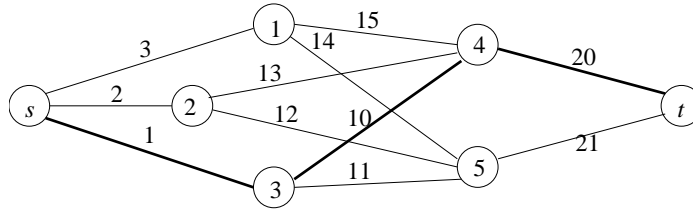


Η μέθοδος της 'Απληστίας' (The Greedy Method)

Προβλήματα βελτιστοποίησης

- Τα προβλήματα αυτά μπορεί να έχουν παραπάνω από μία δυνατές λύσεις.
- Κάθε λύση έχει μία τιμή. Επιθυμούμε να υπολογίσουμε τη λύση με την βέλτιστη (*optimal*) (μέγιστη ή ελάχιστη) τιμή.

Παράδειγμα Να βρεθεί το ελάχιστο μονοπάτι από τον κόμβο s στον κόμβο t .



Μονοπάτι

$$s - 2 - 4 - t$$

$$s - 1 - 5 - 2 - 4 - t$$

$$s - 3 - 4 - t$$

Κόστος

$$2 + 13 + 20 = 35$$

$$3 + 14 + 12 + 13 + 20 = 62$$

$$1 + 10 + 20 = 31$$

Ένας **άπληστος** αλγόριθμος πάντα προτιμά την επιλογή που μοιάζει να είναι η καλύτερη τη δεδομένη αυτή στιγμή.

Προτιμά μια τοπικά βέλτιστη επιλογή με την ελπίδα ότι η εκλογή αυτή θα τον οδηγήσει σε μια γενική βέλτιστη λύση.

ΣΗΜΕΙΩΣΗ Όταν ισχυριζόμαστε ότι ένας άπληστος αλγόριθμος υπολογίζει μία βέλτιστη λύση για κάποιο πρόβλημα βελτιστοποίησης, **πρέπει να αποδεικνύουμε τον ισχυρισμό μας.**

Θα μελετήσουμε:

- Πρόβλημα επιλογής δραστηριοτήτων (activity selection problem).
- Κώδικες Huffman.
- Τον αλγόριθμο του Prim για τη δημιουργία ελάχιστων διασυνδεδετικών δένδρων (minimum spanning trees).
- Τον αλγόριθμο του Dijkstra για εύρεση ελάχιστων μονοπατιών από έναν κόμβο-αφετηρία (single source shortest paths).

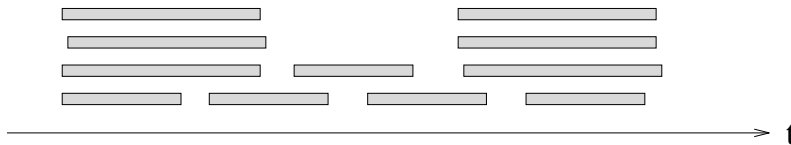
Μία δεύτερη άπληστη προσέγγιση ...

Από το σύνολο των συμβατών με τις ήδη επιλεγμένες δραστηριότητες, επέλεξε τη δραστηριότητα που επικαλύπτεται με τον μικρότερο αριθμό δραστηριοτήτων του συνόλου αυτού.

- Δεν παράγει τη σωστή λύση για το στιγμιότυπο:



- Δεν παράγει τη σωστή λύση για το στιγμιότυπο:



Μία 'άπληστη' λύση

Υποθέστε ότι οι δραστηριότητες είναι διατεταγμένες σε αύξουσα σειρά με βάση τη χρονική στιγμή περάτωσης τους, δηλ., $f_1 \leq f_2 \leq \dots \leq f_n$. (Εάν δεν είναι, τις ταξινομούμε σε $O(n \log n)$ χρόνο.)

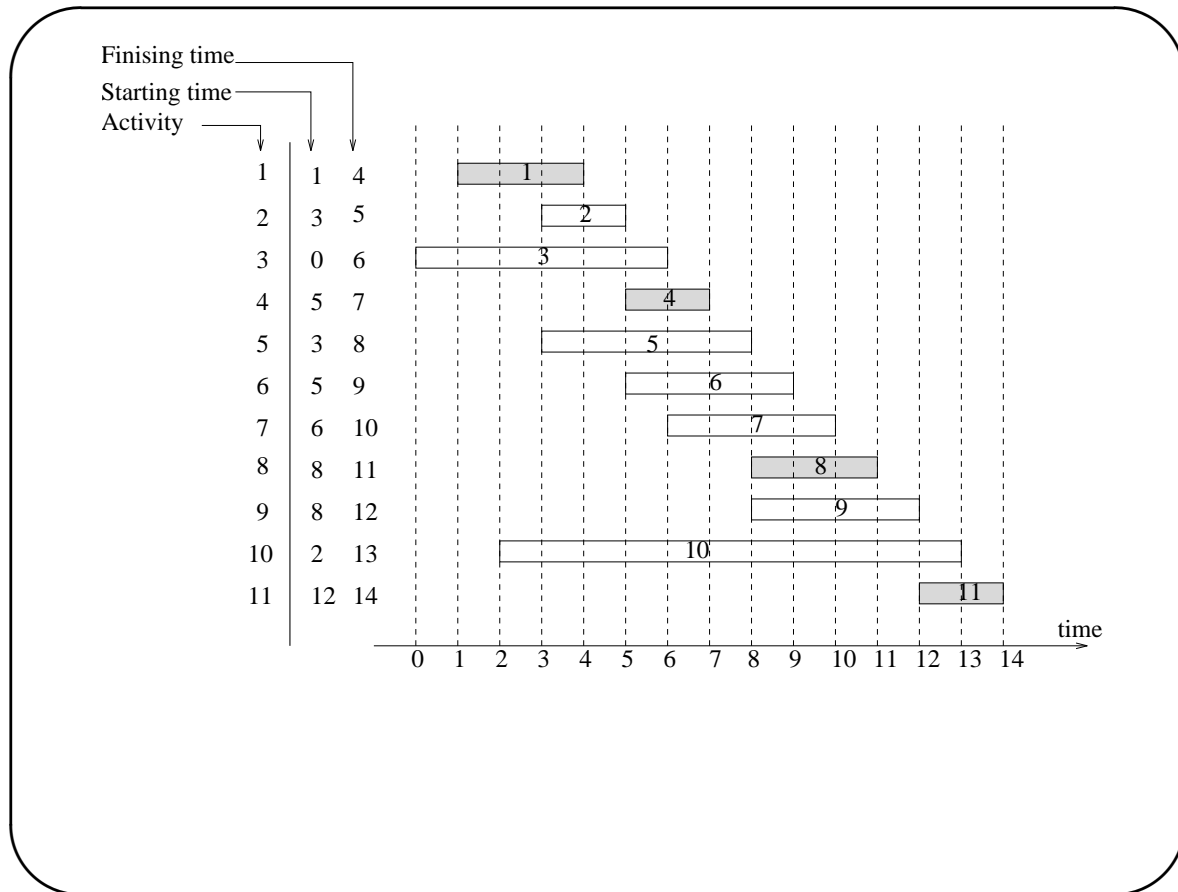
Από όλες τις συμβατές με τις ήδη επιλεγμένες δραστηριότητες, επέλεξε τη δραστηριότητα με τη μικρότερη χρονική στιγμή περάτωσης.

```

Greedy_activity_selection(s, f)
n = length(s)
A = {1}
j = 1
for i = 2 to n do
    if  $s_i \geq f_j$  then
        A = A ∪ {i}
        j = i
return A
    
```

- Η άπληστη φύση του αλγόριθμου υποδηλώνεται με την εκλογή της δραστηριότητας η οποία μεγιστοποιεί το ποσό του χρόνου που μπορεί να διατεθεί στο μέλλον σε άλλες δραστηριότητες.
- Το σύνολο A περιέχει τις επιλεγέντες δραστηριότητες.
- Το j υποδηλώνει τη πιο πρόσφατη δραστηριότητα που προστέθηκε στο A .

Πολυπλοκότητα: $\Theta(n)$



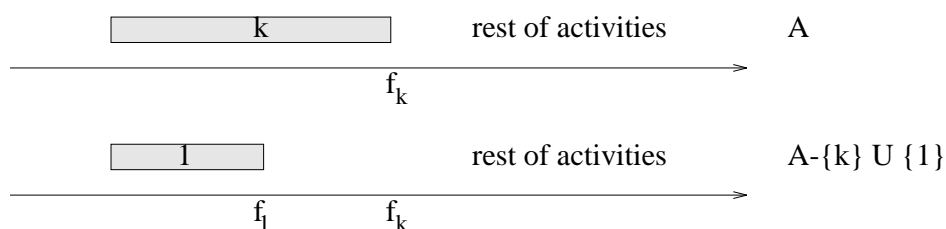
Απόδειξη της ορθότητας του άπληστου αλγορίθμου

Θεώρημα 1 Ο αλγόριθμος *Greedy_activity_selection* παράγει μέγιστου μεγέθους λύσεις για το πρόβλημα επιλογής δραστηριοτήτων.

Λήμμα 1 Υπάρχει μία βέλτιστη λύση η οποία ξεκινά χρονικά με την άπληστη εκλογή, δηλ., τη δραστηριότητα 1.

Απόδειξη Έστω $A \subseteq S$ μία βέλτιστη λύση του προβλήματος επιλογής δραστηριοτήτων.

- Διέταξε τις δραστηριότητες του A με βάση τη χρονική στιγμή περάτωσής τους.
- Έστω k η πρώτη δραστηριότητα.
- Εάν $k = 1$ έχουμε αποδείξει το ζητούμενο.
- Εάν $k \neq 1$ τότε το $A - \{k\} \cup \{1\}$ είναι επίσης μία βέλτιστη λύση ($f_1 \leq f_k$ και $\{1\} \notin A$).



□

Λήμμα 2 Εάν το A είναι μία βέλτιστη λύση του αρχικού προβλήματος S , τότε το $A' = A - \{1\}$ είναι μία βέλτιστη λύση του προβλήματος επιλογής δραστηριοτήτων $S' = \{i \in S : s_i \geq f_1\}$.

Απόδειξη

- Έστω ότι το A περιέχει m δραστηριότητες. \implies Το A' περιέχει $m - 1$ δραστηριότητες.
- Υποθέστε ότι το A' δεν είναι βέλτιστο, δηλ., υπάρχει ένα σύνολο B από συμβατές δραστηριότητες το οποίο περιέχει περισσότερες δραστηριότητες από το A' , $|B| > m - 1$.

Τότε, το $B \cup \{1\}$ είναι επίσης μία λύση για το S και επιπλέον $|B \cup \{1\}| > m$. Άρα, το A δεν είναι βέλτιστο. Αυτό οδηγεί σε άτοπο γιατί αντιφάσκει με την υπόθεσή ότι το A είναι βέλτιστο.

□

Απόδειξη του Θεωρήματος 1 (Με επαγωγή)

- Βάση: Λήμμα 1
- Επαγωγικό βήμα: Λήμμα 2

Μετά από κάθε άπληστη επιλογή μίας δραστηριότητας, έχουμε να επιλύσουμε ένα πρόβλημα ίδιας φύσης με το αρχικό (αλλά μικρότερου μεγέθους).

Με επαγωγή ως προς τον αριθμό των επιλογών που εκτελεί ο αλγόριθμος, αποδεικνύουμε ότι η άπληστη επιλογή της επόμενης δραστηριότητας οδηγεί σε μία βέλτιστη λύση.

□