + Conflicts1.get(k)
              + " and license: " + Conflicts2.get(k));
    out.write("\n");
}

// add the conflicted licenses from the Conflicts1 list ONLY one
// time each
for (int k = 0; k < Conflicts1.size(); k++)
    if (!List1.contains(Conflicts1.get(k)))
        List1.add(Conflicts1.get(k));

```

```

// add the conflicted licenses from the Conflicts2 list ONLY one
// time each
for (int k = 0; k < Conflicts2.size(); k++)
    if (!List1.contains(Conflicts2.get(k)))
        List1.add(Conflicts2.get(k));

// print the files that contain each conflicted license to the
// output file
out.write("\n");

out.write("=====
=====");

out.write("\n");
out.write("=====   Files that contain the conflicted licenses
=====");

out.write("\n");

out.write("=====
=====");

out.write("\n");
out.write("\n");

// find for each conflicted license the files that contain that
// license
for (int k = 0; k < List1.size(); k++) {
    out.write("The files that contain the " + List1.get(k)
        + " license are: ");
    out.write("\n");
    /*print to the file the names of the files for each license*/
    for (int i = 0; i < Files.size(); i++)
        if (Licenses.get(i).equals(List1.get(k))) {
            out.write(Files.get(i));
            out.write("\n");
        }
    out.write("\n");
}

```

```

        }
        out.close();
        System.out.println("Results were written in the output file!");
    }
}

/**
 * @param args
 * @throws IOException
 */

public static void main(String[] args) throws IOException {
    // TODO Auto-generated method stub

    ArrayList<String> Licenses = new ArrayList<String>();// list that

        // contains the

        // licenses of

        // each file
    ArrayList<String> Files = new ArrayList<String>();// parallel list that

        // contains the name

        // of the file
    String[] temp = null;
    int i = 1;

    /***** read the file *****/
    /*
     * read the Compatibility.properties file to get the input and output
     * file
     */
    props.load(new FileInputStream("Compatibility.properties"));

```

```

/* get input property of Compatibility.properties file */
FileInputStream fstream = new FileInputStream(
    props.getProperty("input"));
DataInputStream in = new DataInputStream(fstream);
BufferedReader br = new BufferedReader(new InputStreamReader(in));
String strLine;
String tempString = "";
// Read File Line By Line
while ((strLine = br.readLine()) != null) {
    if (!strLine.isEmpty()) {
        temp = strLine.split(" ");
        Licenses.add(temp[0]);
        /*
         * if there is a space between the name of a file combine it as
         * a one
         */
        while (i < temp.length) {
            tempString = tempString + " " + temp[i];
            i++;
        }
        Files.add(tempString);
        tempString = "";
        i = 1;
    }
}

System.out.println("Input file was read!");

graph = CreateGraph();// function that creates the graph
Compare(Licenses, Files);// function that finds the conflicts
// between the
input licenses
}
}

```

Παράρτημα Γ

Γ-1 Αρχείο readme εργαλείου Μετατροπéας (Converter)

README File Converter Program

1. Program Description

This program is used to convert the output files that Ninka, Fossology and Ohcount tool produces when they scan the input files in order to detect licenses. For each tool there is a function that converts the original output of the tool to a general format and then produces the new output file. The new output file will be used as input to Compatibility program (see ReadMe-Compatiblity.doc)

2. Inputs-Outputs

As input the program can take 3 files, one for each tool. From a menu the user can select the input file of the tool that he wishes to convert. In the main function of the program there are 3 functions:

- Ninka(): takes as input the file inputNinka.txt and converts it and then it produces the new output file with the name ConverterOutput-Ninka.txt. The name of the input file is contained in the inputNinka property and the output file is contained in the outputNinka property of Converter.properties file.
- Fossology():takes as input the file inputFoss.txt and converts it and then it produces the new output file with the name ConverterOutput-Fossology.txt. The name of the input file is contained in the inputFossology property and the output file is contained in the outputFossology property of Converter.properties file.
- Ohcount():takes as input the file inputOhcount.txt and converts it and then it produces the new output file with the name ConverterOutput-Ohcount.txt. The name of the input file is contained in the inputOhcount property and the output file is contained in the outputOhcount property of Converter.properties file.

Γ-2 Αρχείο readme εργαλείου Έλεγχος συμβατότητας (Compatibility Check)

README File

Compatibility Program

1. Program Description

This program is used to check the compatibility and detect any conflicts between the licenses contained in the input file.

2. Inputs

As input the program takes 1 input file “input.txt” that contains the names of the files and the licenses that are included in them. The name of the input file is contained in the input property of Compatibility.properties file.

The input file is basically generated by the Converter program (see ReadMe-Converter.doc).

The format of the input file is:

LicenseName<WS><WS>FileName

Where <WS> is white space.

3. Outputs

In file:

As output the program produces an output.txt file where the conflicts that were detected are described and the files that contain the conflicted licenses are presented.

The name of the output file is contained in the output property of Compatibility.properties file.

If no conflicts are detected then the output.txt file is not generated.

In terminal:

The steps that the program completes appear in the terminal.

Input file was read! : The input file has been read.

Graph created! : The dependencies graph has been created.

Conflicts were detected! : Conflicts were detected between licenses of the input file.

No conflicts were detected! : No conflicts were detected between licenses of the input file.

Results were written in the output file! : The results were written in the output.txt file (only if conflicts were detected) .