**Τεχνητή Νοημοσύνη**

**Χειμερινό Εξάμηνο**

**Εργασία 1**

**Δανοπούλου Έμυ 3170033**

**Μπαλή Νίκη 3170114**

**Χαβιατζή Ελένη 3170172**

**Γενικά**

Το πρόγραμμα δέχεται ως εισόδους δύο αρχεία txt (lessons.txt και teachers.txt) στα οποία βρίσκονται τα δεδομένα για τα μαθήματα (κωδικός, τίτλος μαθήματος, τάξη, εβδομαδιαίες ώρες διδασκαλίας) και για τους καθηγητές (κωδικός, όνομα καθηγητή, κωδικοί μαθημάτων που μπορούν να διδαχθούν από τον συγκεκριμένο καθηγητή, ημερήσιες ώρες διδασκαλίας, εβδομαδιαίες ώρες διδασκαλίας) αντίστοιχα.

Χρησιμοποιήθηκε γενετικός αλγόριθμος με πληθυσμό 10, πιθανότητα μετάλλαξης 0.10, ελάχιστο κόστος 20900 και άνω όριο βημάτων 1000.

Το πρόγραμμα αρχικά διαβάζει τα αρχεία των μαθημάτων και των καθηγητών, δημιουργεί ένα χρωμόσωμα (ώστε να ξεκινήσει ο αλγόριθμος με ένα γεμάτο πρόγραμμα) και, αφού εφαρμοστεί ο γενετικός αλγόριθμος, τυπώνει ένα αρχείο txt (schedule.txt) με το τελικό πρόγραμμα.

**Class ReadFiles**

**Method readfiles (String data)**

Η μέθοδος παίρνει ως όρισμα ένα String που είναι το όνομα του αρχείου που θα διαβαστεί.

Για το διάβασμα των αρχείων χρησιμοποιείται ο BufferesReader. Αρχικά ανοίγει το αρχείο και μετράει τις γραμμές του. Αν το αρχείο είναι το αρχείο των καθηγητών, φτιάχνει έναν πίνακα με διαστάσεις τις γραμμές του αρχείου (που είναι ο αριθμός των καθηγητών) επί 5 (γιατί τα πεδία για τους καθηγητές είναι 5), αλλιώς το αρχείο είναι εκείνο των μαθημάτων και φτιάχνει ένα πίνακα με διαστάσεις τις γραμμές του αρχείου (που είναι ο αριθμός των μαθημάτων) επί 4 (γιατί τα πεδία για τους καθηγητές είναι 4) και κλείνει το αρχείο.

Όταν δημιουργηθεί ο πίνακας, το αρχείο ανοίγει ξανά και τοποθετούνται τα δεδομένα από τα αρχεία στους πίνακες με τη βοήθεια του StringTokenizer, αφού γίνει έλεγχος για το αρχείο. Το αρχείο κλείνει και η μέθοδος επιστρέφει τον πίνακα που δημιουργήθηκε.

**Class Gene**

Αποτελεί την αναπαράσταση ενός γονιδίου του χρωμοσώματος και περιλαμβάνει δύο πεδία, ένα μάθημα (lesson) και έναν καθηγητή (teacher).

Υπάρχουν δύο constructors. Ο Gene(int lesson, int teacher), ο οποίος αρχικοποιεί τον κωδικό του μαθήματος και του καθηγητή ανάλογα με τα ορίσματά του, και ο Gene() που αρχικοποιεί τον κωδικό του μαθήματος και του καθηγητή με -1, αναπαριστώντας το κενό στο πρόγραμμα.

**Method getTeacher()**

Επιστρέφει τον καθηγητή.

**Method getLesson()**

Επιστρέφει το μάθημα.

**Method setTeacher(int t)**

Αλλάζει τον καθηγητή σε t.

**Method setLesson(int l)**

Αλλάζει το μάθημα σε l.

**Class Chromosome**

Αποτελεί την αναπαράσταση ενός χρωμοσώματος και περιλαμβάνει όλες τις μεθόδους που χρησιμοποιούνται από το χρωμόσωμα. Ένα χρωμόσωμα είναι ένα πρόγραμμα γυμνασίου.

Υπάρχουν δύο constructors. Ο Chromosome(Object[ ][ ] lessons, Object[ ][ ] teachers) αρχικοποιεί ένα χρωμόσωμα ως ένα δισδιάστατο πίνακα genes 9 (συνολικά 9 τμήματα με τη σειρά 1η γραμμή->Α1, 2η γραμμή->Α2 κ.ο.κ.) επί 35 (ώρες της εβδομάδας). Μέσα σε κάθε θέση του genes υπάρχει ένα αντικείμενο τύπου Gene, το οποίο περιλαμβάνει ένα μάθημα και έναν καθηγητή. Ο genes αρχικοποιείται τυχαία με μαθήματα και καθηγητές. Έπειτα, με τη βοήθεια ενός for loop βλέπουμε πόσες ώρες την εβδομάδα γίνονται μαθήματα συνολικά σε όλα τα τμήματα και βάζουμε τυχαία τα κενά που χρειάζονται στο genes ώστε να υπάρχουν κενά στο πρόγραμμα. Τέλος, υπολογίζουμε με τη συνάρτηση καταλληλότητας το κόστος του χρωμοσώματος. Ο Chromosome(Gene [ ][ ] genes, Object[ ][ ] lessons, Object[ ][ ] teachers) δημιουργεί ένα αντίγραφο του γονιδίου που παίρνει ως όρισμα.

**Method getGenes()**

Επιστρέφει το χρωμόσωμα.

**Method getFitness()**

Επιστρέφει το κόστος του χρωμοσώματος.

**Method setGenes(Gene[ ][ ] genes)**

Αντιγράφει το χρωμόσωμα του ορίσματος στο χρωμόσωμα.

**Method setFitness(int fitness)**

Αλλάζει το κόστος σε fitness.

**Method calculateFitness()**

Είναι η συνάρτηση καταλληλότητας (ευρετική) και βρίσκει το κόστος του χρωμοσώματος μετρώντας πόσοι περιορισμοί δεν παραβιάζονται με τη μεταβλητή non\_violations. Χωρίζεται σε μικρότερες «υποενότητες».

//LESSON SHOULD MATCH CLASS

Για κάθε τάξη ξεχωριστά ελέγχει αν το μάθημα, αν δεν είναι το κενό, που υπάρχει είναι στη σωστή τάξη. Για κάθε μάθημα διατρέχει τον πίνακα lessons και ψάχνει μαθήματα της αντίστοιχης τάξης. Αν ο κωδικός του μαθήματος στον genes είναι ίδιος με αυτόν από τον lessons, που είναι σίγουρο ότι είναι της σωστής τάξης, τότε το μάθημα είναι στη σωστή τάξη, προσθέτει στη μεταβλητή non\_violations και σταματά τη διάσχιση στον lessons, αλλιώς συνεχίζει την διάσχιση. Αν δεν βρεθεί το μάθημα, σημαίνει ότι δεν είναι στη σωστή τάξη.

//TEACHER SHOULD MATCH LESSON

Για κάθε θέση του genes ελέγχει αν το συγκεκριμένο μάθημα διδάσκεται από τον σωστό καθηγητή. Βρίσκει στον πίνακα teachers τον κωδικό του καθηγητή και με τη βοήθεια της μεθόδου getTeachableLessons ελέγχει αν το συγκεκριμένο μάθημα μπορεί να διδαχθεί από τον συγκεκριμένο καθηγητή. Αν ναι, προσθέτει στη non\_violations.

//TOTAL HOURS FOR EACH LESSON SHOULD BE EXACTLY AS MANY AS LESSONS.TXT SAYS

Για κάθε μάθημα του πίνακα lessons ελέγχει αν το συγκεκριμένο μάθημα μέσα στον genes υπάρχει όσες φορές λέει ο lessons. Αρχικά κρατάει στη μεταβλητή total\_hours τις ώρες που το συγκεκριμένο μάθημα υπάρχει στο πρόγραμμα και μετά τις συγκρίνει με τις ώρες που θα έπρεπε να είναι. Αν είναι οι ίδιες, προσθέτει στη non\_violations.

//THERE SHOULD BE NO EMPTY SLOTS IN A CLASS SCHEDULE

Για κάθε τμήμα χωρίζει τον genes σε 5 κομμάτια, ένα για κάθε μέρα. Σε μια μεταβλητή previous\_spot κρατάει την προηγούμενη θέση που βρέθηκε κενό στη συγκεκριμένη μέρα. Αν όταν τελειώσει η μέρα δεν υπάρχει κενό, τότε προσθέτει στη non\_violations.

//NO TEACHER SHOULD HAVE MORE THAN 2 CONTINUOUS TEACHING HOURS

Αν ένας καθηγητής που διδάσκει σε μια συγκεκριμένη ώρα δεν έχει μάθημα και την επόμενη και τη μεθεπόμενη ώρα, τότε δεν έχει δύο συνεχόμενες ώρες διδασκαλίας και προσθέτει στη non\_violations.

//AN EVENLY DISTRIBUTED SCHEDULE WITHOUT BIG DIFFERENCES EACH DAY FOR EACH CLASS (SIMILAR CUMULATIVE DAILY HOURS)

Βρίσκει το μέσο όρο των ωρών διδασκαλίας ημερησίως για κάθε τμήμα. Αν για κάθε μέρα οι ώρες διδασκαλίας είναι ±1 (δηλαδή είναι περίπου ίδιες) τότε προσθέτει στη non\_violations.

//SUBJECTS SHOULD BE DISTRIBUTED EVENLY ACROSS DAYS

Ελέγχει αν για κάθε τμήμα όλες οι ώρες ενός μαθήματος διδάσκονται μέσα σε μία μέρα.

//TEACHERS SHOULD HAVE SIMILAR WORKLOADS

Βρίσκει το μέσο όρο των ωρών διδασκαλίας για κάθε καθηγητή. Αν οι ώρες διδασκαλίας κάθε καθηγητή είναι ±2 (δηλαδή είναι περίπου ίδιες) τότε προσθέτει στη non\_violations.

//TEACHERS SHOULD NOT TEACH MORE HOURS IN A WEEK THAN TEACHERS.TXT SAYS

Για κάθε καθηγητή του πίνακα teachers ελέγχει αν ο συγκεκριμένος καθηγητής μέσα στον genes υπάρχει όσες φορές λέει ο teachers. Αρχικά κρατάει στη μεταβλητή total\_hours τις ώρες που ο συγκεκριμένος καθηγητής υπάρχει στο πρόγραμμα και μετά τις συγκρίνει με τις ώρες που θα έπρεπε να είναι. Αν είναι οι ίδιες, προσθέτει στη non\_violations.

**Method mutate()**

Αλλάζει τυχαία την τιμή μιας τυχαίας θέσης του πίνακα genes. Διαλέγει πρώτα τυχαίο αριθμό για το μάθημα. Αν επιλεχθεί ο τυχαίος αριθμός -1, τότε δημιουργείται κενό αλλιώς αλλάζει ο καθηγητής επίσης με τυχαίο τρόπο.

**Method getGenes(int i, int j)**

Επιστρέφει το γονίδιο στη θέση (i,j).

**Method getGenesXLength()**

Επιστρέφει το κάθετο ύψος του πίνακα genes.

**Method getGenesYLength()**

Επιστρέφει το οριζόντιο ύψος του πίνακα genes.

**Method getLessonsXLength()**

Επιστρέφει το κάθετο ύψος του πίνακα lessons.

**Method getLessonsYLength()**

Επιστρέφει το οριζόντιο ύψος του πίνακα lessons.

**Method getLessonCode(int i)**

Επιστρέφει τον κωδικό του μαθήματος στη σειρά i από τον πίνακα lessons.

**Method getLessonl(int i)**

Επιστρέφει το όνομα του μαθήματος στη σειρά i από τον πίνακα lessons σε μορφή String.

**Method getLessonClass(int i)**

Επιστρέφει την τάξη του μαθήματος στη σειρά i από τον πίνακα lessons.

**Method getLessonHours(int i)**

Επιστρέφει τις εβδομαδιαίες ώρες του μαθήματος στη σειρά i από τον πίνακα lessons.

**Method getTeachersXLength()**

Επιστρέφει το κάθετο ύψος του πίνακα teachers.

**Method getTeachersYLength()**

Επιστρέφει το οριζόντιο ύψος του πίνακα teachers.

**Method getTeacherCode(int i)**

Επιστρέφει τον κωδικό του μαθήματος στη σειρά i από τον πίνακα teachers.

**Method getTeachert(int i)**

Επιστρέφει το όνομα του καθηγητή στη σειρά i από τον πίνακα teachers σε μορφή String.

**Method getTeachableLessons(int i)**

Βάζει σε έναν πίνακα τα μαθήματα που μπορεί να διδάξει ο καθηγητής στη σειρά i του πίνακα teachers.

**Method getTeacherTotalDailyHours(int i)**

Επιστρέφει το άνω όριο των ημερήσιων ωρών που μπορεί να διδάξει ο καθηγητής στη σειρά i του πίνακα teachers.

**Method getTeacherTotalWeeklyHours(int i)**

Επιστρέφει το άνω όριο των εβδομαδιαίων ωρών που μπορεί να διδάξει ο καθηγητής στη σειρά i του πίνακα teachers.

**Class Genetic**

**Method geneticAlgorithm(int populationSize, double mutationProbability, int minimumFitness, int maximumSteps)**

**Method initializePopulation(int populationSize)**

**Method updateFitnessBounds()**

**Method Chromosome reproduce(Chromosome x, Chromosome y, Object[ ][ ] lessons, Object[ ][ ] teachers)**

**Class PrintFile**

**Method print(Chromosome chromosome)**

**Class Main**

Αρχικά δημιουργούνται δύο στιγμιότυπα της ReadFiles (r1 και r2) ώστε να διαβαστούν τα αρχεία των μαθημάτων και των καθηγητών αντίστοιχα, τα οποία χρησιμοποιούνται στο αρχικό χρωμόσωμα (Chromosome x) πάνω στο οποία θα εφαρμοστεί ο γενετικός κώδικας (Genetic g). Τέλος, δημιουργείται ένα στιγμιότυπο της PrintFile και τυπώνεται το χρωμόσωμα x, που είναι το αποτέλεσμα του γενετικού αλγορίθμου.

**Αποτέλεσμα**



