

## **ΤΕΧΝΙΚΗ ΑΝΑΦΟΡΑ**

ΣΤΕΡΓΙΟΣ ΠΑΝΑΓΙΩΤΗΣ ΜΑΡΙΟΣ ΑΜ:00106

ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ ΣΤΟ ΜΑΘΗΜΑ ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ  
ΠΟΛΥΠΛΟΚΟΤΗΤΑ

## **Περιεχόμενα**

1.ΠΕΡΙΛΗΨΗ	3
2.ΔΕΔΟΜΕΝΑ ΕΡΓΑΣΙΑΣ	4
3.ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ	4
3.1 ΧΡΩΜΑΤΙΣΜΟΣ ΓΡΑΦΗΜΑΤΟΣ	5
4.ΕΠΙΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ	6
4.1 GREY COLORING	6
5.ΠΕΡΙΒΑΛΛΟΝ ΕΡΓΑΣΙΑΣ	7

## **1.ΠΕΡΙΛΗΨΗ**

Στην συγκεκριμένη εργασία εξάμηνου καλούμαστε να υλοποιήσουμε τον κώδικα για την επίλυση ενός προβλήματος για εξετάσεις ακαδημαϊκού εξαμήνου ώστε το πρόγραμμα εξετάσεων να είναι αποδοτικό για όλους τους σπουδαστές. Η εργασία γίνεται στο περιβάλλον intellij idea με γλώσσα προγραμματισμού C++. Καλούμαστε να δώσουμε λύση σε ένα πρόβλημα στο οποίο θα βγαίνουν αποτελέσματα και το πρόγραμμα εξεταστικής με βάση τους σπουδαστές και τα μαθήματα που καλούνται να δώσουν. Το κάθε μάθημα δεν θα πρέπει να πέφτει σε ίδια ώρα και ημερομηνία με μάθημα που θα πρέπει να δώσει ο σπουδαστής. Θα χρησιμοποιηθούν κώδικες χρωματισμού γράφων ώστε να βρεθεί μία αξιόλογη λύση στο πρόβλημα.

## 2.ΔΕΔΟΜΕΝΑ ΕΡΓΑΣΙΑΣ

Η πρώτη γραμμή του αρχείου αντιστοιχεί στον πρώτο σπουδαστή, η δεύτερη γραμμή στο δεύτερο σπουδαστή κ.ο.κ. Για παράδειγμα το αρχείο car-f-92.stu περιέχει 18419 σειρές δεδομένων και ξεκινά με τις ακόλουθες σειρές:

0170  
0156  
0281  
0006  
0154 0156  
0383  
0534 0535 0536

... που σημαίνουν ότι ο φοιτητής 1 έχει εγγραφεί στο μάθημα 0170, ο φοιτητής 2 έχει εγγραφεί στο μάθημα 0156, ο φοιτητής 3 έχει εγγραφεί στο μάθημα 0281, ο φοιτητής 4 έχει εγγραφεί στο μάθημα 0006, ο φοιτητής 5 στα μαθήματα 0154 0156 κ.ο.κ.

## 3.ΓΕΝΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ

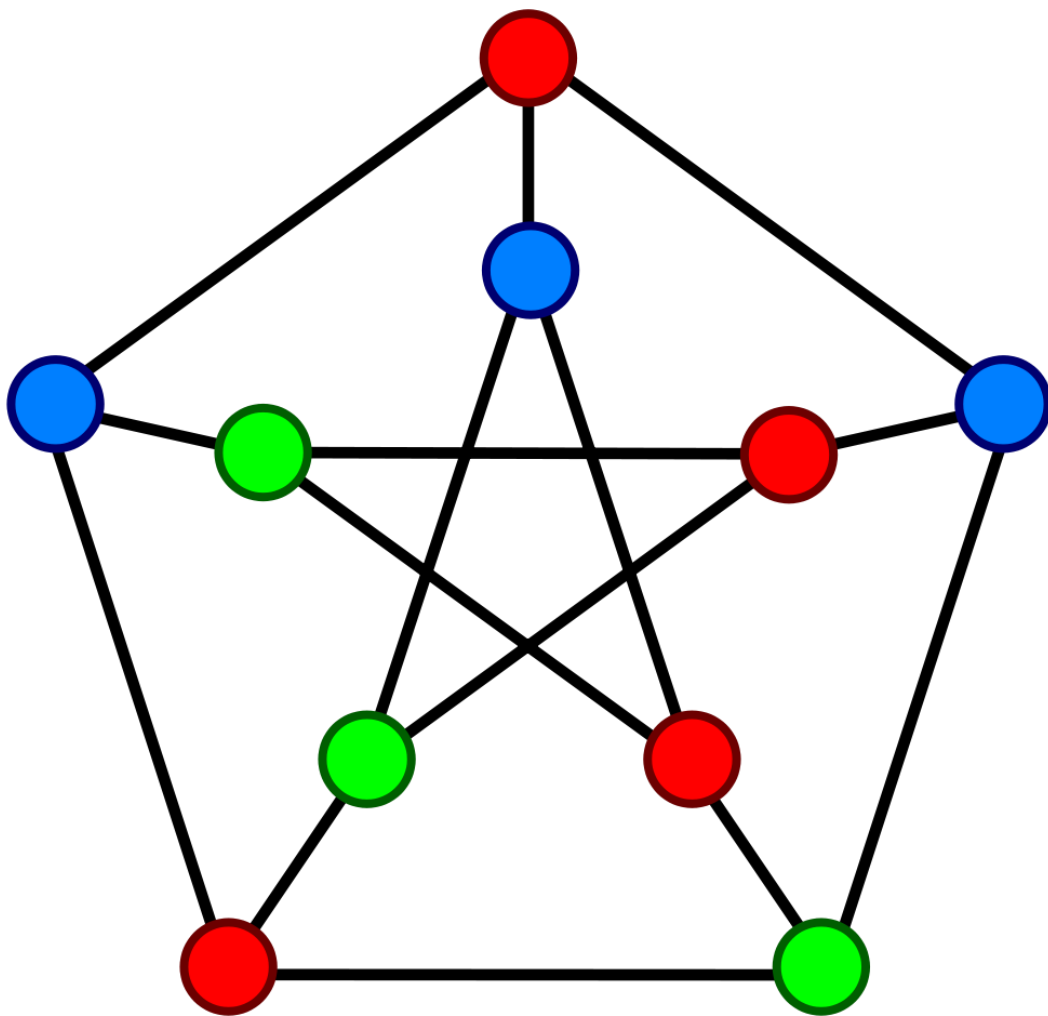
Το πρόβλημα που καλούμαστε να υλοποιήσουμε είναι της κατηγορίας NPC (np – complete) πιο συγκεκριμένα :

1. Είναι ένα πρόβλημα για το οποίο η ορθότητα κάθε λύσης μπορεί να επαληθευτεί γρήγορα με τη χρήση ενός αλγόριθμου brute force δοκιμάζοντας όλες τις πιθανές λύσεις στο πρόβλημα.

2. Τέτοιου είδους προβλήματα μπορούν να χρησιμοποιηθούν για να κάνουν προσομοίωση άλλων προβλημάτων και να ελέγξουμε γρήγορα ότι η λύση είναι αληθής.

### 3.1 ΧΡΩΜΑΤΙΣΜΟΣ ΓΡΑΦΗΜΑΤΟΣ

Είναι ένας τρόπος να χρωματίσουμε τις κορυφές ενός γραφήματος έτσι ώστε δύο γειτονικές κορυφές να μην έχουν το ίδιο χρώμα, αυτό ονομάζεται χρωματισμός κορυφής. Υπάρχει και ο χρωματισμός άκρης ο οποίος αποδίδει ένα χρώμα σε κάθε άκρη έτσι ώστε δύο γειτονικές άκρες να μην έχουν το ίδιο χρώμα.



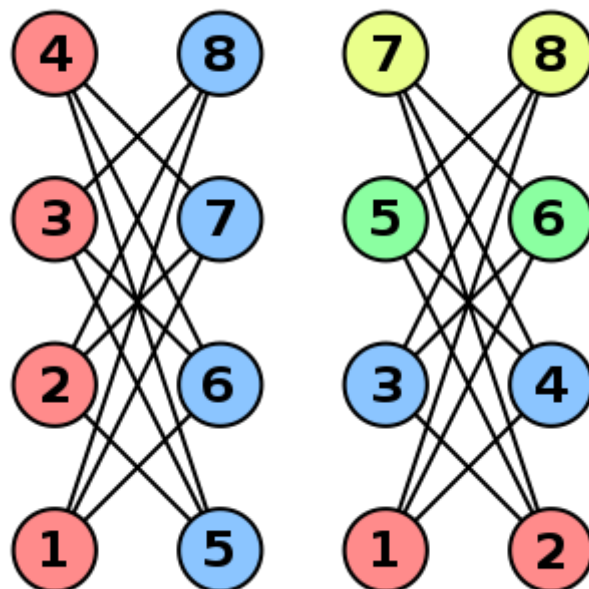
Ο χρωματισμός **Vertex** είναι το πιο κοινό πρόβλημα χρωματισμού γραφήματος. Το πρόβλημα είναι, δεδομένου ότι τα χρώματα  $m$ , βρίσκουν έναν τρόπο χρωματισμού των κορυφών ενός γραφήματος έτσι ώστε να μην χρωματίζονται δύο παρακείμενες κορυφές χρησιμοποιώντας το ίδιο χρώμα. Τα άλλα προβλήματα χρωματισμού γραφήματος, όπως ο χρωματισμός άκρων (Δεν υπάρχει κορυφή σε δύο άκρες του ίδιου χρώματος) και ο χρωματισμός

προσώπου ( χρωματισμός γεωγραφικού χάρτη) μπορούν να μετατραπούν σε χρωματισμό κορυφής.

## 4.Επίλυση προβλήματος

### 4.1 Greedy Coloring

Χρησιμοποίησα τον αλγόριθμο greedy coloring για την λύση του προβλήματος σύμφωνα και με όσα είχαμε δει στο μάθημα. Ο greedy coloring για μια συγκεκριμένη σειρά κορυφής μπορεί να υπολογιστεί από έναν αλγόριθμο που εκτελείται σε γραμμικό χρόνο . Ο αλγόριθμος επεξεργάζεται τις κορυφές στη δεδομένη σειρά, εκχωρώντας ένα χρώμα σε κάθε μία καθώς επεξεργάζεται. Τα χρώματα μπορεί να αντιπροσωπεύονται από τους αριθμούς και σε κάθε κορυφή δίνεται το χρώμα με τον μικρότερο αριθμό που δεν χρησιμοποιείται ήδη από έναν από τους γείτονές του. Για να βρείτε το μικρότερο διαθέσιμο χρώμα, μπορείτε να χρησιμοποιήσετε έναν πίνακα για να μετρήσετε τον αριθμό των γειτόνων κάθε χρώματος (ή εναλλακτικά, για να αναπαραστήσετε το σύνολο των χρωμάτων των γειτόνων) και, στη συνέχεια, να σαρώσετε τον πίνακα για να βρείτε τον δείκτη του πρώτου μηδέν.



## **5. Περιβάλλον Εργασίας**

Έγραψα την εργασία στο IntelliJ IDEA καθώς είναι ένα πολύ εύκολο πρόγραμμα με γρήγορη απόκριση και χωρίς προβλήματα, γνώριζα τον περιβάλλον εργασίας και από προηγούμενα μαθήματα του μεταπτυχιακού προγράμματος. Τέλος όλοι οι κώδικες είναι γραμμένοι σε Java.