

ΕΘΝΙΚΟ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Καθηγητής : Κουμπάρακης Μανόλης

Ημ/νία παράδοσης: 11/01/2011

Όνομ/μο φοιτητή : Μπεγέτης Νικόλαος
A.M.: 1115200700281

Δεύτερο πακέτο ασκήσεων

Σχόλια και παρουσίαση αποτελεσμάτων για Πρόβλημα 2:

Όπως ζητείται κατά την εκκίνηση του προγράμματος ορίζουμε τρία αντικείμενα *Australia, USA, France* στα οποία θέλουμε να βρεθεί λύση του χρωματισμού των χαρτών τους με τρία, τέσσερα και τέσσερα χρώματα αντίστοιχα, ορίζοντας τα προβλήματα αναζήτησης λύσης ως πρόβλημα ικανοποίησης περιορισμών, δηλαδή για να συμπληρωθούν όλες οι πολιτείες κάθε μίας χώρας με ένα χρώμα θα πρέπει να γίνουν κάποιοι έλεγχοι συνέπειας που υπακούουν σε κάποιους περιορισμούς που δίνονται από τη συνάρτηση *has_conflict*.

Πιο συγκεκριμένα, στην άσκηση αυτή μας δίνεται υλοποιημένος ο κώδικας για τα CSP προβλήματα και για το χρωματισμό χαρτών και το μόνο που καλούμαστε να κάνουμε είναι να υλοποιήσουμε το κυρίως πρόγραμμα (*main*) στο οποίο αφού γίνει η αρχικοποίηση του βασικών αντικειμένων *Australia, USA, France* ως προβλήματα CSP ο χρήστης θα καλείται να επιλέξει ανάμεσα στους 3 αλγόριθμους προβλημάτων ικανοποίησης περιορισμών που δίνονται από την εκφώνηση (*FC+MRV, MAC, Min-Conflicts*) για να βρει λύση στο map-coloring. Αφού επιλέξει, ξεκινάει ο αλγόριθμος οπισθοδρόμησης για τα *FC+MVR* και *MAC* ή ο αλγόριθμος τοπικής αναζήτησης με τον ευρεστικό μηχανισμό των ελαχίστων συγκρούσεων να χρονομετρείται και σταματάει μόλις θα έχει βρεθεί λύση. Έπειτα εκτυπώνονται τα αποτελέσματα:

- *Εκτύπωση λύσης του χρωματισμού των χαρτών και των τριών χωρών*
- *Συνολικές συγκρούσεις περιορισμών για έλεγχο συνέπειας*
- *Συνολικές αναθέσεις τιμών στις μεταβλητές του CSP*
- *συνολικός χρόνος εκτέλεσης αλγορίθμου.*

Συνεπώς έπειτα από όλα τα παραπάνω μπορούμε να συμπεράνουμε ότι ο χρωματισμός των χαρτών μας όντως είναι μοντελοποιημένος ως πρόβλημα ικανοποίησης περιορισμών αφού μπορούμε να το διατυπώσουμε με τα εξής τρία στοιχεία:

Μεταβλητές: Ονόματα πολιτειών κάθε χώρας π.χ για την Αυστραλία: {SA, WA, NT, Q, NSW, V, T}

Πεδίο τιμών: Η γκάμα των τριών ή τεσσάρων χρωμάτων ({R, G, B} για την Αυστραλία και {R, G, B, Y} για την Αμερική και τη Γαλλία)

Περιορισμοί: Καμία πολιτεία να μην έχει το ίδιο χρώμα με κάποια από τις γειτονικές της.

Όπως ξαναγράψαμε προηγουμένως ζητείται να χρησιμοποιηθούν τρεις αλγόριθμοι προβλημάτων ικανοποίησης περιορισμών (FC+MRV, MAC, Min-Conflicts) για να βρεθεί λύση στο πρόβλημα. Λίγα λόγια για αυτούς πριν περάσουμε στις εκτυπώσεις και το σχολιασμό των αποτελεσμάτων:

- **FC+MRV :** Ο αλγόριθμος FC+MRV συνδυάζει τον πρώιμο έλεγχο μαζί με την ευρετική συνάρτηση ελάχιστων απομενουσών τιμών με πολύ αποδοτικό τρόπο και γι' αυτό το λόγο θεωρείται ως ο αποδοτικότερος στην κατηγορία του. Συγκεκριμένα, ο αλγόριθμος οπισθοδρόμησης χρησιμοποιεί την FC ως έναν τρόπο να αξιοποιεί καλύτερα τους περιορισμούς κατά την αναζήτηση. Αυτό γίνεται με τον εξής τρόπο: Όποτε ανατίθεται τιμή σε μία μεταβλητή X, η διαδικασία του πρώιμου ελέγχου εξετάζει κάθε μεταβλητή Y που δεν της έχει ανατεθεί τιμή η οποία συνδέεται με την X με έναν περιορισμό, και διαγράφει από το πεδία της Y οποιαδήποτε τιμή που είναι συνεπής με την τιμή που επιλέχθηκε για την X. Επίσης, δεν επιλέγει απλώς την επόμενη μεταβλητή στην οποία δεν έχει ανατεθεί τιμή με τη σειρά που προκύπτει από τη λίστα των μεταβλητών, αλλά χρησιμοποιεί τον ευρεστικό μηχανισμό των ελάχιστων απομενουσών τιμών(MRV) μειώνοντας έτσι τον παράγοντα διακλάδωσης των μελλοντικών επιλογών με την εκλογή της μεταβλητής που ενέχεται στο μεγαλύτερο αριθμό περιορισμών ως προς τις άλλες μεταβλητές στις οποίες δεν έχει ανατεθεί τιμή.
- **MAC :** Αρχικά ο αλγόριθμος MAC ή αλλιώς *συνέπειας τόξου* είναι αρκετά ισχυρότερος από τον FC+MRV. Με τον όρο τόξο εννοούμε ένα προσανατολισμένο τόξο στο γράφημα περιορισμών, όπως το τόξο από τη μεταβλητή SA στην NSW όπως δίνεται από το σύγγραμμα στη σελίδα 188. Ένα τόξο λένε ότι είναι συνεπές μόνο αν για κάθε τιμή x μιας μεταβλητής X υπάρχει κάποια τιμή y μιας μεταβλητής Y που είναι συνεπής με τη X. Τέλος ο MAC υλοποιείται πλήρως με τη χρήση ουράς για να παρακολουθεί τα τόξα που χρειάζεται να ελέγχονται για ασυνέπεια(AC-3)
- **Min-Conflicts :** Ανήκει στην κατηγορία των τοπικών αλγορίθμων οι οποίοι αποδεικνύεται ότι είναι από τους πιο αποτελεσματικούς για την επίλυση προβλημάτων ικανοποίησης περιορισμών. Χρησιμοποιεί μία διατύπωση με πλήρεις καταστάσεις: Η αρχική κατάσταση αναθέτει τιμή σε κάθε μεταβλητή και η συνάρτηση διαδόχων συνήθως αναλαμβάνει να αλλάζει την τιμή μίας-μίας μεταβλητής. Για την επιλογή νέας τιμής για μία μεταβλητή ο ευρετικός μηχανισμός που χρησιμοποιείται είναι να επιλέγεται η τιμή που προκύπτει με τον ελάχιστο αριθμό συγκρούσεων με άλλες με μεταβλητές.

Ακολουθούν οι τρεις εκτυπώσεις, μία για κάθε αλγόριθμό που μας ζητείται στην εκφώνηση. Έπειτα ακολουθούν συγκρίσεις των αλγορίθμων με βάση τις συγκρούσεις, τις αναθέσεις τιμών και τον χρόνο εκτέλεσης των αλγορίθμων:

```
Python Shell
File Edit Shell Debug Options Windows Help
Type "copyright", "credits" or "license()" for more information.
>>> ----- RESTART -----
>>>
Switch the algorithm you want to use for finding a solution for map coloring Australia, USA and France :
1. FC+MRV (forward checking search with the heuristic mechanism MRV)
2. MAC (maintaining arc consistency)
3. Min-Conflicts (minimum conflicts)

Type : 1
Forward checking search with the heuristic mechanism MRV(FC+MRV) is selected

Map Coloring solution:

AUSTRALIA
Map coloring result: {'WA': 'R', 'Q': 'R', 'T': 'R', 'V': 'R', 'SA': 'G', 'NT': 'B', 'NSW': 'B'}

Total puzzle conflicts: 30
Total assignments: 7
Execution time of Australia map coloring : 0.001 seconds

USA
Map coloring result: {'WA': 'R', 'WI': 'G', 'WV': 'B', 'FL': 'B', 'WY': 'R', 'NH': 'G', 'NJ': 'B', 'NM': 'R', 'NC': 'R', 'ND': 'R', 'NE': 'B', 'NY': 'G', 'RI': 'G', 'NV': 'R', 'CO': 'G', 'CA': 'G', 'GA': 'G', 'CT': 'R', 'KA': 'R', 'OK': 'B', 'OH': 'Y', 'SC': 'B', 'KY': 'R', 'OR': 'B', 'SD': 'G', 'DE': 'G', 'DC': 'R', 'HI': 'R', 'TX': 'G', 'LA': 'B', 'TN': 'B', 'PA': 'R', 'VA': 'G', 'AK': 'R', 'AL': 'R', 'AR': 'R', 'VT': 'R', 'IL': 'B', 'IN': 'G', 'IA': 'R', 'AZ': 'B', 'ID': 'G', 'ME': 'R', 'MD': 'Y', 'MA': 'B', 'UT': 'Y', 'MO': 'G', 'MN': 'B', 'MI': 'R', 'MT': 'B', 'MS': 'G'}

Total puzzle conflicts: 406
Total assignments: 51
Execution time of USA map coloring : 0.011 seconds

FRANCE
Map coloring result: {'NH': 'R', 'MP': 'Y', 'EA': 'R', 'NO': 'R', 'LO': 'G', 'CA': 'R', 'BO': 'G', 'AL': 'R', 'CE': 'Y', 'LI': 'G', 'AQ': 'B', 'PC': 'R', 'FC': 'B', 'AU': 'R', 'RA': 'Y', 'BR': 'R', 'LR': 'G', 'PI': 'G', 'NB': 'G', 'PL': 'B', 'IF': 'B'}

Total puzzle conflicts: 267
Total assignments: 21
Execution time of France map coloring : 0.004 seconds
>>> |
Ln: 38 Col: 4
```

```
Python Shell
File Edit Shell Debug Options Windows Help
Execution time of France map coloring : 0.004 seconds
>>> ----- RESTART -----
>>>
Switch the algorithm you want to use for finding a solution for map coloring Australia, USA and France :
1. FC+MRV (forward checking search with the heuristic mechanism MRV)
2. MAC (maintaining arc consistency)
3. Min-Conflicts (minimum conflicts)

Type : 2
Maintaining arc consistency (MAC) is selected

Map Coloring solution:

AUSTRALIA
Map coloring result: {'WA': 'R', 'Q': 'R', 'T': 'R', 'V': 'R', 'SA': 'G', 'NT': 'B', 'NSW': 'B'}

Total puzzle conflicts: 68
Total assignments: 7
Execution time of Australia map coloring : 0.001 seconds

USA
Map coloring result: {'WA': 'R', 'DE': 'R', 'DC': 'R', 'WI': 'R', 'WV': 'R', 'HI': 'R', 'FL': 'R', 'WY': 'R', 'NH': 'R', 'NJ': 'G', 'NM': 'R', 'TX': 'G', 'LA': 'R', 'NC': 'R', 'ND': 'R', 'NE': 'B', 'TN': 'B', 'NY': 'Y', 'PA': 'B', 'RI': 'R', 'NV': 'R', 'VA': 'G', 'CO': 'G', 'CA': 'Y', 'AL': 'Y', 'AR': 'Y', 'VT': 'G', 'IL': 'B', 'GA': 'G', 'IN': 'R', 'IA': 'G', 'OK': 'B', 'AZ': 'B', 'ID': 'B', 'CT': 'G', 'ME': 'G', 'MD': 'Y', 'KA': 'Y', 'MA': 'B', 'OH': 'G', 'UT': 'Y', 'MO': 'R', 'MN': 'B', 'MI': 'B', 'AK': 'R', 'MT': 'G', 'MS': 'G', 'SC': 'Y', 'KY': 'Y', 'OR': 'G', 'SD': 'Y'}

Total puzzle conflicts: 29484
Total assignments: 527
Execution time of USA map coloring : 0.123 seconds

FRANCE
Map coloring result: {'NH': 'R', 'NO': 'R', 'FC': 'Y', 'LO': 'R', 'CA': 'B', 'BO': 'R', 'RA': 'B', 'AL': 'G', 'CE': 'G', 'LI': 'R', 'AQ': 'G', 'PC': 'B', 'FA': 'R', 'AU': 'Y', 'MP': 'B', 'BR': 'R', 'LR': 'G', 'PI': 'G', 'NB': 'B', 'PL': 'Y', 'IF': 'Y'}

Total puzzle conflicts: 1140
Total assignments: 22
Execution time of France map coloring : 0.003 seconds
>>>
Ln: 73 Col: 4
```

```

Python Shell
File Edit Shell Debug Options Windows Help
Execution time of France map coloring : 0.003 seconds
>>> ----- RESTART -----
>>>
Switch the algorithm you want to use for finding a solution for map coloring Australia, USA and France :
1. FC+MRV (forward checking search with the heuristic mechanism MRV)
2. MAC (maintaining arc consistency)
3. Min-Conflicts (minimum conflicts)

Type : 3
Minimum conflicts (Min-Conflicts) is selected

Map Coloring solution:

AUSTRALIA
Map coloring result: {'WA': 'B', 'Q': 'B', 'T': 'R', 'V': 'B', 'SA': 'R', 'NT': 'G', 'NSW': 'G'}

Total puzzle conflicts: 42
Total assignments: 12
Execution time of Australia map coloring : 0.002 seconds

USA
Map coloring result: {'WA': 'R', 'DE': 'Y', 'DC': 'B', 'WI': 'G', 'NV': 'Y', 'HI': 'R', 'FL': 'R', 'WY': 'Y', 'NH': 'R', 'NJ': 'G', 'NM': 'B', 'TX': 'G', 'LA': 'R', 'NC': 'B', 'ND': 'G', 'NE': 'R', 'TN': 'R', 'NY': 'R', 'PA': 'B', 'RI': 'G', 'NV': 'B', 'VA': 'G', 'CO': 'G', 'CA': 'G', 'AL': 'G', 'AR': 'Y', 'VT': 'B', 'IL': 'R', 'GA': 'Y', 'IN': 'Y', 'IA': 'Y', 'OK': 'R', 'AZ': 'Y', 'ID': 'G', 'CT': 'B', 'ME': 'Y', 'MD': 'R', 'KA': 'B', 'MA': 'Y', 'OH': 'R', 'UT': 'R', 'MO': 'G', 'MN': 'R', 'MI': 'B', 'AK': 'Y', 'MT': 'R', 'MS': 'B', 'SC': 'R', 'KY': 'B', 'OR': 'Y', 'SD': 'B'}

Total puzzle conflicts: 1326
Total assignments: 76
Execution time of USA map coloring : 0.007 seconds

FRANCE
Map coloring result: {'NH': 'B', 'NO': 'B', 'FC': 'R', 'LO': 'Y', 'CA': 'G', 'BO': 'B', 'RA': 'G', 'AL': 'G', 'CE': 'R', 'LI': 'B', 'AQ': 'R', 'PC': 'G', 'PA': 'Y', 'AU': 'Y', 'MP': 'G', 'BR': 'B', 'LR': 'R', 'PI': 'R', 'NB': 'G', 'PL': 'Y', 'IF': 'Y'}

Total puzzle conflicts: 609
Total assignments: 49
Execution time of France map coloring : 0.004 seconds
>>> |
Ln: 108 Col: 4

```

Συγκεντρωτικά τα αποτελέσματα των εκτυπώσεων ανά μέτρο σύγκρισης είναι τα εξής:

Πίνακας συνολικών συγκρούσεων για έλεγχο συνέπειας

Map-coloring	FC+MRV	MAC	Min-Conflicts
Australia	30	68	42
USA	406	29484	1326
France	267	1140	609

Πίνακας συνολικών αναθέσεων τιμών

Map-coloring	FC+MRV	MAC	Min-Conflicts
Australia	7	7	12
USA	51	527	76
France	21	22	49

Πίνακας συνολικών χρόνων εκτέλεσης αλγορίθμων

Map-coloring	FC+MRV	MAC	Min-Conflicts
Australia	0.001 sec	0.001 sec	0.002 sec
USA	0.011 sec	0.123 sec	0.007 sec
France	0.004 sec	0.003 sec	0.004 sec

Όπως φαίνεται από τις παραπάνω εκτυπώσεις για το χάρτη της Αυστραλίας που είναι και ο ευκολότερος και οι τρεις αλγόριθμοι που χρησιμοποιήσαμε κατάφεραν και βρήκαν λύση που να ικανοποιεί τους περιορισμούς όπως ορίζονται στους κανόνες σε πάρα πολύ μικρό χρόνο. Εδώ παρατηρούμε ότι γενικά από τους τρεις αλγόριθμους υπερέχει ο *FC+MRV* παρόλο που υποθετικά είναι ο χειρότερος. Αυτό συμβαίνει γιατί ο πρώτος έλεγχος σε συνδυασμό με την ευρετική ελαχίστων εναπομενουσών τιμών ταιριάζει ακριβώς στο χρωματισμό του χάρτη της Αυστραλίας και αυτό γιατί και οι μεταβλητές είναι λίγες και το πεδίο τιμών δίνει λίγες επιλογές. Συνεπώς για αυτό το λόγο και αυτές οι επιλογές κατά πολύ μεγάλη πιθανότητα θα είναι και οι σωστές. Ο αλγόριθμος *MAC* όπως ήταν αναμενόμενο δεν υστερεί σε χρόνο και σε αναθέσεις τιμών αλλά έχει κάνει τις διπλάσιες συγκρίσεις κάτι που για αυτό το πρόβλημα ήταν περιττό. Ο αλγόριθμος *Min-Conflicts* και αυτός όπως ήταν αναμενόμενο δεν άργησε και όπως είναι λογικό δεν ήταν και ο καλύτερος γιατί βασίζεται σε τυχαία ανάθεση που σίγουρα βοηθάει περισσότερο σε μεγαλύτερους χάρτες.

Πηγαίνοντας τώρα στον αμέσως μεγαλύτερο χάρτη, αυτόν της Γαλλίας παρατηρούμε ότι και πάλι ο *MAC* αλγόριθμος αν και χρησιμοποιεί πιο γρήγορη ευρεστική είναι χειρότερος από τον *FC+MRV* όσον αφορά το σύνολο συγκρούσεων (αυτή τη φορά πενταπλάσιο) γιατί την χρησιμοποιεί περισσότερες φορές. Όσον αφορά το χρόνο και τις αναθέσεις είναι περίπου ισότιμοι και οι δύο. Ο αλγόριθμος *Min-Conflicts* αυτή τη φορά φαίνεται ότι είναι καλύτερος απ' όσο ήταν στο χάρτη της Αυστραλίας. Έχει κάνει τον ίδιο χρόνο με τους άλλους δύο αλγόριθμους και έχει τις μισές συγκρούσεις απ' ότι ο *MAC* αλλά τις διπλάσιες από τον *FC+MRV*, παρόλο που η ευρεστική του συνάρτηση είναι τέτοια ώστε να έχει τις ελάχιστες συγκρούσεις.

Στο δυσκολότερο χάρτη, σε αυτό της Αμερικής, παρατηρούμε ότι παρόλο που για μια ακόμη φορά ο *FC+MRV* έχει τις ελάχιστες τιμές συγκρούσεων και αναθέσεων, εδώ φαίνεται πόσο κοστίζει ο πρώτος έλεγχος σε χρόνο. Αυτό φαίνεται γιατί ο *Min-Conflicts* έχει τις τριπλάσιες συγκρούσεις και περίπου μισές παραπάνω αναθέσεις τιμών και όμως έχει αρκετά καλύτερο χρόνο από τον *FC+MRV*. Ένα ακόμα αξιοσημείωτο είναι ότι ο *MAC* αυξάνει σταθερά εκθετικά το σύνολο αναθέσεων και συγκρούσεων και συνεπώς μπορούμε πλέον να βγάλουμε το συμπέρασμα ότι παρόλο ο *MAC* θεωρείται καλύτερος από τον *FC+MRV* γιατί αναγνωρίζει πολύ γρηγορότερα τα αδιέξοδα, για το πρόβλημα του χρωματισμού χαρτών κρίνεται ακατάλληλος και μη αποδοτικός μιας και ο χρόνος, οι συγκρούσεις και οι αναθέσεις αυξάνονται εκθετικά. Ο *FC+MRV* κρίνεται πολύ χρήσιμος για το χρωματισμό χαρτών καθώς και ο *Min-Conflicts* αν και ο δεύτερος δεν έχει πάντα σταθερό αποτέλεσμα.