

A. Μελέτη Περίπτωσης:

Το πρόβλημα φόρτωσης της UPak

Η UPak αποτελεί θυγατρική εταιρεία μιας μεταφορικής εταιρείας με φορτηγά τύπου LTL.

Οι πελάτες της UPak προσφέρουν για κάθε φορτίο μια τιμή *σε χρηματικές μονάδες ανά κυβικό μέτρο* για τον όγκο που καταλαμβάνει. Η αποδοχή ή όχι της παραγγελίας προς μεταφορά επαφίεται στην UPak. Ο αριθμός των εν δυνάμει παραγγελιών είναι μια παράμετρος N του προβλήματος. Η τιμή που προσφέρουν οι πελάτες για κάθε φορτίο δίνεται στο διάνυσμα $Price[N]$ ενώ ο όγκος των φορτίων σε κυβικά μέτρα δίνεται στο διάνυσμα $Size[N]$.

Οι πελάτες φέρνουν τα φορτία τους στον τερματικό σταθμό της UPak για να φορτωθούν σε ρυμουλκούμενα οχήματα των οποίων η χωρητικότητα είναι πεπερασμένη. Να σημειωθεί ότι ο αριθμός των οχημάτων που διαθέτει η εταιρεία είναι μια παράμετρος T και οι χωρητικότητες των οχημάτων σε κυβικά μέτρα δίνονται στο διάνυσμα $Capacities[T]$. Επισημαίνεται ότι δεν επιτρέπεται η μερική φόρτωση μιας παραγγελίας. Με άλλα λόγια κάθε ένα φορτίο μπορεί να φορτωθεί το πολύ σε ένα όχημα.

Η διεύθυνση της UPak θέλει να αυτοματοποιήσει την διαδικασία επιλογής και ανάθεσης εκείνων των παραγγελιών στα διαθέσιμα οχήματα με τρόπο που να μεγιστοποιεί τα κέρδη της. Για τον σκοπό αυτό καλείστε να αναπτύξετε ένα μοντέλο Ακεραίου Προγραμματισμού και να προχωρήσετε στην υλοποίηση του με τη γλώσσα προγραμματισμού Python και της βιβλιοθήκης Pyomo.

Πιο αναλυτικά η μελέτη που έχετε αναλάβει χωρίζεται σε δύο μέρη που περιγράφονται αναλυτικά στις σελίδες [2-5].

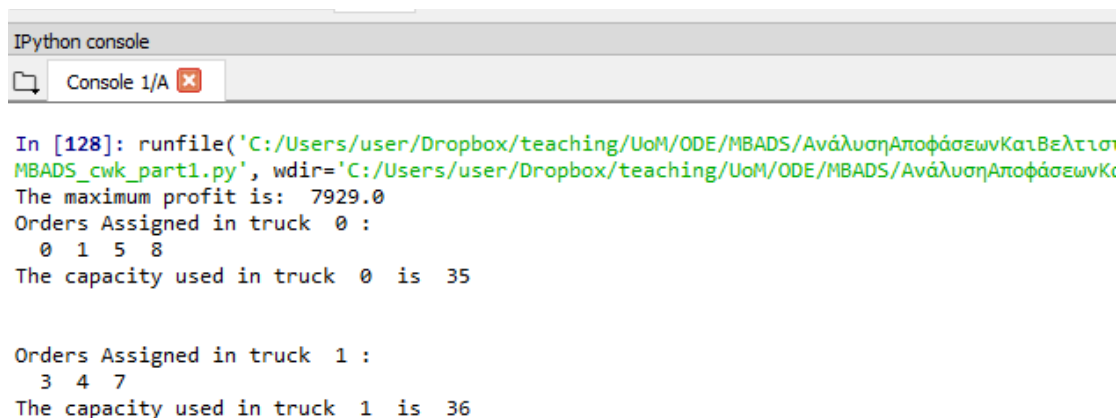
Μέρος 1

Για την μελέτη περίπτωσης που περιγράφηκε παραπάνω θα πρέπει να:

1. Να αναπτύξετε την αναλυτική μορφή του μοντέλου όπου θα περιγράφετε τα σύνολα μεταβλητών, περιορισμών και παραμέτρων του μοντέλου. (40%)
2. Να αναπτύξετε το μοντέλο βελτιστοποίησης σε Python/Pyomo. Η UPak για να σας διευκολύνει σας παρέχει τα δεδομένα του πίνακα 1, καθώς και το αντίστοιχο βέλτιστο πλάνο φόρτωσης που δίνεται στην εικόνα 1. Να σημειωθεί ωστόσο ότι ιδανικά ο κώδικας σας θα πρέπει να είναι κλιμακούμενος (scalable), ικανός δηλαδή να αντιμετωπίσει προβλήματα διαφορετικών διαστάσεων.
Ο κώδικας θα πρέπει να επιστρέφει το βέλτιστο πλάνο φόρτωσης και το αντίστοιχο μέγιστο κέρδος. (40%)

Number of Trucks :	2
Number of Items:	10
Capacity of Truck i :	36,36
Profit of item i:	120,93,70,85,125,104,98,130,140,65
Size of item i:	5,11,22,15,7,9,18,14,10,12

Πίνακας 1



```

IPython console
Console 1/A x

In [128]: runfile('C:/Users/user/Dropbox/teaching/UoM/ODE/MBADS/ΑνάλυσηΑποφάσεωνΚαιΒελτισ
MBADS_cwk_part1.py', wdir='C:/Users/user/Dropbox/teaching/UoM/ODE/MBADS/ΑνάλυσηΑποφάσεωνΚ
The maximum profit is: 7929.0
Orders Assigned in truck 0 :
 0 1 5 8
The capacity used in truck 0 is 35

Orders Assigned in truck 1 :
 3 4 7
The capacity used in truck 1 is 36
  
```

Εικόνα 1

Μέρος 2

Υποθέστε ότι για το μοντέλο της UPak που αναπτύξατε στο Μέρος 1 προστίθενται τώρα οι παρακάτω απαιτήσεις. Υπάρχουν ζεύγη φορτίων που επιβάλλεται να φορτωθούν στο ίδιο φορτηγό για λόγους οικονομικών κλίμακας (π.χ. απευθύνονται στο ίδιο πελάτη).

Υπάρχουν επίσης φορτία που για λόγους ασφαλείας επιβάλλεται να φορτωθούν σε διαφορετικά φορτηγά (π.χ. υλικά που δεν πρέπει να βρίσκονται στο ίδιο όχημα).

Τα σύνολα των παραπάνω ζευγών φορτίων δίνονται στα διανύσματα `Bundle[A]` και `Forbidden[B]`, όπου `A` και `B` είναι οι διαστάσεις των διανυσμάτων. Πιο συγκεκριμένα τα διανύσματα αυτά περιέχουν δείκτες φορτίων. Για παράδειγμα αν στο `Bundle[A]` περιλαμβάνεται το ζεύγος (1,2), αυτό σημαίνει ότι το δεύτερο και το τρίτο φορτίο θα πρέπει αν φορτωθούν, να τοποθετηθούν στο ίδιο φορτηγό (υπενθυμίζουμε ότι υποθέτουμε πως τα φορτία αριθμούνται σειριακά από το 0 ως το $N-1$). Αντίστοιχα αν στο διάνυσμα `Forbidden[B]` περιλαμβάνεται το ζεύγος «5,6» αυτό θα σημαίνει ότι η έκτη και η έβδομη παραγγελία θα πρέπει να φορτωθούν σε διαφορετικά φορτηγά.

1. Αναπτύξτε την αναλυτική μορφή του μοντέλου όπου θα περιγράψετε όλα τα νέα σε σχέση με το πρώτο μέρος σύνολα μεταβλητών, περιορισμών και παραμέτρων του μοντέλου.
(10%)
2. Να αναπτύξετε το μοντέλο βελτιστοποίησης σε Python/Pyomo. Η UPak για να σας διευκολύνει σας παρέχει τα δεδομένα του πίνακα 2. Να σημειωθεί ωστόσο ότι ιδανικά ο κώδικας σας θα πρέπει να είναι κλιμακούμενος (scalable), ικανός δηλαδή να αντιμετωπίσει προβλήματα διαφορετικών διαστάσεων.
Ο κώδικας θα πρέπει να επιστρέφει το βέλτιστο πλάνο φόρτωσης και το αντίστοιχο μέγιστο κέρδος.
(10%)

Number of Trucks :	2
--------------------	---

Number of Items:	10
Capacity of Truck i :	36,36
Profit of item i:	120,93,70,85,125,104,98,130,140,65
Size of item i:	5,11,22,15,7,9,18,14,10,12
Bundle:	(1,2), (6,7)
Forbidden:	(5,6)

Πίνακας 2

Β. Παραδοτέα:

1. Ένα αρχείο `.py` με τον κώδικα που αντιστοιχεί σε κάθε ένα από τα μέρη 1 έως 2 (δηλ συνολικά μέχρι και δύο αρχεία). Το όνομα του αρχείου θα πρέπει να είναι στη μορφή `BA_Opt_<επώνυμο>_part#.py` (π.χ. `BA_Opt_kaparis_part1.py`).
2. Ένα αρχείο `.dat` με τα δεδομένα εισόδου για κάθε ένα από τα μέρη 1 έως και 2 (δηλ συνολικά μέχρι και δύο αρχεία) για τα οποία αναπτύξατε ένα `abstract` μοντέλο. Το όνομα του αρχείου θα πρέπει να είναι στη μορφή `BA_Opt_<επώνυμο>_part#.dat` (π.χ. `BA_Opt_kaparis_part1.dat`).
3. Ένα μοναδικό αρχείο `.pdf` που θα περιέχει την περιγραφή του μοντέλου για κάθε ένα από τα 2 μέρη.
4. Όλα τα αρχεία θα κατατεθούν ως ένα μοναδικό `.ZIP` αρχείο με όνομα που θα πρέπει να είναι της μορφής `BA_Opt_CWK_<επώνυμο>.zip`

Γ. Λοιπές Πληροφορίες & Σχόλια:

- Σαν ελάχιστη απαίτηση στο Μέρος 1.2 και 2.2 θα πρέπει να αναπτύξετε το `concrete` μοντέλο για τα παραδείγματα αρχείων δεδομένων που σας δίνονται. **Ωστόσο για να διεκδικήσετε το άριστα θα πρέπει ο κώδικας σας να είναι κλιμακούμενος (scalable).** Σύμφωνα με την ύλη που καλύψαμε στην `Pyomo`, αυτό μπορείτε να το πετύχετε με την χρήση `abstract Pyomo` μοντέλων.
- Τα δύο παραδείγματα σας δίνουν κάποια δεδομένα για να επιβεβαιώσετε αριθμητικά αποτελέσματα σας για αυτές τις συγκεκριμένες περιπτώσεις. Ωστόσο, σας συμβουλεύω να **αναπτύξετε και δικά σας αρχεία ώστε να ελέγξετε την εγκυρότητα του κώδικά σας για διάφορες διαστάσεις δεδομένων.**
- Τα σωστά αριθμητικά αποτελέσματα **δεν συνεπάγονται απαραίτητα** ότι και ο κώδικας και/ή το μοντέλο σας είναι «σωστά».
- Μπορείτε να υποθέσετε ότι τα δεδομένα εισόδου είναι πάντα έγκυρα.
- Μπορείτε να υποθέσετε ότι οι δείκτες των φορτηγών και των αντικειμένων είναι ακέραιοι η αρίθμηση των οποίων ξεκινά από το 0 (μηδέν).
- Η παράδοση των εργασιών θα γίνει μέσω του `eclass`.
- Η καταληκτική ημέρα και ώρα κατάθεσης της εργασίας είναι **23:55, Πέμπτη 29/06/2023**. Να σημειωθεί ότι το σύστημα δε θα δέχεται καταχωρήσεις μετά από αυτήν την ημερομηνία και καταθέσεις εργασιών μέσω email δε θα γίνονται δεκτές.
- Σε περίπτωση που διαπιστωθεί συνεργασία στην ανάπτυξη του μοντέλου ή του κώδικα η συνέπεια θα είναι μηδενισμός της εργασίας όσων εμπλέκονται.

- Στον παρακάτω πίνακα (version control) δίνεται ένας πίνακας όπου καταγράφεται οποιαδήποτε πιθανή αλλαγή/διόρθωση γίνει στο αρχείο από τη στιγμή της ανάρτησης του μέχρι την καταληκτική ημερομηνία της εργασίας.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ!

Version Control

Version	Date	Author	Changes
1.0	31/05/2023	K.Kaparis	Release Version