



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Τίτλος: «“Αλγόριθμοι και Πολυπλοκότητα”

Χρονοπρογραμματισμός εξετάσεων Πανεπιστημίου V1.0»

Όνομα Καθηγητή: Γκόγκος Χρήστος

Τεχνική Αναφορά

Τίτλος: Εργασία στο μάθημα “Αλγόριθμοι και Πολυπλοκότητα”
Χρονοπρογραμματισμός εξετάσεων Πανεπιστημίου V1.0

Συντάκτες:

1. Παγγές Ιωάννης Α.Μ:00089

Ημερομηνία: Ιανουάριος 2021



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Πίνακας Περιεχομένων

1.0 Περίληψη	3
1.1 Εισαγωγή	4
2.0 Χρονοπρογραμματισμός εξετάσεων Πανεπιστημίου	5
2.1 Εκτέλεση προγράμματος	7
3.0 Συμπεράσματα	8
Βιβλιογραφία	9
Παράρτημα Α.....	10



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

1.0 Περίληψη

Ο χρονοπρογραμματισμός των εξετάσεων είναι μια από τις πιο σημαντικές διοικητικές δραστηριότητες που πραγματοποιούνται σε όλα τα ακαδημαϊκά ιδρύματα. Η αποδοτική δημιουργία προγραμμάτων εξετάσεων είναι ένα σημαντικό και επαναλαμβανόμενο πρόβλημα το οποίο καλούνται να αντιμετωπίσουν τα εκπαιδευτικά ιδρύματα σε όλο τον κόσμο. Μια απλοποιημένη μορφή του προβλήματος έχει προταθεί από τους Carter κ.ά. οι οποίοι διέθεσαν δημόσια 13 στιγμιότυπα προβλημάτων που εν συνεχεία χρησιμοποιήθηκαν σε πληθώρα επιστημονικών εργασιών χρονοπρογραμματισμού. Στα πλαίσια του μεταπτυχιακού προγράμματος οδηγηθήκαμε να δημιουργήσουμε ένα πρόγραμμα για την λύση του χρονοπρογραμματισμού των εξετάσεων σε ακαδημαϊκά ιδρύματα.

Λέξεις κλειδιά: χρονοπρογραμματισμός, πρόγραμμα, εξετάσεις



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

1.1 Εισαγωγή

Το πρόβλημα αφορούσε κυρίως εξετάσεις, σπουδαστές και συνεχόμενες περιόδους σε κάθε μια από τις οποίες μπορούν να διεξαχθούν μια ή περισσότερες εξετάσεις. Κάθε εξέταση διαθέτει μια λίστα από σπουδαστές και κάθε σπουδαστής μπορεί να είναι εγγεγραμμένος σε μια ή περισσότερες εξετάσεις. Η λύση του προβλήματος συνίσταται στην ανάθεση εξετάσεων σε περιόδους έτσι ώστε να μην υπάρχουν συγκρούσεις, δηλαδή να μην υπάρχουν σπουδαστές που θα έπρεπε να συμμετάσχουν σε εξετάσεις σε περισσότερα του ενός μαθήματα στην ίδια περίοδο. Καθώς είναι ενδεχόμενο να υπάρχουν πολλά εναλλακτικά προγράμματα που ικανοποιούν τον ανωτέρω περιορισμό, προτιμότερο θεωρείται εκείνο το πρόγραμμα που διαθέτει επαρκή διαστήματα προετοιμασίας ανάμεσα σε διαδοχικές εξετάσεις για όλους τους φοιτητές συνολικά. Το πρόγραμμα υλοποιήθηκε κατά κύριο λόγο στη γλώσσα προγραμματισμού C.

2.0 Χρονοπρογραμματισμός εξετάσεων Πανεπιστημίου

Τα προβλήματα χρονοπρογραμματισμού αποτελούν μια ιδιαίτερη κατηγορία προβλημάτων συνδυαστικής βελτιστοποίησης τα οποία εκδηλώνονται συχνά στην πράξη. Υπάρχουν πολλά προβλήματα χρονοπρογραμματισμού (χρονοπρογραμματισμός εκπαιδευτικών ιδρυμάτων, χρονοπρογραμματισμός προσωπικού ανθρώπινου δυναμικού, χρονοπρογραμματισμός αθλητικών γεγονότων).

Οι Burke, Kingston and De Werra(2004) έδωσαν ένα σημαντικό ορισμό για το πρόβλημα του χρονοπρογραμματισμού. Όπου ένας χρονοπρογραμματισμός προβλήματος αποτελείτε από τέσσερις βασικές παραμέτρους:

- **T** ένα πεπερασμένο σύνολο χρόνοθυρίδων (timeslot)
- **P** ένα πεπερασμένο σύνολο πόρων (resource)
- **M** ένα πεπερασμένο σύνολο από εξετάσεις (meetings-exams)
- **C** ένα πεπερασμένο σύνολο από περιορισμούς (constraints)

Το πρόβλημα είναι πως μπορούμε να τοποθετήσουμε χρόνο και πόρους στο σύνολο των εξετάσεων έτσι ώστε να μπορέσουμε να ικανοποιήσουμε τους περισσότερους δυνατούς περιορισμούς, αυτό ονομάζεται το λεγόμενο NP πρόβλημα.

Στόχος της παρούσας εργασίας είναι να δημιουργήσουμε-αναπτύξουμε ένα πρόγραμμα που τα λύνει το πρόβλημα του χρονοπρογραμματισμού των εξετάσεων. Για να γίνει αυτό εφικτό πήραμε δεκατρία πραγματικά προβλήματα (University of Toronto Benchmark Data), έτσι ώστε να μπορέσουμε να δοκιμάσουμε τις λύσεις για να μπορέσουμε να τα συγκρίνουμε.

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Πρόβλημα	Αρχείο Δεδομένων	Εξετάσεις	Φοιτητές	Εγγραφές	Περίοδοι	Πυκνότητα
car-f-92	car-f-92.stu	543	18419	55522	32	0.14
car-s-91	car-s-91.stu	682	16925	56877	35	0.13
ear-f-83	ear-f-83.stu	190	1125	8109	24	0.27
hec-s-92	hec-s-92.stu	81	2823	10632	18	0.42
kfu-s-93	kfu-s-93.stu	461	5349	25113	20	0.06
lse-f-91	lse-f-91.stu	381	2726	10918	18	0.06
pur-s-93	pur-s-93.stu	2419	30029	120681	42	0.03
rye-s-93	rye-s-93.stu	486	11483	45051	23	0.07
sta-f-83	sta-f-83.stu	139	611	5751	13	0.14
tre-s-92	tre-s-92.stu	261	4360	14901	23	0.18
uta-s-92	uta-s-92.stu	622	21266	58979	35	0.13
ute-s-92	ute-s-92.stu	184	2749	11793	10	0.08
yor-f-83	yor-f-83.stu	181	941	6034	21	0.29

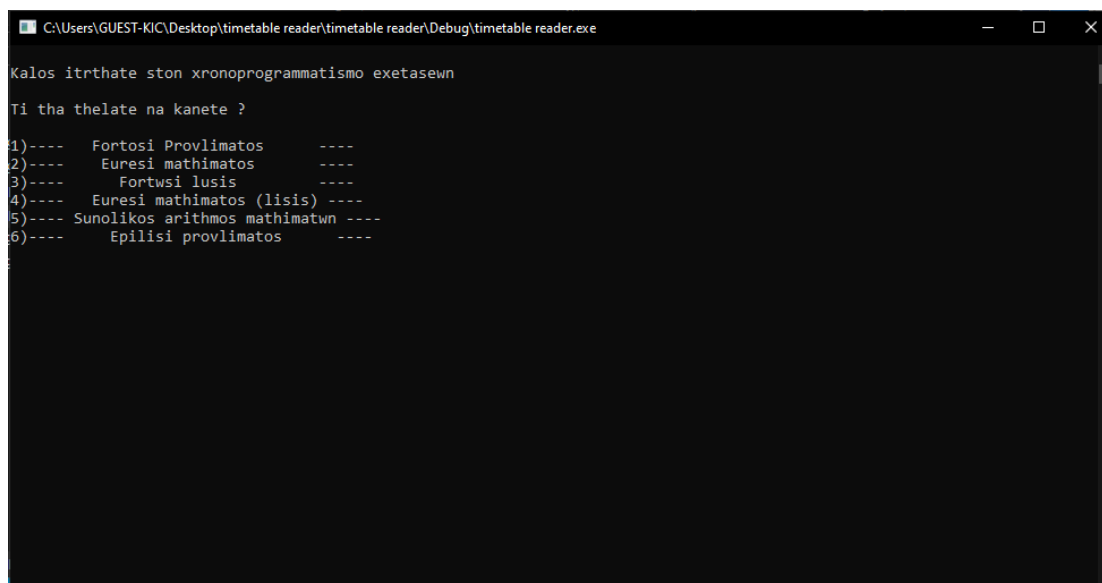
Εικόνα 1 Δεδομένα Προβλημάτων

Αρχείο λύσης	Κόστος λύσης
car-f-92(3.71).sol	3.71
car-s-91(4.39).sol	4.39
ear-f-83(32.63).sol	32.63
hec-s-92(10.05).sol	10.04
kfu-s-93(12.90).sol	12.90
lse-f-91(9.82).sol	9.82
pur-s-93(4.49).sol	4.49
rye-s-93(7.93).sol	7.93
sta-f-83(157.03).sol	157.03
tre-s-92(7.72).sol	7.72
uta-s-92(3.04).sol	3.04
ute-s-92(24.77).sol	24.77
yor-f-83(34.71).sol	34.71

Εικόνα 2 Κόστος Λύσεων

2.1 Εκτέλεση προγράμματος

Τα βασικά βήματα ώστε ο χρήστης να τρέξει ένα πρόβλημα του χρονοπρογραμματισμού των εξετάσεων θα περιγραφούν παρακάτω. Εκτελώντας το .exe μας παρουσιάζεται το shell. Η πρώτη επιλογή (Fortosi provlimatos) είναι για φόρτωση του αρχείου προβλήματος .stu που είναι τα μαθήματα. Η δεύτερη επιλογή (Euresi mathimatos), η οποία είναι χρήσιμη μόνο μετά την (1) είναι αναζήτηση του αριθμού των μαθητών που έχουν πάρει ένα μάθημα. Η επιλογή (3) είναι για φόρτωση του αρχείου .stu που είναι τα μαθήματα που έχουν πάρει όλοι οι φοιτητές του ιδρύματος. Μετά την εκτέλεση της υπάρχουν οι επιλογές (4-Euresi mathimatos lisis) και (5- Sunolikos arithmos mathimatwn) που είναι για αναζήτηση των μαθημάτων που έχει πάρει ένας τυχαίος φοιτητής καθώς και για την αναζήτηση του συνολικού αριθμού των μαθημάτων στο ίδρυμα. Έπειτα και αφού έχουν φορτωθεί τα δυο αρχεία του προβλήματος, με την επιλογή (6-Epilisi provlimatos) εισάγουμε αρχείο με κατάληξη .sol έτσι ώστε να συγκρίνουμε το κόστος λύσεων, με την επιλογή (6-Epilisi provlimatos) πρέπει να εισάγουμε και τον αριθμό περιόδου. Ο κώδικας της εφαρμογής βρίσκεται [εδώ](#)



```
C:\Users\GUEST-KIC\Desktop\timetable reader\timetable reader\Debug\timetable reader.exe

Kalos itrthate ston xronoprogrammatismo exetasewn

Ti tha thelate na kanete ?

1)---- Fortosi Provlmatos      ----
2)---- Euresi mathimatos      ----
3)---- Fortwsi lusi          ----
4)---- Euresi mathimatos (lisis) ----
5)---- Sunolikos arithmos mathimatwn ----
6)---- Epilisi provlimatos    ----
```

Εικόνα 3 Γραφικό Περιβάλλον "Χρονοπρογραμματισμού Εξετάσεων"



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

3.0 Συμπεράσματα

Ο χρονοπρογραμματισμός των εξετάσεων είναι μια από τις πιο σημαντικές διοικητικές δραστηριότητες που πραγματοποιούνται σε όλα τα ακαδημαϊκά ιδρύματα. Πολλές μέθοδοι αναπτύχθηκαν, για την επίλυσή του, άλλες με επιτυχία και άλλες με όχι και τόσο καλά αποτελέσματα.

Η αντιμετώπιση του προβλήματος του χρονοπρογραμματισμού των εξετάσεων από τους εξελικτικούς αλγορίθμους, ήρθε με την παρατήρηση ότι αυτό το πρόβλημα μπορεί να παρομοιασθεί και να μετατραπεί σε πρόβλημα χρωματισμού ενός γράφου (graph colouring problem), κάτι το οποίο έχει επιτυχία. Τέλος, η εφαρμογή που αναπτύχθηκε εφαρμόστηκε σε μία σειρά πειραμάτων, πάνω σε δοκιμαστικά δεδομένα, και τα αποτελέσματα που προέκυψαν ήταν πάρα πολύ κοντά στις λύσεις.



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

Βιβλιογραφία

- [1] Κομηνός Χαράλαμπου-Γαβριήλ, “Υπολογιστικές εφαρμογές σε περιβάλλον παράλληλης επεξεργασίας,” Πανεπιστήμιο Πάτρας, 2013.
- [2] Χ. Γ. ΓΚΟΓΚΟΣ, “ΑΛΓΟΡΙΘΜΟΙ ΣΥΝΔΥΑΣΤΙΚΗΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΜΕ ΕΜΦΑΣΗ ΣΕ ΜΕΤΑΕΥΡΕΤΙΚΕΣ ΤΕΧΝΙΚΕΣ- ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ,” ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΑΣ, ΠΑΤΡΑ, 2009.
- [3] M. N. M. K. Ashis Kumar Mandal, *Solving Examination Timetabling Problem Using Partial Exam Assignment with Hill Climbing Search*. IEEE Xplore, 2015.
- [4] T. U. of Nottigham, “Benchmark Exam Timetabling Datasets.” <http://128.243.21.198/external/resources/index.html> (accessed Jan. 20, 2021).
- [5] The University of Nottigham, “Benchmark Exam Timetabling Datasets.” <http://www.cs.nott.ac.uk/~pszrq/data.htm> (accessed Jan. 20, 2021).
- [6] R. Qu, E. K. Burke, B. McCollum, L. T. G. Merlot, and S. Y. Lee, “A survey of search methodologies and automated system development for examination timetabling,” *J. Sched.*, vol. 12, no. 1, pp. 55–89, Feb. 2009, doi: 10.1007/s10951-008-0077-5.

Παράρτημα Α

- `int choices(void)`: Συνάρτηση που δέχεται την αριθμητική επιλογή του χρήστη από το πληκτρολόγιο και την επιστρέφει σε ακέραιο.
- `struct course *readcourseslist(int *globalcoursesaddress)`: Συνάρτηση που δέχεται σαν όρισμα την διεύθυνση μιας ακέрайης τιμής ώστε να αποθηκεύσει τον συνολικό αριθμό των μαθημάτων και επιστρέφει τον root της Linked list με τα μαθήματα.
- `struct student *load_student(void)`: Συνάρτηση που επιστρέφει τον root της linked list των μαθητών.
- `void printcourses(struct course *ptr,int crs_to_find)`: Συνάρτηση που δέχεται τον Root της linked list των μαθημάτων και έναν ακέραιο Id προς αναζήτηση και αν υπάρχει στην Ram το αντίστοιχο μάθημα εκτυπώνει τα enrollments
- `void printstudent(struct student *ptr,int Idlookup)`: Συνάρτηση που δέχεται τον root της linked list των μαθητών και το Id ενός μαθητή και αν αυτός υπάρχει εκτυπώνει τα μαθήματα που έχει πάρει αυτός.
- `void printallstudents(struct student *ptr)`: Συνάρτηση που παίρνει τον root της λίστας των μαθητών και τους εκτυπώνει όλους.
- `struct courseintb** Make2dintArray(int arraySizeX)`: Συνάρτηση η οποία δέχεται σαν όρισμα ένα μέγεθος και επιστρέφει έναν πίνακα δεικτών προς δομές τύπου `Courseintb` μεγέθους `arraySizeX`.
- `void filltable(struct courseintb **timetable,int** conflictmatrix,int courses,int *sortedmatrix,int timeslots)`: Συνάρτηση που έχει σαν σκοπό να γεμίσει το άδειο `timetable` ενός χρωμοσώματος. Δέχεται σαν ορίσματα το `timetable`, τον `CD`, τα μαθήματα, τον πίνακα του `LD` και τα `timeslots`
- `double calculatecost(struct student *ptr,struct courseintb **timetable,int timeslots)`: Συνάρτηση υπολογισμού κόστους που επιστρέφει το κόστος και δέχεται σαν ορίσματα τον Root των μαθητών, ένα `timetable` για να υπολογίσει το κόστος του και τα `timeslots`.
- `double costforstudent(int *student,struct courseintb **timetable,int timeslots)`: Συνάρτηση που υπολογίζει το κόστος για έναν μαθητή και δέχεται σαν ορίσματα τον πίνακα των μαθημάτων του μαθητή, το `timetable` και τα `timeslots`.



ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΚΤΥΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

- `int lookforcourse(int coursetofind, struct courseintb **timetable, int timeslots):`
Συνάρτηση αναζήτηση μαθήματος(όρισμα 1) σε ένα timetable(όρισμα 2). Τελευταίο όρισμα είναι τα timeslots.
- `void countcourses(struct courseintb **timetable, int timeslots):` Συνάρτηση που μετράει τα μαθήματα που βρίσκονται σε ένα timetable(1).
- `double distancecost(int A, int B):` Συνάρτηση που επιστρέφει το κόστος δυο μαθημάτων τύπου double που βρίσκονται στα timeslots A,B που είναι τα ορίσματα της.