

Πανεπιστήμιο Ιωαννίνων  
Τμήμα Πληροφορικής και Τηλεπικοινωνιών  
Μεταπτυχιακό Πληροφορικής και Δικτύων (MSc)

## Χρονοπρογραμματισμός εξετάσεων Πανεπιστημίου Με χρήση τεχνικών βασισμένες σε γραφήματα

Παχούλας Γεώργιος AM 091  
Ιανουάριου 2021

## Περιεχόμενα

Περίληψη .....	3
Εισαγωγή .....	4
1 Προβλήματα Χρονοπρογραμματισμού .....	5
1.1 Περιγραφή του προβλήματος.....	5
1.2 Χρονοδιάγραμμα εξετάσεων.....	5
1.3 Δεδομένα του προβλήματος .....	6
2 Περιγραφή του προγράμματος.....	6
2.1 Μενού επιλογών.....	6
2.2 Σύντομη περιγραφή του κώδικα .....	7
3 Επίλυση των προβλημάτων.....	7
3.1 Graph Coloring .....	7
3.2 Αποτελέσματα – Συμπεράσματα .....	8
4 Βιβλιογραφία .....	11

## Περίληψη

Ο χρονοπρογραμματισμός αποτελεί σημαντικό κομμάτι σε πολλούς τομείς της καθημερινότητας. Ένας από αυτούς είναι στο κομμάτι της εκπαίδευσης στον οποίο εμφανίζεται κυρίως με δύο μορφές οι οποίες είναι το χρονοδιάγραμμα των τάξεων και το χρονοδιάγραμμα εξετάσεων. Ο χρονοπρογραμματισμός εξετάσεων είναι μια ιδιαίτερα σημαντική αλλά και ταυτόχρονα αρκετά δύσκολη διαδικασία λόγω της πολυπλοκότητας του προβλήματος προκειμένου να αποφευχθούν οι συγκρούσεις επικαλύψεις μεταξύ των μαθημάτων. Στην συγκεκριμένη έρευνα παρουσιάζεται μια επίλυση του προβλήματος χρονοπρογραμματισμού εξετάσεων σε 13 πραγματικά προβλήματα από διάφορα πανεπιστήμια και σχολεία μέσω της χρήσης αλγορίθμων χρωματισμού γράφων.

## Εισαγωγή

Σε αυτή την εργασία σκοπός είναι να παρουσιάσουμε μια λύση για έναν αριθμό προβλημάτων χρονοπρογραμματισμού. Στο πρώτο κεφάλαιο γίνεται μια γενική περιγραφή σχετικά με τα προβλήματα χρονοπρογραμματισμού και συγκεκριμένα με τα προβλήματα που αφορούν τον τομέα της εκπαίδευσης. Στην συνέχεια περιγράφουμε τα δεδομένα τα οποία θα χρησιμοποιηθούν ως προς επίλυση καθώς και την δομή με την οποία είναι αποθηκευμένα στα αρχεία. Στο δεύτερο κεφάλαιο γίνεται μία σύντομη περιγραφή στο τρόπο λειτουργίας της εφαρμογής και συγκεκριμένα στο μενού επιλογών που εμφανίζεται στον χρήστη. Έπειτα ακολουθεί μια μικρή περιγραφή των κυριότερων συναρτήσεων που χρησιμοποιούνται μέσα στον κώδικα. Τέλος περιγράφουμε τον τρόπο επίλυσης των δεδομένων με την χρήση του Greedy Coloring algorithm και τα αποτελέσματα από την επίλυση των προβλημάτων καθώς επίσης και ορισμένους τρόπους βελτιστοποίησης των λύσεων.

# 1 Προβλήματα Χρονοπρογραμματισμού

## 1.1 Περιγραφή του προβλήματος

Τα προβλήματα χρονοδιαγράμματος ή χρονοπρογραμματισμού προκύπτουν σε διάφορους τομείς όπως στον τομέα της εκπαίδευσης (δημιουργία προγράμματος μαθημάτων καθώς και εξεταστικής), στον τομέα της υγείας (δημιουργία εβδομαδιαίου προγράμματος για νοσοκόμες) και σε πολλούς άλλους τομείς. Τα προβλήματα αυτά αποτελούν ένα απαιτητικό κομμάτι για τους ερευνητές τόσο στην τεχνητή νοημοσύνη όσο και στην επιχειρησιακή έρευνα με αποτέλεσμα την διοργάνωση διαφόρων συνεδρίων για την πρακτική και θεωρητική μελέτη για τον αυτοματοποιημένο χρονοπρογραμματισμό - PATAT (the **P**ractise and **T**heory on **A**utomated **T**imetabling). [1]

Μια γενική μορφή ενός προβλήματος χρονοπρογραμματισμού περιγράφεται από τις εξής παραμέτρους:

- **T**: Ένα πεπερασμένο σύνολο χρόνων
- **R**: Ένα πεπερασμένο σύνολο πόρων
- **M**: Ένα πεπερασμένο σύνολο συναντήσεων
- **C**: Ένα πεπερασμένο σύνολο περιορισμών

## 1.2 Χρονοδιάγραμμα εξετάσεων

Ο χρονοπρογραμματισμός στην εκπαίδευση αποτελεί ένα από τα πιο ευρέως μελετημένα προβλήματα καθώς αποτελεί ένα από τα πιο σημαντικά και χρονοβόρα καθήκοντα. Διαθέτει διάφορες εκδοχές όπως σχολικό χρονοδιάγραμμα (προγραμματισμός καθηγητή – τάξης), μαθημάτων πανεπιστημίου (χρονοδιάγραμμα εξετάσεων) κτλ. [2]

Το πρόβλημα που μελετάτε σε αυτή την έρευνα αφορά την εκδοχή για τον χρονοπρογραμματισμό των εξετάσεων δηλαδή για την δημιουργία ενός προγράμματος εξεταστικής τέτοιο ώστε να μην υπάρχουν συγκρούσεις δηλαδή οι φοιτητές να μην δίνουν παραπάνω από ένα μάθημα σε κάθε περίοδο καθώς και οι «αποστάσεις» των περιόδων για κάθε μάθημα να είναι όσο πιο μεγάλες προκειμένου να προλαβαίνουν οι φοιτητές να διαβάσουν.

Γενικά οι περιορισμοί σε προβλήματα χρονοδιαγράμματος ταξινομούνται σε 2 κατηγορίες

**Σκληροί περιορισμοί (Hard constraints):** Οι περιορισμοί αυτοί πρέπει να ικανοποιούνται πάντα προκειμένου η παραγόμενη λύση να είναι ορθή και χωρίς συγκρούσεις. Για παράδειγμα στο συγκεκριμένο πρόβλημα ως σκληρός περιορισμός θεωρείται ότι κάθε φοιτητής θα μπορεί να συμμετάσχει σε ένα μάθημα ανά περίοδο.

**Μαλακοί περιορισμοί (Soft constraints):** Συμβάλουν στην μεγιστοποίηση της απόδοσης της παραγόμενης λύσης και δεν απαιτείται απαραίτητα να ικανοποιηθούν. Στο συγκεκριμένο πρόβλημα ως μαλακός περιορισμός θεωρείται η απόδοση τιμών ποινών προκειμένου να υπάρχει ένα διάστημα προετοιμασίας.

### 1.3 Δεδομένα του προβλήματος

Τα δεδομένα που χρησιμοποιούνται αποτελούν μια απλοποιημένη μορφή του προβλήματος η οποία έχει προταθεί το 1996 από τους Carter κ.ά. Τα δεδομένα αποτελούνται από ένα σύνολο 13 πραγματικών προβλημάτων χρονοπρογραμματισμού και συγκεκριμένα από 3 Καναδικά γυμνάσια, 5 Καναδικά πανεπιστήμια, 1 Αμερικανικό πανεπιστήμιο, 1 Βρετανικό πανεπιστήμιο και ένα πανεπιστήμιο στην Σαουδική Αραβία.

Τα δεδομένα σε κάθε αρχείο είναι αποθηκευμένα με την εξής μορφή:

- Οι γραμμές αντιπροσωπεύουν τους φοιτητές.
- Κάθε γραμμή / φοιτητής περιέχει τα μαθήματα / εξετάσεις στα οποία είναι εγγεγραμμένος.

## 2 Περιγραφή του προγράμματος

### 2.1 Μενού επιλογών

Με την έναρξη της εφαρμογής εμφανίζεται στον χρήστη το μενού επιλογών της εφαρμογής. Το μενού αποτελείται από 3 επιλογές οι οποίες είναι οι εξής:

**Load data:** Εμφανίζει όλα τα αρχεία – προβλήματα από τα οποία ο χρήστης επιλέγει ένα το οποίο θα φορτώσει προς επεξεργασία. Με την ολοκλήρωση της φόρτωσης του αρχείου εμφανίζονται

ορισμένα στοιχεία του όπως αριθμός φοιτητών, εξετάσεων, εγγραφών και η πυκνότητα του προβλήματος.

**Solve problem:** Επιλύει το πρόβλημα που έχει επιλεγθεί από την πρώτη επιλογή (Load data). Με την ολοκλήρωση της λύσης εμφανίζει τον αριθμό των περιόδων που χρησιμοποιήθηκαν καθώς και το κόστος λύσης. Επίσης δημιουργεί ένα αρχείο .csv στο οποίο αποθηκεύει την εξέταση και την περίοδο στην οποία έχει τοποθετηθεί.

**Exit:** Τερματίζει την λειτουργία του προγράμματος.

## 2.2 Σύντομη περιγραφή του κώδικα

Ο κώδικας χωρίζεται σε συναρτήσεις κάθε μια από αυτές υλοποιεί μια συγκεκριμένη λειτουργία. Οι σημαντικότερες από αυτές είναι:

**readFile():** Διαβάζει το αρχείο – πρόβλημα και δημιουργεί ένα λεξικό που κάθε εγγραφή έχει ως κλειδί τον αριθμό της εξέτασης και ως τιμή ένα πίνακα με τους αριθμούς των φοιτητών οι οποίοι είναι εγγεγραμμένοι σε αυτό.

**conflictTable():** Δημιουργεί τον πίνακα συγκρούσεων και υπολογίζει την πυκνότητα του προβλήματος.

**problemInfo():** Εμφανίζει τα στοιχεία του προβλήματος όπως αριθμό εγγραφών, φοιτητών, εξετάσεων καθώς και την πυκνότητα του προβλήματος.

**colorGraph():** Επιλύει το τρέχον πρόβλημα με την χρήση της μεθόδου χρωματισμού γράφων (graph coloring).

**calculateCost():** Υπολογίζει το κόστος της λύσης με βάση τα διαστήματα προετοιμασίας στις εξετάσεις που είχαν σημειωθεί η συγκρούσεις.

**exportSolution():** Δημιουργεί το αρχείο csv με την λύση του προβλήματος.

## 3 Επίλυση των προβλημάτων

### 3.1 Graph Coloring

Η μέθοδος που έχει επιλεγεί για την επίλυση των προβλημάτων είναι ο χρωματισμός γράφων (Graph Coloring). Στη θεωρία των γραφημάτων ο χρωματισμός γράφων αποτελεί μια μέθοδο με την οποία χρωματίζει τις κορυφές ενός γραφήματος έτσι ώστε καμία γειτονική κορυφή να έχει το ίδιο χρώμα. [3] Στο συγκεκριμένο πρόβλημα αντί για χρωματισμούς έχουμε περιόδους εξετάσεων οι οποίες αναθέτονται σε στοιχεία – κόμβους του γραφήματος που αντιστοιχούν στις εξετάσεις. Δυστυχώς δεν υπάρχει κάποιος αποτελεσματικός αλγόριθμος για χρωματισμό ενός γραφήματος ο οποίος να εξασφαλίζει τον ελάχιστο αριθμό χρωμάτων καθώς ο χρωματισμός γράφων αποτελεί ένα πρόβλημα NP complete.

Ο αλγόριθμος που χρησιμοποιήθηκε συγκεκριμένα από την θεωρία του χρωματισμού γράφων είναι ο άπληστος αλγόριθμος χρωματισμού (Greedy Coloring). Ο αλγόριθμος αυτός δεν εγγυάται τη χρήση ελάχιστων χρωμάτων αλλά εγγυάται ένα ανώτερο όριο στον αριθμό χρωμάτων δηλαδή δεν χρησιμοποιεί ποτέ περισσότερα από  $m + 1$  χρώματα όπου  $m$  είναι ο μέγιστος βαθμός μιας κορυφής στο τρέχων γράφημα. [4]

### Σύντομη περιγραφή του αλγόριθμου

1. Χρωμάτισε την πρώτη κορυφή με το πρώτο χρώμα
2. Κάνε το ίδιο για τις υπόλοιπες  $V-1$  κορυφές **ΑΛΛΑ** χρωμάτισε την τρέχουσα κορυφή με χρώμα το οποίο δεν ίδιο με κάποια κορυφή η οποία να συνδέεται με την τρέχουσα. Αν όλα τα χρώματα έχουν χρησιμοποιηθεί και δεν υπάρχει κάποιο διαθέσιμο τότε πρόσθεσε ένα καινούργιο χρώμα.

### 3.2 Αποτελέσματα – Συμπεράσματα

Εκτελέσαμε τον κώδικα και για τα 13 προβλήματα. Όπως φαίνεται και στον παρακάτω πίνακα Table1 οι λύσεις που προέκυψαν από τις δικές μας λύσεις (MyCOST) σε σχέση με τις βελτιστοποιημένες λύσεις (OptCost) που παρέχονται από την εκφώνηση του προβλήματος. Με την έννοια βελτιστοποιημένες αναφερόμαστε στο γεγονός ότι πέρα από την λύση του προβλήματος γίνεται και αλλαγές στην ανάθεση των περιόδων με τέτοιο τρόπο ώστε να αυξάνεται το διάστημα μεταξύ των εξετάσεων προκειμένου να προετοιμαστούν οι φοιτητές. Άλλο σημαντικό χαρακτηριστικό που περιγράφει κάθε πρόβλημα είναι η πυκνότητα συγκρούσεων δηλαδή η αναλογία των συγκρούσεων σε σχέση με τα μαθήματα. Όσο πιο μικρή είναι η πυκνότητα ενός



προβλήματος τόσο λιγότερες συγκρούσεις εμφανίζονται στο συγκεκριμένο πρόβλημα. Έπειτα από την ολοκλήρωση της επίλυσης δημιουργείται ένα αρχείο .csv το οποίο περιέχει τις εξετάσεις καθώς και τις περιόδους στις οποίες έχουν τοποθετηθεί.

Table 1 Αποτελέσματα από τις εκτελέσεις των 13 διαφορετικών προβλημάτων

Problem	OptCost	MyCost	Periods	Density
car-f-92	3.71	9.7	33	0.14
car-s-91	4.39	12.56	34	0.13
ear-f-83	32.63	69.33	24	0.27
hec-s-92	10.04	23.5	19	0.42
kfu-s-93	12.9	34.39	22	0.06
lse-f-91	9.82	29.92	18	0.06
pur-s-93	4.49	16.89	37	0.03
rye-s-93	7.93	27.35	22	0.08
sta-f-83	167.03	193.97	13	0.14
tre-s-92	7.72	16.6	22	0.18
uta-s-92	3.04	8.2	34	0.13
ute-s-92	24.77	55.25	10	0.08
yor-f-83	34.71	59.06	22	0.29

Αυτή η διαφορά των λύσεων βελτιστοποιημένων και μη γίνεται καλύτερα εμφανής και στο παρακάτω ραβδόγραμμα.

Table 2 Σύγκριση των μη και των βελτιστοποιημένων λύσεων.



## Πιθανές βελτιώσεις

Κάποιες πιθανές βελτιώσεις στο πρόγραμμα θα ήταν η εφαρμογή ορισμένων αλγορίθμων οι οποίοι θα έβρισκαν την αποδοτική λύση όσο αναφορά ως προς το κόστος λύσης κάθε προβλήματος όπως ο αλγόριθμος αποικίας μυρμηγκιών (Ant Colony optimization), υβριδικός εξελεγκτικός αλγόριθμος, Hill Climbing κτλ.

Στον παρακάτω πίνακα παρουσιάζονται τα αποτελέσματα χρησιμοποιώντας ορισμένους αλγορίθμους βελτιστοποίησης στα 13 προβλήματα. Βλέποντας τον πίνακα προκύπτει ότι ο πρώτος στην κατάταξη είναι ο αλγόριθμος Hill Climbing με συνολικό κόστος 279.

Table 3 Συγκριτικός πίνακας ανάμεσα σε αλγορίθμους βελτιστοποίησης των λύσεων

Name	V	Density	Vertex degree			Colours at Cut-off: Mean and best (in parenthesis)					
			Min;Med;Max	Mean	CV	TABUCOL	PARTIALCOL	HEA	ANTCOL	HC	Bktr
hec-s-92	81	0.415	9; 33; 62	33.7	36.3%	17.22 (17)	17.00 (17)	17.00 (17)	17.04 (17)	17.00 (17)	19.00 (19)
sta-f-83	139	0.143	7; 16; 61	19.9	67.4%	13.35 (13)	13.00 (13)	13.00 (13)	13.13 (13)	13.00 (13)	13.00 (13)
yor-f-83	181	0.287	7; 51; 117	52	35.2%	19.74 (19)	19.00 (19)	19.06 (19)	19.87 (19)	19.00 (19)	20.00 (20)
ute-s-92	184	0.084	2; 13; 58	15.5	69.1%	10.00 (10)	10.00 (10)	10.00 (10)	11.09 (10)	10.00 (10)	10.00 (10)
car-f-83	190	0.266	4; 45; 134	50.5	56.1%	26.21 (24)	22.46 (22)	22.02 (22)	22.48 (22)	22.00 (22)	22.00 (22)
tro-s-92	261	0.18	0; 45; 145	47	59.6%	20.58 (20)	20.00 (20)	20.00 (20)	20.04 (20)	20.00 (20)	23.00 (23)
lse-f-91	381	0.062	0; 16; 134	23.8	93.2%	19.42 (18)	17.02 (17)	17.00 (17)	17.00 (17)	17.00 (17)	17.00 (17)
kfu-s-93	461	0.055	0; 18; 247	25.6	120.0%	20.76 (19)	19.00 (19)	19.00 (19)	19.00 (19)	19.00 (19)	19.00 (19)
rye-s-93	486	0.075	0; 24; 274	36.5	111.8%	22.40 (21)	21.06 (21)	21.04 (21)	21.55 (21)	<b>21.00</b> (21)	22.00 (22)
car-f-92	543	0.138	0; 64; 381	74.8	75.3%	39.92 (36)	32.48 (31)	28.50 (28)	30.04 (29)	27.96 (27)	<b>27.00</b> (27)
uta-s-92	622	0.125	1; 65; 303	78	73.7%	41.65 (39)	35.66 (34)	30.80 (30)	32.89 (32)	30.27 (30)	<b>29.00</b> (29)
car-s-91	682	0.128	0; 77; 472	87.4	70.9%	39.10 (32)	30.20 (29)	29.04 (28)	29.23 (29)	29.10 (28)	<b>28.00</b> (28)
pur-s-93	2419	0.029	0; 47; 857	71.3	129.5%	50.70 (47)	45.48 (42)	33.70 (33)	33.47 (33)	33.87 (33)	<b>33.00</b> (33)
<b>Total</b>						341.05 (315)	302.36 (294)	280.16 (277)	286.84 (281)	<b>279.20</b> (276)	282.00 (282)
<b>Rank</b>						(6)	(5)	(2)	(4)	(1)	(3)

## 4 Βιβλιογραφία

- [1] 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.*, pp. 55-89, 2009.
- [2] H. Babaei, J. Karimpour and A. Hadidi, "A survey of approaches for university course timetabling problem," *Comput. Ind. Eng.*, pp. 43-59, 2015.
- [3] M. Parahar, «Tutorialspoint,» 23 August 2019. [Ηλεκτρονικό]. Available: <https://www.tutorialspoint.com/the-graph-coloring>. [Πρόσβαση 15 January 2021].
- [4] L. Kučera, «The greedy coloring is a bad probabilistic algorithm,» *J. Algorithm.*, pp. 674-684, 1991.