

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ**

**ΣΧΟΛΗ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΕΡΓΑΣΙΑ ΜΑΘΗΜΑΤΟΣ ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ ΠΟΛΥΠΛΟΚΟΤΗΤΑ**

**ΑΛΓΟΡΙΘΜΟΙ ΧΡΩΜΑΤΙΣΜΟΥ ΓΡΑΦΗΜΑΤΩΝ**

Παύλος Μπόζιος

15372 / 1290

Άρτα 2020

# ΠΕΡΙΛΗΨΗ

Το πρόβλημα του χρωματισμού γραφήματος είναι ένα NP‐hard πρόβλημα συνδυαστικής βελτιστοποίησης. Αφορά την ανάθεση ενός χρώματος σε κάθε κορυφή ενός γραφήματος έτσι ώστε γειτονικές κορυφές να χρωματίζονται με διαφορετικό χρώμα, ενώ παράλληλα χρησιμοποιείται ο ελάχιστος αριθμός διαφορετικών χρωμάτων. Στην παρούσα εργασία ζητείται η υλοποίηση τεσσάρων αλγορίθμων χρωματισμού γραφημάτων και η εφαρμογή τους σε γνωστά προβλήματα από τη βιβλιογραφία.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

[ΠΕΡΙΛΗΨΗ 2](#_Toc61184618)

[ΕΙΣΑΓΩΓΗ 5](#_Toc61184619)

[Κεφάλαιο 1. Περιγραφή προβλήματος χρωματισμού γραφήματος 6](#_Toc61184620)

[1.1 Κατηγορίες υπολογιστικών προβλημάτων και πολυωνυμικός χρόνος 6](#_Toc61184621)

[1.1.1 Class P (Polynomial time) 6](#_Toc61184622)

[1.1.2 Class NP (Nondeterministic Polynomial time) 6](#_Toc61184623)

[1.1.3 Class NP-Hard (Nondeterministic Polynomial-time hard) 6](#_Toc61184624)

[1.1.4 Class NP-Complete 6](#_Toc61184625)

[1.2 Ορισμός προβλήματος χρωματισμού γραφήματος 7](#_Toc61184626)

[Κεφάλαιο 2. Προσεγγίσεις επίλυσης 9](#_Toc61184627)

[2.1 Δεδομένα προβλήματος 9](#_Toc61184628)

[2.2 Στατιστικά στοιχεία 9](#_Toc61184629)

[2.2.1 Πίνακας στατιστικών στοιχείων 10](#_Toc61184630)

[Κεφάλαιο 3. Επίλυση προβλήματος 11](#_Toc61184631)

[3.1 First Fit (FF) 11](#_Toc61184632)

[3.2 Degree of Saturation (DSatur) 11](#_Toc61184633)

[3.3 Recursive Large First (RLF) 11](#_Toc61184634)

[3.4 Backtracking Dsatur 12](#_Toc61184635)

[Κεφάλαιο 4. Αποτελέσματα 13](#_Toc61184636)

[4.1 Χρωματισμός γραφημάτων με τον αλγόριθμο First Fit 13](#_Toc61184637)

[ΒΙΒΛΙΟΓΡΑΦΙΑ 14](#_Toc61184653)

# ΕΙΣΑΓΩΓΗ

Το πρόβλημα χρωματισμού γραφήματος τυπικά ορίζεται ως εξής. Δεδομένου ενός μη κατευθυνόμενου απλού γραφήματος G = (V, E) με ένα σύνολο κορυφών V και ένα σύνολο ακμών E, ζητείται η ανάθεση σε κάθε κορυφή v ∈ V ενός ακεραίου c(v) ∈ {1, 2, ..., k} έτσι ώστε το k να ελαχιστοποιείται και να ισχύει ότι c(v) ≠ c(u) ∀{v, u} ∈ E. Το πρόβλημα συναντάται σε μεγάλο αριθμό πρακτικών εφαρμογών όπως ο χρονοπρογραμματισμός εκπαιδευτικών ιδρυμάτων (educational timetabling), ο χρονοπογραμματισμός αθλητικών γεγονότων (sports scheduling), η ανάθεση συχνοτήτων (frequency assignment), η ανάθεση καταχωρητών στους μεταγλωττιστές (compiler register allocation) και άλλα. Πολλοί αλγόριθμοι χρωματισμού γραφημάτων έχουν προταθεί τα τελευταία 50 έτη. Στην παρούσα εργασία θα εξεταστούν τέσσερις αλγόριθμοι που ανήκουν στις λεγόμενες κατασκευαστικές τεχνικές (constructive techniques). Οι κατασκευαστικές τεχνικές δημιουργούν λύσεις βήμα προς βήμα, αναθέτοντας στη σειρά σε κάθε κορυφή ένα χρώμα. Οι αλγόριθμοι που θα εξεταστούν είναι ο αλγόριθμος First Fit, ο αλγόριθμος DSATUR, ο αλγόριθμος Recursive Largest First και ο αλγόριθμος backtracking DSATUR.

# Κεφάλαιο 1. Περιγραφή προβλήματος χρωματισμού γραφήματος

## 1.1 Κατηγορίες υπολογιστικών προβλημάτων και πολυωνυμικός χρόνος

### 1.1.1 Class P (Polynomial time)

Οποιοδήποτε υπολογιστικό πρόβλημα που μπορεί να λυθεί σε πολυωνυμικό χρόνο θεωρείται μέλος της κατηγορίας P. Δεδομένου ενός υπολογιστικού προβλήματος, εάν υπάρχει ένας πολυώνυμος αλγόριθμος χρόνου που λύνει το πρόβλημα όπου θα πρέπει να αποδείξει ότι ο αλγόριθμος πράγματι παρέχει πάντα τη βέλτιστη λύση στο πρόβλημα και θα ταξινομηθεί ως κατηγορία P.

### 1.1.2 Class NP (Nondeterministic Polynomial time)

Οποιοδήποτε υπολογιστικό πρόβλημα όπου μπορεί επαληθευθεί ως προς την ορθότητα του σε πολυωνυμικό χρόνο θεωρείται NP. Ωστόσο, παρόλο που πρέπει να επαληθευθεί μια λύση σε πολυωνυμικό χρόνο, δεν χρειάζεται απαραίτητα υπολογισθεί μια σωστή λύση σε πολυωνυμικό χρόνο, δηλαδή δεν είναι απαραίτητο να υπάρχει πολυωνυμικός χρόνος σε ένα αλγόριθμο ο οποίος να επιλύει το πρόβλημα βέλτιστα. Εάν τα NP είναι κατηγορία υπολογιστικών προβλημάτων που μπορούν να επαληθευτούν σε πολυωνυμικό χρόνο, ανεξάρτητα από το αν μπορούν να τα λυθούν σε πολυωνυμικό χρόνο, θα πρέπει να είναι σαφές ότι το P είναι ένα υποσύνολο του NP: εάν ένα πρόβλημα σε πολυωνυμικό χρόνο μπορεί να επιλυθεί, τότε σίγουρα μπορεί να επαληθευθεί μια προτεινόμενη απάντηση στο πρόβλημα σε πολυωνυμικό χρόνο.

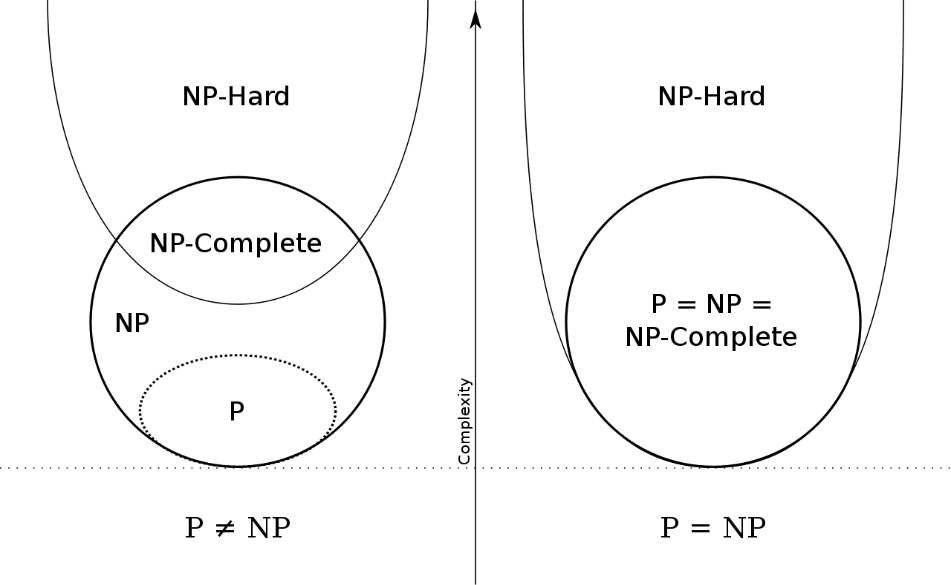
### 1.1.3 Class NP-Hard (Nondeterministic Polynomial-time hard)

Ένα πρόβλημα μπορεί να θεωρηθεί NP-Hard, εαν είναι τουλάχιστον το ίδιο δύσκολο με τα πιο δύσκολα προβλήματα NP. Πιο συγκεκριμένα, ένα πρόβλημα H είναι NP-Hard όταν κάθε πρόβλημα L στο NP μπορεί να "μειωθεί" ή να μετατραπεί σε πρόβλημα H σε πολυωνυμικό χρόνο. Ως αποτέλεσμα, αν έχει βρεθεί ένας αλγόριθμος πολυωνυμικού χρόνου για να λύσει οποιοδήποτε πρόβλημα NP-Hard, αυτό θα είχε ως αποτέλεσμα αλγόριθμους πολυωνυμικού χρόνου για όλα τα προβλήματα στο NP.

### 1.1.4 Class NP-Complete

Η τελευταία κατηγορία υπολογιστικών προβλημάτων που θα συζητήσουμε είναι το NP-Complete, το οποίο είναι απλώς η διασταύρωση μεταξύ NP και NP-Hard. Με άλλα λόγια, ένα πρόβλημα NP-Hard θεωρείται NP-Complete αν μπορεί να επαληθευτεί σε πολυωνυμικό χρόνο δηλαδή είναι επίσης NP).

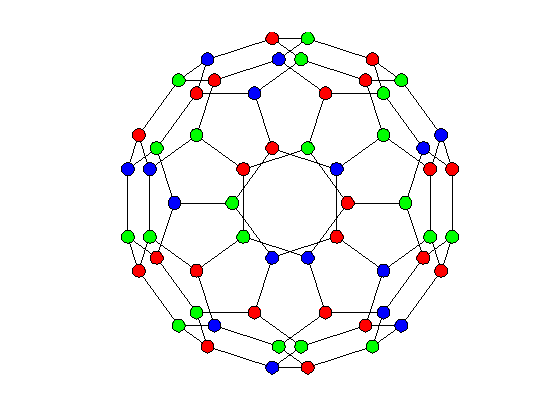
Ένα ενδιαφέρον πρόβλημα NP-Complete είναι το "Boolean Satisfiability Problem", το οποίο αποτελεί τη βάση της σύγχρονης κρυπτογράφησης. Όταν κρυπτογραφούμε ευαίσθητα δεδομένα, αισθανόμαστε ασφαλείς διότι, χωρίς να γνωρίζουμε το κλειδί κρυπτογράφησης, ένα κακόβουλο άτομο θα πρέπει να λύσει το "Boolean Satisfiability Problem" για να είναι σε θέση να αποκρυπτογραφήσει τα δεδομένα μας, κάτι που είναι ανέφικτο λόγω της σκληρότητας του προβλήματος.



Εικόνα 1. Διάγραμμα Συσχετίσεων

## 1.2 Ορισμός προβλήματος χρωματισμού γραφήματος

Ένα μη κατευθυνόμενο γράφημα αποτελείται από ένα σύνολο κορυφών και ένα σύνολο μη ταξινομημένων ζευγών κορυφών που ονομάζονται ακμές . Το πρόβλημα χρωματισμού γραφήματος (**G**raph **C**oloring **P**roblem) αναφέρεται στην εκχώρηση χρώματος σε κάθε κορυφή του γραφήματος G, με τέτοιο τρόπο ώστε δύο γειτονικές κορυφές να έχουν διαφορετικά χρώματα και ο αριθμός των διαφορετικών χρωμάτων που χρησιμοποιούνται να είναι όσο το δυνατόν μικρότερος. Ο ελάχιστος αριθμός χρωμάτων που απαιτούνται για το χρώμα ονομάζεται χρωματικός αριθμός και υποδηλώνεται με . Η απόφαση για το εάν ένα γράφημα μπορεί να χρωματιστεί χρησιμοποιώντας το πολύ k χρώματα αναφέρεται ως πρόβλημα χρωματικότητας (k-colorability). Το GCP είναι ένα από τα πιο μελετημένα προβλήματα συνδυαστικής βελτιστοποίησης NP-hard. Είναι ενδιαφέρον όχι μόνο για τη μελέτη της υπολογιστικής πολυπλοκότητας αλλά και για πρακτικές του εφαρμογές. Πολλά προβλήματα του πραγματικού κόσμου μπορούν φυσικά να προσδιοριστούν ως GCP και να επιλυθούν χρησιμοποιώντας έναν αλγόριθμο GCP.



Εικόνα 2. Graph Coloring

Oι αλγόριθμοι GCP εντάσσονται σε δύο κατηγορίες: στους ακριβείς αλγόριθμους (exact algorithms) και στους κατά προσέγγιση αλγόριθμους (approximate algorithms). Οι κατά προσέγγιση αλγόριθμοι περιλαμβάνουν: τη μέθοδο κατασκευής (construction method), την τοπική αναζήτηση (local search), τους εξελικτικούς αλγόριθμους με βάση τους πληθυσμούς (evolutionary algorithms based on populations), τους υβριδικούς αλγόριθμους (hybrid algorithms) και άλλες ευρετικές μεθόδους (heuristic methods). Eνα πρόβλημα GCP που κωδικοποιεί ένα πραγματικό πρόβλημα υλοποίησης μιας εφαρμογής είναι συχνά πολύ μεγάλο και ενδέχεται να μην μπορεί να ικανοποιήσει τις συνθήκες της κατηγορίας ακριβείς αλγόριθμοι, επειδή πρέπει να απαριθμήσουν τον χώρο αναζήτησης ενός προβλήματος και δυσκολεύονται στην επίλυση μεγάλων περιπτώσεων δηλαδή περιπτώσεων με μεγάλη πολυπλοκότητα. Ωστόσο, οι κατά προσέγγιση αλγόριθμοι συνήθως δεν είναι σε θέση να αποδείξουν τη βέλτιστη λύση. Έτσι, οι ακριβείς αλγόριθμοι παραμένουν πολύ χρήσιμοι και απαραίτητοι. Οι περιορισμοί σε ένα GCP είναι πολύ εύκολο να εκφραστούν και να κατανοηθούν, λέγοντας ότι σε δύο γειτονικές κορυφές δεν μπορεί να αποδοθεί το ίδιο χρώμα. Ωστόσο, αυτοί οι απλοί ρητοί περιορισμοί μεταξύ γειτονικών κορυφών συνεπάγονται πολλούς πολύπλοκους περιορισμούς μεταξύ μη γειτονικών κορυφών. Η συγκεκριμένη κατηγορία προβλημάτων αναγνωρίζεται ως NP-Hard πρόβλημα.

# Κεφάλαιο 2. Προσεγγίσεις επίλυσης

## 2.1 Δεδομένα προβλήματος

Το πρόβλημα χρονοπρογραμματισμού εξετάσεων αφορά φοιτητές που έχουν πραγματοποιήσει εγγραφές σε εξετάσεις μαθημάτων. Για κάθε εξέταση διατίθεται μια λίστα από φοιτητές και κάθε φοιτητής μπορεί να είναι εγγεγραμμένος σε μια ή περισσότερες εξετάσεις. Κάθε εξέταση θα πρέπει να τοποθετηθεί σε μια περίοδο εξέτασης και η λύση του προβλήματος συνίσταται στην ανάθεση όλων των εξετάσεων στο μικρότερο δυνατό αριθμό περιόδων έτσι ώστε να μην υπάρχουν συγκρούσεις, δηλαδή να μην υπάρχουν φοιτητές που θα έπρεπε να συμμετάσχουν σε εξετάσεις σε περισσότερα του ενός μαθήματα στην ίδια περίοδο. Ως δεδομένα του προβλήματος θα χρησιμοποιηθούν τα δεδομένα του προβλήματος χρονοπρογραμματισμού εξετάσεων Toronto τα οποία είναι διαθέσιμα προς μεταφόρτωση στη διεύθυνση <https://github.com/chgogos/datasets/blob/main/UETT/toronto.zip>. Τα αρχεία δεδομένων (κατάληξη .stu) διαθέτουν για κάθε σπουδαστή μια γραμμή που περιέχει τους αριθμούς των μαθημάτων στα οποία είναι εγγεγραμμένος χωρισμένους μεταξύ τους με κενά. Η πρώτη γραμμή του αρχείου αντιστοιχεί στον πρώτο σπουδαστή, η δεύτερη γραμμή στο δεύτερο σπουδαστή κ.ο.κ.

## 2.2 Στατιστικά στοιχεία

Θα εμφανίζονται τα ακόλουθα στατιστικά στοιχεία για καθένα από τα 13 προβλήματα του Toronto dataset:

1. Αριθμός κορυφών.

2. Πυκνότητα. Για τον υπολογισμό της πυκνότητας θα πρέπει να κατασκευαστεί ο πίνακας συγκρούσεων. Η πυκνότητα συγκρούσεων υπολογίζεται διαιρώντας τον αριθμό των στοιχείων του πίνακα συγκρούσεων που έχουν την τιμή 1 με το συνολικό πλήθος των στοιχείων του πίνακα.

3. Για τους βαθμούς (degrees) των κορυφών η ελάχιστη τιμή (min), η διάμεσος τιμή (median), η μέγιστη τιμή (max), η μέση τιμή (mean) καθώς και ο συντελεστής διακύμανσης (CV=coefficient of variation) που ορίζεται ως η τυπική απόκλιση προς τη μέση τιμή.

### 2.2.1 Πίνακας στατιστικών στοιχείων

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | |V| | Density | Max,Med,Min | Vertex Degree  Mean | CV |
| hec-s-92 | 81 | 0.415485 | 62,33,9 | 33.6543 | 36.3264 |
| sta-f-83 | 139 | 0.142953 | 61,16,7 | 19.8705 | 67.3648 |
| yor-f-83 | 181 | 0.287293 | 117,51,7 | 52 | 35.2268 |
| ute-s-92 | 184 | 0.0844754 | 58,13,2 | 15.5435 | 69.1351 |
| ear-f-83 | 190 | 0.26554 | 134,45,4 | 50.4526 | 56.1132 |
| tre-s-92 | 261 | 0.180003 | 145,45,0 | 46.9808 | 59.6187 |
| lse-f-91 | 381 | 0.0624272 | 134,16,0 | 23.7848 | 93.156 |
| kfu-s-93 | 461 | 0.055458 | 247,18,0 | 25.5662 | 119.987 |
| rye-s-93 | 486 | 0.075124 | 274,24,0 | 36.5103 | 111.876 |
| car-f-92 | 543 | 0.137732 | 381,64,0 | 74.7882 | 75.3453 |
| uta-s-92 | 622 | 0.125355 | 303,65,1 | 77.9711 | 73.671 |
| car-s-91 | 682 | 0.128198 | 472,77,0 | 87.4311 | 70.9621 |
| pur-s-93 | 2419 | 0.0294831 | 857,47,0 | 71.3196 | 129.506 |

# Κεφάλαιο 3. Επίλυση προβλήματος

Η επίλυση του προβλήματος θα πραγματοποιηθεί με παρουσία συγκεκριμένων αλγορίθμων οι οποίοι θα αναλυθούν στη συνέχεια.

## 3.1 First Fit (FF)

Ο απλούστερος χρωματισμός ενός γραφήματος γίνεται με τον αλγόριθμο First-Fit (FF) μερικές φορές ονομαζόμενος ως «greedy algorithm». Δεδομένου (G, <) ως είσοδο, το FF λειτουργεί λαμβάνοντας τις κορυφές του γραφήματος G, μία κορυφή τη φορά, με τη δεδομένη σειρά και εκχωρεί το μικρότερο δυνατό ακέραιο από το ως το χρώμα στην κορυφή δηλαδή το μικρότερο χρώμα που δεν έχει ακόμη αντιστοιχιστεί σε καμία κορυφή δίπλα στο vi, μεταξύ των προηγουμένως χρωματισμένων κορυφών. Εάν οι κορυφές του G θεωρούνται ως μια ιδανική ακολουθία τότε Για να κατασκευασθεί μια τέτοια ακολουθία, πρέπει πρώτα να βρεθεί ένας βέλτιστος χρωματισμός του G και στη συνέχεια να τοποθετηθούν όλες οι κορυφές με το ίδιο χρώμα σε διαδοχικές θέσεις στην ακολουθία. Πολλοί ερευνητές έχουν μελετήσει εκτενώς αλγόριθμους χρωματισμού, το μεγαλύτερο μέρος της δουλειάς τους αφιερώνεται στην απόδειξη των ανώτερων ορίων για το , δηλαδή η χειρότερη περίπτωση του αλγορίθμου χρωματισμού FF.

## 3.2 Degree of Saturation (DSatur)

Το DSatur είναι ένας αλγόριθμος χρωματισμού γραφημάτων που προτάθηκε από τον Daniel Brélaz το 1979. Ομοίως με τον άπληστο αλγόριθμο (FF), το DSatur χρωματίζει τις κορυφές ενός γραφήματος τη μια μετά την άλλη, αναλώνοντας έτσι ένα προηγουμένως αχρησιμοποίητο χρώμα όταν χρειάζεται. Μόλις χρωματιστεί μια νέα κορυφή, ο αλγόριθμος καθορίζει ποια από τις υπόλοιπες άχρωμες κορυφές έχει τον υψηλότερο αριθμό χρωμάτων στη γειτονιά του και χρωματίζει αυτήν την κορυφή στη συνέχεια. Ο Brélaz ορίζει αυτόν τον αριθμό ως τον βαθμό κορεσμού μιας δεδομένης κορυφής. Η συστολή του βαθμού κορεσμού σχηματίζει το όνομα του αλγορίθμου. Το DSatur είναι ένας ευρετικός αλγόριθμος χρωματισμού γραφημάτων, παράγει όμως ακριβή αποτελέσματα για διμερή, κύκλο και γραφήματα τροχών. Το DSatur έχει επίσης την ονομασία κορεσμός LF.

## 3.3 Recursive Large First (RLF)

Ο αλγόριθμος Recursive Largest First (RLF) είναι μια από τις πιο δημοφιλείς άπληστες ευρετικές μεθόδους για προβλήματα χρωματισμού. Δημιουργεί διαδοχικά κλάσεις χρωμάτων με βάση τις άπληστες επιλογές. Πιο συγκεκριμένα, η πρώτη κορυφή που τοποθετείται σε μια έγχρωμη κλάση C, είναι μία κλάση με μέγιστο αριθμό αυτό των άχρωμων γειτόνων. Οι επόμενες κορυφές που τοποθετούνται στο C επιλέγονται έτσι ώστε να έχουν τόσους άχρωμους γείτονες που να μην μπορούν να τοποθετηθούν στο C. Αυτές οι άπληστες επιλογές μπορούν να έχουν σημαντικό αντίκτυπο στην απόδοση του αλγορίθμου. Ο αλγόριθμος Recursive Largest First (RLF) προτάθηκε το 1979 από τον F. Leighton. Σε γενικές γραμμές, αυτός ο αλγόριθμος δημιουργεί μια ακολουθία σταθερών συνόλων, όπου το καθένα αντιστοιχεί σε μια κλάση χρωμάτων. Υποθετικά ότι το C είναι μια κλάση χρωμάτων που θα κατασκευαστεί, το U υποδηλώνει το σύνολο των άχρωμων κορυφών και το W το σύνολο (αρχικά κενό) άχρωμων κορυφών με τουλάχιστον έναν γείτονα στο C. Κάθε φορά που επιλέγεται μια κορυφή στο U να μετακινηθεί στο C, όλοι οι γείτονές του στο U μετακινούνται από το U στο W. Η πρώτη κορυφή που θα συμπεριληφθεί στο C είναι αυτή με τον μεγαλύτερο αριθμό γειτόνων στο U. Το υπόλοιπο C είναι κατασκευασμένο ως εξής: ενώ το U δεν είναι άδειο, η επόμενη κορυφή που θα μετακινηθεί από το U στο C είναι εκείνη που έχει τον μεγαλύτερο αριθμό γειτόνων στο W. Για μια κορυφή , δηλώνουμε το και το τον αριθμό των γειτόνων του σε U και W, αντίστοιχα. Επίσης, όταν το v είναι η πρώτη κορυφή που τοποθετείται σε μια κλάση χρωμάτων, δηλώνουμε το Cv ως την κλάση χρωμάτων που την περιέχει.

## 3.4 Backtracking Dsatur

O αλγόριθμος backtracking DSATUR είναι η τέταρτη εφαρμογή αλγορίθμου της παρούσας εργασίας, με κύριο χαρακτηριστικό την εκχώρηση χρωμάτων σε όλες τις κορυφές του γραφήματος με σκοπό τον χρωματισμό του. Δεδομένου ενός μη κατευθυνόμενου γραφήματος και ενός ακεραίου αριθμού m, καθορίζεται εάν το γράφημα μπορεί να χρωματιστεί με το πολύ m χρώματα, έτσι ώστε να μην χρωματίζονται δύο γειτονικές κορυφές του γραφήματος με το ίδιο χρώμα. Εδώ ο χρωματισμός ενός γραφήματος σημαίνει την εκχώρηση χρωμάτων σε όλες τις κορυφές. Η αρχική σειρά των κορυφών καθορίζεται από τον αλγόριθμο DSATUR. Ως εκ τούτου, οι κορυφές με τα λιγότερα διαθέσιμα χρώματα χρωματίζονται πρώτα, με τους δεσμούς να σπάζουν με βάση των βαθμών (degrees) και οι περαιτέρω δεσμοί να σπάζουν τυχαία. Αφού εκτελεστεί ένα βήμα προς τα πίσω, οι κορυφές αναδιατάσσονται δυναμικά, έτσι ώστε η επόμενη κορυφή που πρέπει να χρωματιστεί να είναι και αυτή με τα λιγότερα διαθέσιμα χρώματα. Εάν η κορυφή δεν διαθέτει εφικτά χρώματα, ο αλγόριθμος παίρνει ένα ακόμη βήμα προς τα πίσω (Backtracking).

# Κεφάλαιο 4. Αποτελέσματα

## 4.1 Χρωματισμός γραφημάτων με τον αλγόριθμο First Fit

Στον παρακάτω πίνακα διακρίνεται αναλυτικά το δείγμα των δεδομένων ονομαστικά καθώς και το πόσα χρώματα χρησιμοποιήθηκαν για το καθένα με την εκτέλεση του greedy αλγορίθμου First Fit.

|  |  |
| --- | --- |
| **Name** | **Colors Used** |
| hec-s-92 | 22 |
| sta-f-83 | 13 |
| yor-f-83 | 27 |
| ute-s-92 | 13 |
| ear-f-83 | 29 |
| tre-s-92 | 29 |
| lse-f-91 | 22 |
| kfu-s-93 | 25 |
| rye-s-93 | 28 |
| car-f-92 | 44 |
| uta-s-92 | 43 |
| car-s-91 | 48 |
| pur-s-93 | 54 |

# ΒΙΒΛΙΟΓΡΑΦΙΑ

Mourchid , A., Alain , H., & Martine , B. (2015, November 2). A new efficient RLF-like Algorithm for the Vertex Coloring Problem.

Niema , M., & Liz , I. (2016). *Data Structures.*

Nikolopoulos, S., & Papadopoulos, C. (2000, April 19). On the performance of the first-fit coloring algorithm. *Information Processing Letters 75*, σσ. 265-273.

R. M. R. Lewis (2015, October 27). A Guide to Graph ColouringAlgorithms and Applications. *Algorithm Case Studies*, pp. 88-89.

*Wikipedia*. (2020, October 6). Ανάκτηση από Wikipedia: https://en.wikipedia.org/wiki/DSatur

Zhaoyang, Z., Chu-Min, L., Chong, H., & Ruchu, X. (2014, November). An exact algorithm with learning for the graph coloring problem. *Computers & Operations Research*, pp. 282-301.