ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΙΣΤΟΡΙΑΣ-ΑΡΧΑΙΟΛΟΓΙΑΣ

ΣΧΕΔΙΑΣΗ ΚΑΙ ΑΝΑΛΥΣΗ ΑΛΓΟΡΙΘΜΩΝ

ΔΙΔΑΚΤΙΚΗ ΕΝΟΤΗΤΑ 10

ΘΕΜΑ: ΤΕΧΝΙΚΕΣ ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΑΝΑΛΥΣΗΣ ΑΛΓΟΡΙΘΜΩΝ ΓΙΑ ΠΡΟΒΛΗΜΑΤΑ ΜΕ ΑΠΑΓΟΡΕΥΤΙΚΟ ΑΡΙΘΜΟ ΠΕΡΙΠΤΩΣΕΩΝ

Δρ. Δημήτριος Κ. Κουκόπουλος
Αναπληρωτής Καθηγητής

ΠΕΡΙΕΧΟΜΕΝΟ

- Προβλήματα με Απαγορευτικό Αριθμό Περιπτώσεων
- Η Μέθοδος της Απληστίας
- Επίλυση Προβλημάτων με τη Μέθοδο της Απληστίας
  - Πρόβλημα εύρεσης μεγίστου συνάρτησης
  - Πρόβλημα κατάταξης έργων με προθεσμίες
  - Πρόβλημα πλανόδιου πωλητή
- Παραδείγματα

ΠΡΟΒΛΗΜΑΤΑ ΜΕ ΑΠΑΓΟΡΕΥΤΙΚΟ ΑΡΙΘΜΟ ΠΕΡΙΠΤΩΣΕΩΝ

- **Ορισμός**

  Προβλήματα με απαγορευτικό αριθμό περιπτώσεων είναι εκείνα όπου δεν είναι δυνατό να κατασκευασθούν ή να εξετασθούν όλες οι λύσεις /περιπτώσεις, ώστε να επιλεγεί μια λύση που να ικανοποιεί ορισμένες προϋποθέσεις.

- Είναι προβλήματα που το αντίστοιχό τους πρόβλημα απόφασης ανήκει στα NP-πλήρη.

- Τα προβλήματα που εξετάζουμε έχουν δεκτές υποψήφιες λύσεις, αλλά ψάχνουμε τη βέλτιστη. Η σύγκριση των υποψηφίων λύσεων γίνεται βάσει κριτηρίων.

- Σε προβλήματα όπου το ζητούμενο δεν είναι η εύρεση της βέλτιστης λύσης, αλλά μιας οποιαδήποτε δεκτής λύσης κάνουμε χρήση **ευρετικών** αλγορίθμων. Οι ευρετικοί αλγόριθμοι είναι προσεγγιστικοί, και συνήθως δίνουν λύση κοντά στη βέλτιστη, εύκολα και γρήγορα.

Η ΜΕΘΟΔΟΣ ΤΗΣ ΑΠΛΗΣΤΙΑΣ

Εφαρμογή μεθόδου: Ζητείται βέλτιστη λύση ανάμεσα σε απαγορευτικό αριθμό πριπτώσεων.

Μέθοδος: Μεγιστοποίηση κέρδους σε κάθε βήμα χωριστά, χωρίς εξέταση των επιπτώσεων που αυτό μπορεί να έχει στα επόμενα βήματα και επομένως στο τελικό αποτέλεσμα.

Κριτήριο: Απόκτηση του μέγιστου κέρδους σε κάθε βήμα. Σε κάθε βήμα απορρίπτεται μεγάλος αριθμός υποψήφιων λύσεων περιορίζοντας έτσι τον αριθμό τους.

Τοπική μέθοδος: Δεν παίρνει υπόψη τη συνολική εικόνα. Έτσι, δε δίνει πάντα το σωστό αποτέλεσμα, αλλά ένα αποτέλεσμα κοντά στο βέλτιστο.

ΠΡΟΒΛΗΜΑ ΕΥΡΕΣΗΣ ΜΕΓΙΣΤΟΥ ΣΥΝΑΡΤΗΣΗΣ 1/2

Πρόβλημα: Ζητείται να βρεθεί το μέγιστο για τη συνάρτηση

$$\boldsymbol{\varphi(x)=-6x^5+45x^4-110x^3+90x^2}$$

Λύση

- Ο αλγόριθμος επιλέγει ένα σημείο εκκίνησης, εδώ $\mathbf{x_0=1,5}$.
- Σε κάθε σημείο ελέγχεται αν η συνάρτηση είναι αύξουσα ή φθίνουσα. Αν είναι αύξουσα επιλέγεται το επόμενο σημείο προς τα δεξιά, αλλιώς προς τα αριστερά με κάποιο προκαθορισμένο τρόπο.
- Εφαρμόζεται η τεχνική της απληστίας. Έτσι, σε κάθε βήμα αυξάνεται το κέρδος (ανηφορίζω) μέχρι αυτό να μην είναι πλέον δυνατό, οπότε ο αλγόριθμος τερματίζει.

ΠΡΟΒΛΗΜΑ ΕΥΡΕΣΗΣ ΜΕΓΙΣΤΟΥ ΣΥΝΑΡΤΗΣΗΣ 2/2

- Ανηφορίζοντας από το σημείο εκκίνησης ο αλγόριθμος απληστίας δίνει για **x=1, φ(1)=19** μέγιστο.

- Ένας λιγότερο τοπικός αλγόριθμος θα κατηφόριζε δεξιά από το $\mathbf{x_0}$ παραβαίνοντας την αρχή μεγιστοποίησης του κέρδους σε κάθε βήμα. Αυτό θα οδηγούσε σε χαμηλότερο σημείο και θα κατέληγε τελικά στο αληθινό μέγιστο της συνάρτησης **φ**, που είναι για **x=3, φ(3)=27**.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ

Πρόβλημα:

Δίνονται **n** έργα $\mathbf{E_1,\ldots,E_n}$. Σε κάθε έργο $\mathbf{E_i}$ αντιστοιχεί μια προθεσμία $\mathbf{d_i>0}$, που είναι ένας ακέραιος αριθμός χρονικών μονάδων, όπως και ένα κέρδος $\mathbf{p_i>0}$ που αποκτάται μόνο όταν το έργο εκτελείται μέσα στην προθεσμία του. Η εκτέλεση ενός έργου απαιτεί μια χρονική μονάδα. Ποια έργα και με ποια σειρά πρέπει να εκτελεστούν, ώστε να μεγιστοποιηθεί το συνολικό κέρδος;

Αν **Λ** μια εφικτή λύση τότε το συνολικό κέρδος είναι $k = \sum_{\mathrm{E}_i \in \Lambda} p_i$

ΠΑΡΑΔΕΙΓΜΑ

Πρόβλημα

Έστω τρία έργα $\mathbf{E_1, E_2, E_3}$ με προθεσμίες $\mathbf{(d_1, d_2, d_3)=(2, 2, 1)}$ και κέρδη $\mathbf{(p_1, p_2, p_3)=(50, 100, 60)}$**.** Ποια έργα και με ποια σειρά πρέπει να εκτελεστούν, ώστε να μεγιστοποιηθεί το συνολικό κέρδος;

Απάντηση

Εφικτή λύση: Εκτελούνται τα έργα $\mathbf{E_1, E_2}$ με κέρδος 150.

Βέλτιστη λύση: Εκτελούνται τα έργα $\mathbf{E_3, E_2}$ με κέρδος 160.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΑΛΓΟΡΙΘΜΟΣ ΕΠΙΛΥΣΗΣ

Βήματα:

1. Έστω αλγόριθμος απληστίας που αρχίζει με **Λ={}** και σε κάθε βήμα επιλέγει έργο με μεγαλύτερο δυνατό κέρδος και το εισάγει στο **Λ**, εφόσον αυτό συνεχίζει να δίνει μια εφικτή λύση.

2. Πρέπει να ελεγχθούν όλες οι δυνατές μεταθέσεις των έργων στο **Λ** για να πιστοποιηθεί ότι η προσαυξημένη λύση είναι η σωστή. Αλλά, αν έχω **n** έργα υπάρχουν **n!** δυνατές μεταθέσεις, γεγονός απαγορευτικό για έλεγχο. Όμως, από Λήμμα 1 αρκεί να ελεγχθεί μόνο η μετάθεση με αύξουσα σειρά προθεσμιών.

3. Αν η σειρά εκτέλεσης που καθορίζεται από την μετάθεση με αύξουσα σειρά προθεσμιών παραβιάζει κάποια προθεσμία το νέο έργο δε μπαίνει στη λύση και από τα υπόλοιπα έργα επιλέγεται πάλι αυτό που δίνει το μέγιστο κέρδος.

4. Ο αλγόριθμος τερματίζει όταν δεν υπάρχουν άλλα προς εξέταση έργα.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΛΗΜΜΑ 1 (1/2)

Λήμμα 1

Μια λύση **Λ** για το πρόβλημα της κατάταξης έργων με προθεσμίες είναι εφικτή αν και μόνο αν η εκτέλεση των έργων της **Λ** με αύξουσα σειρά προθεσμιών μπορεί να γίνει χωρίς παραβίαση των προθεσμιών.

Απόδειξη

- Αν η εκτέλεση των έργων μιας λύσης **Λ** με αύξουσα σειρά προθεσμιών μπορεί να γίνει χωρίς παραβίαση των προθεσμιών, τότε προφανώς η **Λ** είναι μια εφικτή λύση για το πρόβλημα της κατάταξης έργων με προθεσμίες.
- Για το άλλο σκέλος της λύσης, υποθέτω ότι η **Λ** είναι εφικτή, δηλαδή υποθέτω ότι υπάρχει μια σειρά εκτέλεσης έργων $\mathbf{E_1,\ldots,E_k}$ χωρίς παραβίαση προθεσμιών, δηλαδή $d_1 \leq d_2 \leq \cdots \leq d_k$.
- Αν $\mathbf{E_{i1},\ldots,E_{ik}}$ είναι μια άλλη σειρά εκτέλεσης των ίδιων έργων με αύξουσες προθεσμίες, δηλαδή αν $d_{i1} \leq d_{i2} \leq \cdots \leq d_{ik}$ πρέπει να δείξω ότι και αυτή η σειρά εκτέλεσης έργων δεν παραβιάζει τις προθεσμίες.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΛΗΜΜΑ 1 (2/2)

- Αν **im=m, m=1,…,k**, δηλαδή αν οι δυο σειρές εκτέλεσης έργων συμπίπτουν μέχρι το δείκτη **m**, έστω **iα** ο πρώτος δείκτης όπου οι δυο σειρές διαφέρουν. Τότε, ο δείκτης **iα** θα είναι ένας από τους δείκτες μετά τον **α**, **iα>α,** στη σειρά εκτέλεσης της εφικτής λύσης.

- Αλλάζω τη σειρά των έργων $\mathbf{E_{\alpha}}$ και $\mathbf{E_{i\alpha}}$ στην εφικτή λύση και επειδή $d_{i\alpha} \leq d_{a}$ οι προθεσμίες δεν παραβιάζονται.

- Τώρα, έχουμε δυο σειρές εκτέλεσης έργων όπου αυτή που δεν παραβιάζει προθεσμίες και αυτή που έχει αύξουσα σειρά προθεσμιών συμπίπτουν και στο δείκτη **iα.**

- Συνεχίζοντας με τον ίδιο τρόπο θα συμπίπτουν τελικά σε όλους τους δείκτες.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΘΕΩΡΗΜΑ

Θεώρημα

Ο αλγόριθμος απληστίας σε σχέση με τις προθεσμίες δίνει πάντα μια βέλτιστη λύση στο πρόβλημα της κατάταξης έργων με προθεσμίες.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΑΠΟΔΕΙΞΗ ΘΕΩΡΗΜΑΤΟΣ (1/3)

- Έστω **Λ** η λύση που δίνει ο αλγόριθμος απληστίας και **Μ** μια βέλτιστη λύση. Αν **Λ**=**Μ** τέλος. Άρα, έστω **Λ != Μ.**
- Αρκεί να δείξω ότι μπορώ να επιφέρω μερικές αλλαγές στην **Μ** για να πάρω την **Λ** χωρίς να ελαττωθεί το συνολικό κέρδος που δίνει η **Μ.**
- Έστω τυχόν κοινά έργα των **Λ** και **Μ,** $\mathrm{E}_i \in \Lambda \cap \mathrm{M}$.
- Έστω **[t, t+1]** και **[t', t'+1]** η χρονική μονάδα εκτέλεσης του $\mathbf{E_i}$ στην **Λ** και **Μ** αντίστοιχα.
- Αν **t < t'** μεταθέτω τη χρονική στιγμή εκτέλεσης του $\mathbf{E_i}$ στην **Λ** από την **t** στην **t'.** Όποιο έργο της **Λ** έχει σχεδιαστεί για εκτέλεση στην **t'** μεταφέρεται στην **t.**
- Αν **t > t'** κάνω τις ίδιες μεταβολές στην **Μ**.
- Η νέα σειρά εκτέλεσης για τις **Λ** και **Μ** δεν παραβιάζει προθεσμίες. Επομένως, τα κοινά έργα των **Λ** και **Μ** εκτελούνται στις ίδιες χρονικές περιόδους.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΑΠΟΔΕΙΞΗ ΘΕΩΡΗΜΑΤΟΣ (2/3)

- Εκτός από **Λ!=Μ** ισχύει ότι $\Lambda \not\subset M$ και $M \not\subset \Lambda$.
- Αν $M \subset \Lambda$ η **Μ** δεν είναι βέλτιστη.
- Αν $\Lambda \subset M$ τότε ο αλγόριθμος απληστίας δε θα είχε τερματιστεί.
- Άρα υπάρχουν έργα που ανήκουν στην **Λ** και όχι στην **Μ** και αντίστροφα.
- Από τα έργα στο **Λ-Μ** διαλέγω ένα με μέγιστο κέρδος $\mathbf{E_a}$.
- Για όλα τα έργα $\mathbf{E_b}$ του **Μ-Λ** ισχύει $\mathbf{p_a>p_b}$, διαφορετικά η μέθοδος της απληστίας θα είχε εισάγει το $\mathbf{E_b}$ στην **Λ** πριν το $\mathbf{E_a}$.
- Το χρονικό διάστημα εκτέλεσης του $\mathbf{E_a}$ στην **Λ** είναι **[t, t+1].**
- Στο διάστημα **[t, t+1]** εκτελείται και κάποιο έργο $\mathbf{E_b}$ στην **Μ** που δεν ανήκει στην **Λ**, αφού κοινά έργα εκτελούνται σε κοινούς χρόνους.

ΠΡΟΒΛΗΜΑ ΚΑΤΑΤΑΞΗΣ ΕΡΓΩΝ ΜΕ ΠΡΟΘΕΣΜΙΕΣ-ΑΠΟΔΕΙΞΗ ΘΕΩΡΗΜΑΤΟΣ (3/3)

- Αν διώξουμε το $\mathbf{E_b}$ από τη **M** και βάλουμε το $\mathbf{E_a}$ δεν ελαττώνεται το συνολικό κέρδος της **M** και οι δυο λύσεις έχουν ένα ακόμα κοινό έργο, το $\mathbf{E_a}$.
- Συνεχίζοντας έτσι καταλήγουμε στην **Λ** χωρίς μείωση του συνολικού κέρδους της **M**. Επομένως, η λύση **Λ** είναι και αυτή βέλτιστη.

ΠΡΟΒΛΗΜΑ ΠΛΑΝΟΔΙΟΥ ΠΩΛΗΤΗ

Πρόβλημα

Ένας πωλητής πρέπει να επισκεφτεί **n** πόλεις, την καθεμιά ακριβώς μια φορά, αρχίζοντας με μια από αυτές (πόλη-βάση) και επιστρέφοντας σε αυτή. Δίνεται το κόστος μετάβασης από κάθε πόλη σε κάθε άλλη και ζητείται το πρόγραμμα του ταξιδιού, ώστε το συνολικό κόστος να είναι ελάχιστο.

ΠΡΟΒΛΗΜΑ ΠΛΑΝΟΔΙΟΥ ΠΩΛΗΤΗ-ΑΛΓΟΡΙΘΜΟΣ

Είσοδος: Αριθμός πόλεων n και πίνακας κόστους c(i, j) όπου i, j=0,…, n-1

Έξοδος: Πίνακας σειράς επίσκεψης πόλεων Round μεγέθους n+1 και συνολικό κόστος cost.

```
/*Αρχικοποίηση*/
Round(0)=0
cost=0
Visit(0)=1
count=0
/*Καθορισμός γύρου και κόστους*/
for i=1 to n-1 do
    Επιλογή δείκτη j όπου min=c(count, j)=min{c(count, k); Visit(k)=0 και k=0,…,n-1}
    Visit(j)=1
    cost=cost+min
    count=j
    Round(i)=j
end_for
Round(n)=0
Cost=cost+c(count,0)
```

Δρ. Δημήτριος Κ. Κουκόπουλος
Αναπληρωτής Καθηγητής

ΠΡΟΒΛΗΜΑ ΠΛΑΝΟΔΙΟΥ ΠΩΛΗΤΗ-ΠΑΡΑΔΕΙΓΜΑ

Πρόβλημα: Να λυθεί το πρόβλημα του πλανόδιου πωλητή, όταν υπάρχουν n=5 πόλεις αριθμημένες από 0 έως 4 (0 είναι α αριθμός της πόλης-βάσης) και το κόστος επίσκεψης c(i, j) μεταξύ δυο πόλεων με αριθμούς i και j δίνεται από τον παρακάτω πίνακα:

$$C = \begin{bmatrix} 1000 & 2 & 1000 & 5 & 1 \\ 3 & 1000 & 1 & 5 & 1000 \\ 3 & 2 & 1000 & 5 & 1 \\ 4 & 3 & 1 & 1000 & 10 \\ 2 & 3 & 4 & 1 & 1000 \end{bmatrix}$$

ΑΠΑΝΤΗΣΗ

1. Επιλέγουμε j=4 γιατί c(0, 4)<c(0, k) για k=0,…, 4. Τότε count=4, Visit(4)=1, cost=1 και Round(1)=4.
2. Επιλέγουμε j=3 γιατί c(4, 3)<c(4, k) για k=0,…, 4. Τότε count=3, Visit(3)=1, cost=2 και Round(2)=3.
3. Επιλέγουμε j=2 γιατί c(3, 2)<c(3, k) για k=0,…, 4. Τότε count=2, Visit(2)=1, cost=3 και Round(3)=2.
4. Επιλέγουμε j=1 γιατί c(2, 1)<c(2, k) για k=0,…, 4 και Visit(k)=0. Τότε count=1, Visit(1)=1, cost=5 και Round(4)=1.
5. Round(5)=0, cost=8.

Τελικά, το **πρόγραμμα επίσκεψης** των 5 πόλεων είναι: 0, 4, 3, 2, 1, 0.
Το **συνολικό κόστος** είναι 8.