

## Δήλωση προβλήματος

Σε αυτή την άσκηση θα σχεδιάσετε διαφορετικούς αλγορίθμους για να λύσετε το περίφημο πρόβλημα Knapsack. Σας παρέχεται σακίδιο με περιορισμένο χώρο και μια συλλογή από αντικείμενα με διαφορετικές τιμές και βάρη. Ο στόχος σας είναι να μεγιστοποιήσετε την αξία των στοιχείων που συσκευάζονται χωρίς να υπερβείτε τη συνολική χωρητικότητά του.

## Data Format

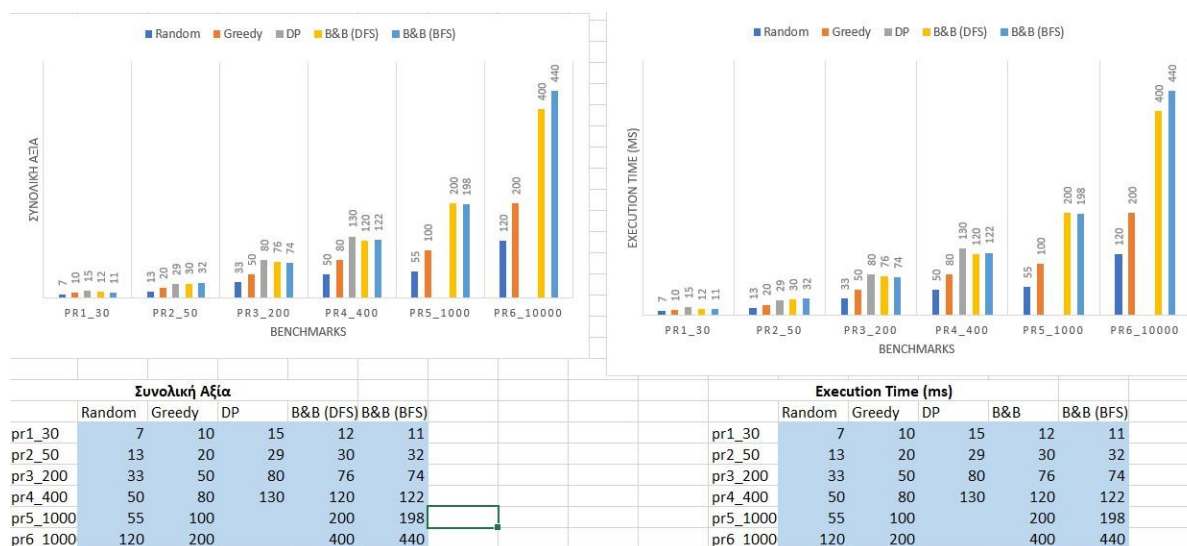
Κάθε πρόβλημα αναπαρίσταται από ένα αρχείο χωρισμένο σε γραμμές. Το πλήθος των γραμμών είναι  $n+1$ , όπου  $n$  το πλήθος των διαθέσιμων αντικειμένων.

Η πρώτη γραμμή διαθέτει 2 ακέραιους. Ο πρώτος υποδεικνύει τη συνολική χωρητικότητα του σακιδίου ενώ ο δεύτερος το πλήθος των αντικειμένων. Οι υπόλοιπες  $n$  γραμμές είναι ζεύγη χωρητικότητας, αξίας αντικειμένου.

## Βαθμολόγηση

- Η ποιότητα των τελικών λύσεων.
- Ο χρόνος εκτέλεσης.
- Η ορθότητα της κωδικοποίησης.

Τα αποτελέσματα σας θα πρέπει να τοποθετηθούν στο excel **solutions.xls**. Αρχικά θα σβήσετε και στη συνέχεια θα συμπληρώσετε τα δεδομένα που βρίσκονται στις μπλε περιοχές του excel. Οι τιμές που παρουσιάζονται στην Εικόνα 1 είναι τυχαίες.



Εικόνα 1. Excel συμπλήρωσης αποτελεσμάτων

## Προγραμματιστικές τεχνικές

- Random: Τυχαία επιλογή στοιχείων μέχρι να γεμίσει το σακίδιο.
- Greedy: Άπληστη τεχνική (αξία/βάρος)
- DP: Δυναμικός Προγραμματισμός
- B&B (DFS): Branch & Bound με Depth First Search
- B&B (BFS): Branch & Bound με Best First Search
- (Bonus) B&B (LDS): Limited Discrepancy Search

Τα αρχεία pr1.txt, pr2.txt, pr3.txt, pr4.txt, pr5.txt, pr6.txt (Τελικές λύσεις) Αποτελούνται από 5 γραμμές. Κάθε γραμμή διαθέτει ένα σύνολο από 0-1 διαχωρισμένα με κόμμα και αντιπροσωπεύει την τελική λύση για κάθε προγραμματιστική τεχνική. 0 το αντικείμενο δεν επιλέγεται, 1 το αντικείμενο επιλέγεται. **Προσοχή!** η διάταξη των αντικειμένων θα πρέπει να παραμείνει η ίδια με την αρχική.