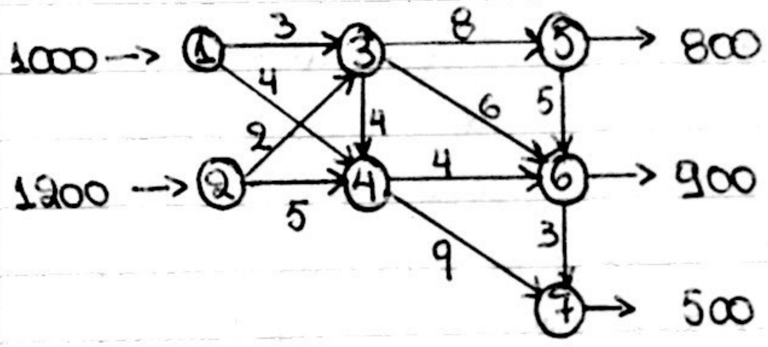
16/5/16

Το πρόβλημα της μεταφόρτωσης:



Μοντελοποίηση:

$\min \sum\sum c_{ij} x_{ij}$

$x_{13} + x_{14} \leq 1000$

$x_{23} + x_{24} \leq 1200$

$x_{13} + x_{23} = x_{34} + x_{36} + x_{35}$ ('Ο,τι βελάκι "μπαίνει" πρέπει και να "βγαίνει")

$x_{14} + x_{24} = x_{46} + x_{47}$

$x_{35} = 800 + x_{56}$

$x_{36} + x_{46} + x_{56} = x_{67} + 900$

$x_{47} + x_{67} = 500$

$x_{ij} \geq 0$.

C_{ij} κόστος, είναι οι αριθμοί στα βέλη.
Πώς μπορούμε να το λύσουμε τώρα;
Τους σταθμούς μεταφόρτωσης τους θεωρώ ως σταθμούς παραγωγής αλλά και ως σταθμούς προορισμού ταυτόχρονα.

	3	4	5	6	7	
1	3	4	M	M	M	1000
2	2	5	M	M	M	1200
3	0	7	8	6	M	2200
4	M	0	M	4	9	2200
5	M	M	0	5	M	2200
6	M	M	M	0	3	2200
	2200	2200	2200 +800	~~2200~~ 1900	500	

Το πρόβλημά μου είναι ισορροπημένο.
Όπου δεν υπάρχει τρόπος να πάω τότε M το κόστος με $M \gg 0$. Με αυτόν τον τρόπο έχουμε ένα πρόβλημα μεταφοράς και σύμφωνα με τους αλγόριθμους του τελευταίου μαθήματος μπορεί να λυθεί.

Πρόβλημα εκχώρησης
(Υποψήφιοι σε συγκεκριμένες θέσεις)

	B_1	B_2	B_3
A_1	30	52	76
A_2	42	36	80
A_3	55	48	60

Πώς πρέπει να βάλουμε τους υποψήφιους στην εκάστοτε θέση.

Ορίζουμε $x_{ij} = \begin{cases} 1, & \text{αν το άτομο } A_i \text{ πάρει την θέση } B_j \\ 0, & \text{αλλιώς} \end{cases}$

Θέλουμε να μεγιστοποιήσουμε την απόδοση.

$$\max\ 30x_{11} + 52x_{12} + \dots + 60x_{33}$$

$$x_{11} + x_{12} + x_{13} = 1$$
$$x_{21} + x_{22} + x_{23} = 1$$
$$x_{31} + x_{32} + x_{33} = 1$$
$$x_{11} + x_{21} + x_{31} = 1$$
$$x_{12} + x_{22} + x_{32} = 1$$
$$x_{13} + x_{23} + x_{33} = 1$$

* Περιορισμοί, κάθε άτομο να πάρει μια θέση και αντιστρόφως.

Προκειμένου να λύσουμε τώρα το πρόβλημα
Ουγγρικός Αλγόριθμος.

1) α] Αφαιρούμε το μεγαλύτερο c_{ij} κάθε γραμμής από κάθε c_{ij} της γραμμής.

β] Στο νέο tableau αφαιρούμε το μεγαλύτερο c_{ij} κάθε στήλης από κάθε c_{ij} της στήλης.

2) Με τον ελάχιστο αριθμό ευθειών r διαγράφουμε τα μηδενικά του τελευταίου tableau.
Αν $r=m$ τότε το tableau δίνει άριστη λύση.
Αν $r<m$ τότε πάμε στο επόμενο βήμα.

3) Αφαιρούμε το μεγαλύτερο μη διαγραμμένο c_{ij} από κάθε μη διαγραμμένο c_{ij} και το προσθέτουμε στα c_{ij} που είναι στη διασταύρωση 2 ευθειών.

4) Επιστρέφουμε στο 2).

30	52	76
42	36	80
55	48	60

→

-46	-24	0
-38	-44	0
-5	-12	0

→

-41	-12	0
-33	-32	0
0	0	0

(ο τρόπος με τον οποίο κάλυψα τα μηδενικά.)

2<3, επομένως δεν έχει τελειώσει

	-29	0*	0
→	-21	-20	0*
	0*	0	-12

- Για το *, όταν βρίσκαμε γραμμή ή στήλη με ένα μηδενικό, το "σημαδεύαμε" και ύστερα ξανακάναμε τη διαδικασία χωρίς όμως να συμπεριλάβουμε τη γραμμή ή στήλη που "σημαδέψαμε".
- Εδώ είχαμε είχαμε πρόβλημα μεγιστοποίησης, εάν είχαμε ελαχιστοποίησης τότε το μετατρέπαμε σε μεγιστοποίησης.
- Μέγιστη τιμή της αντικειμενικής συνάρτησης 52+55+80

	B_1	B_2	B_3	B_4
A_1	12	40	60	35
A_2	48	30	75	60
A_3	55	62	18	40
A_4	0	0	0	0

*(Έχουμε θέσεις > υποψηφίους, έτσι προσθέτουμε έναν υποψήφιο

	-48	-20	0	-25
→	-27	-45	0	-15
	7	0	-44	22
	0	0	0	0

→

-33	-5	0*	-10
-12	-30	0	0*
-7	0*	-59	-22
0*	0	-15	0

Επομένως η 1η θέση θα μείνει ακάλυτη.

	B_1	B_2	B_3	B_4	B_5
Ελεύθερο	61	58	53	54	62
Ύπτιο	63	65	61	57	58
Πρόσθιο	57	59	56	61	62
Πεταλούδα	58	60	56	55	57
	0	0	0	0	0

* Όπως και πριν επειδή έχουμε διαφορετικούς αθλητές > αθλήματα προσθέσαμε ένα άθλημα.
* Εμείς θέλαμε το μικρότερο δυνατό χρόνο, αφού μιλάμε για αθλητές, οπότε έχουμε πρόβλημα ελαχιστοποίησης οπότε το μετατρέπουμε σε πρόβλημα μεγιστοποίησης βάζοντας "–".

-61	-58	-53	-54	-62
-63	-65	-61	-57	-58
-57	-59	-56	-61	-62
-58	-60	-56	-55	-57
0	0	0	0	0

→

-8	-5	0	-1	-9
-6	-8	-4	0	-1
-1	-3	0	-5	-6
-3	-5	-1	0	-2
0	0	0	0	0

3<5, επόμενο βήμα

→

-7	-4	0*	-1	-8
-5	-7	-4	0	0
0*	-2	0	-5	-5
-2	-4	-1	0*	-1
0	0*	-1	-1	0

Άρα, ο αθλητής 2 δεν θα χρησιμοποιηθεί, ο 3 ελεύθερο, ο 5 ύπτιο, ο 1 πρόσθιο, ο 4 πεταλούδα.

...πλεια) · 3 άτομα - 4 εργασίες.
Κάθε άτομο μπορεί να χρησιμοποιηθεί 2 φορές, να ελαχιστοποιηθεί ο χρόνος.

	B_1	B_2	B_3	B_4
A_1	35	2	43	3
A_2	5	28	4	36
A_3	4	3	5	4

ΟΧΙ ΓΙΑ ΤΙΣ ΕΞΕΤΑΣΕΙΣ

1) Πρόβλημα συντομότερης διαδρομής

2) Πρόβλημα ελαχίστου ζευγνύοντος δέντρου

3) Πρόβλημα μέγιστης ροής

1) Θέλω να ελαχιστοποιήσω τις αποστάσεις μια

$$\min \sum_{(i,j)\in E} d_{ij} x_{ij}$$

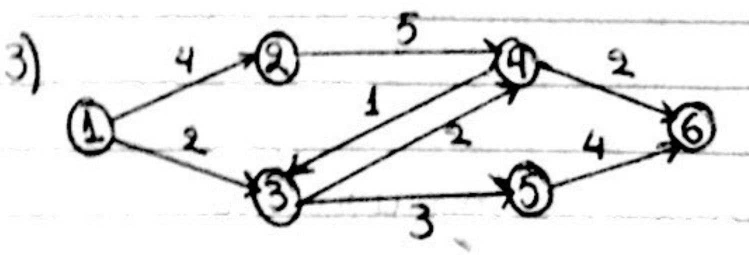
$$\underbrace{\sum_{i:(i,k)\in E} x_{ik}}_{\text{μπαίνουν}} - \underbrace{\sum x_{kj}}_{\text{βγαίνουν}} = \begin{cases} -1 & k=s \\ 1 & k=t \\ 0 & k\in E-\{s,t\} \end{cases}$$

$$x_{ij} = \begin{cases} 1 \\ 0 \end{cases}$$

2) $\sum_{(i,j)\in E} x_{ij} = n-1$ εάν έχω n-σημεία

$\sum_{(i,j)\in E,\ i\in S,\ j\in S} x_{ij} \le |S|-1$ ∀S για οποιοδήποτε υποσύνολο να μην χρησιμοποιώ κύκλους

3)



Να μεγιστοποιήσω την ποσότητα που πάει στο 6.

x_{ij} η ποσότητα από τον i στο j κόμβο.

$\max x_{46}+x_{56}$

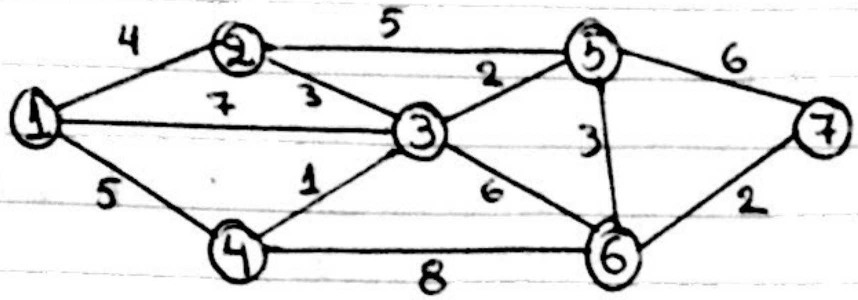
$x_{12}\le 4,\ x_{13}\le 2, \dots$

$x_{12} = x_{24},\ x_{13} = x_{34}+x_{35}$

⋮

$x_{i,j} \ge 0$

1)

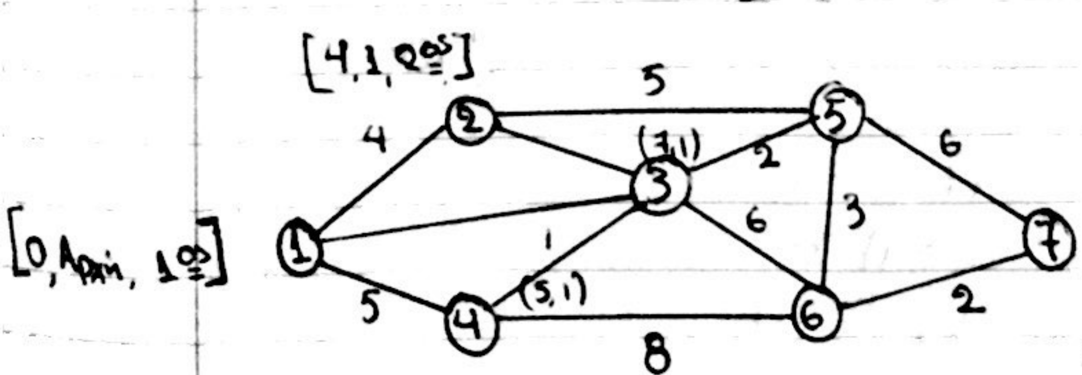


1) Καταγραφή του συνόλου Λ των μόνιμων κόμβων
2) Εντοπισμός όλων των κόμβων που είναι άμεσα συνδεδεμένοι με τουλάχιστον 1 από τους κόμβους του Λ (π.χ Λ = 1 τότε οι άμεσα συνδεδεμένοι 2,3,4)

* Υπολογισμός του μήκους της (προσωρινής) διαδρομής από την αφετηρία σε κάθε ένα απ' αυτούς.
* Έλεγχος για βελτίωση (υπαρχουσών προσωρινών) διαδρομών.
* Επιλογή του κόμβου που αντιστοιχεί στη συντομότερη διαδρομή για είσοδο στο Λ. Η διαδρομή προς το συγκεκριμένο κόμβο δεν επιδέχεται περαιτέρω βελτίωση.

3) Επαναλαμβάνουμε τα 1 ως 2 μέχρι όλοι οι κόμβοι να γίνουν μόνιμοι ή εισέλθει στο Λ ο κόμβος προορισμού.

Λυμένοι κόμβοι	Ακμή, άμεσα συνδ. κόμβοι	Προσωρινό μήκος διαδρομής	Σχόλια
{1}	1-2	4	Λυμένος κόμβος
	1-3	7	
	1-4	8	

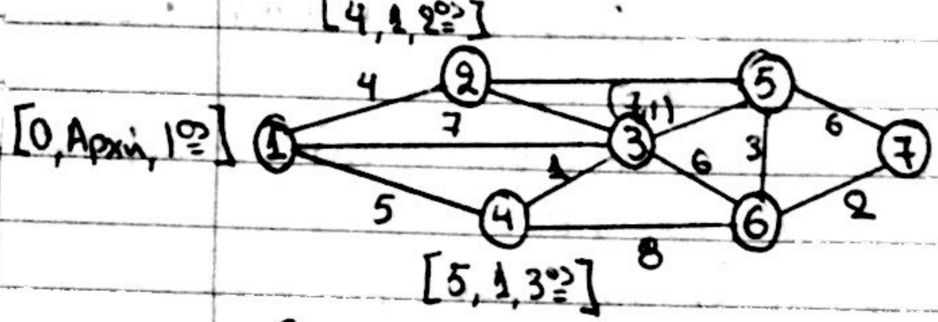


Scanned by CamScanner

2η επανάληψη

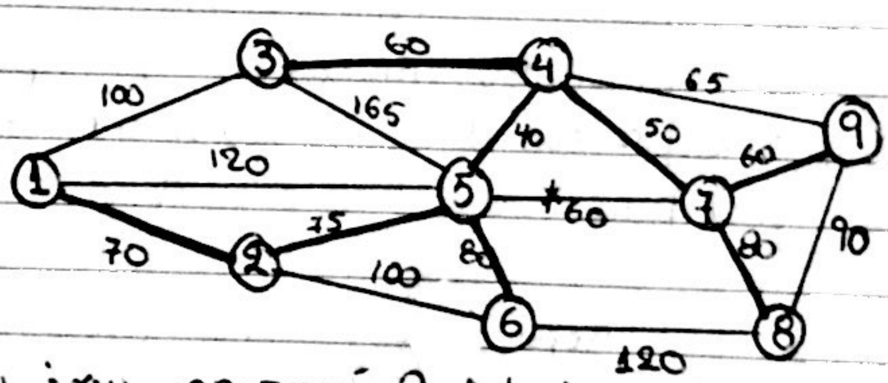
{1,2}	1-3	7	∄ βελτίωση
	1-4	5	
	2-3	4+3=7	
	2-5	4+5=9	

Επομένως ο κόμβος που θα γίνει μόνιμος είναι ο 4



κ.λ.π.

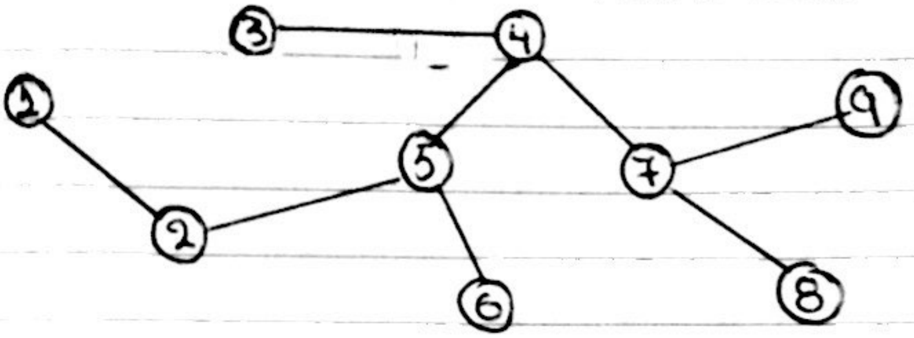
2)



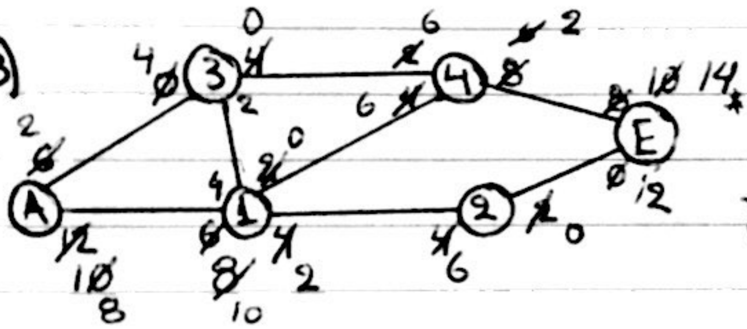
Δεν έχω προσανατολισμό μπορώ να πάω όπου θέλω. Θέλουμε να συνδέσουμε όλους τους κόμβους. Διατάσσουμε ως προς αύξουσα σειρά

(5,4) 40	(1,2) 70	(1,5) 120
(4,7) 50	(2,5) 75	(6,8) 120
(5,7) 60	(5,6) 80	
(3,4) 60	(7,8) 80	
(7,9) 60	(8,9) 90	
(3,5) 65	(1,3) 100	
(4,9) 65	(2,6) 100	

Παίρναμε το μικρότερο, το (5,7) δεν το επιλέξαμε γιατί βρόγχο, θέλω όλοι συνδεδεμένοι και από το μικρότερο προσθέτω κόμβους χωρίς να δημιουργείται κύκλος - βρόγχος.



3)



(δύο νούμερα στις γραμμές, γιατί και στέλνω και παίρνω)

διαλέξαμε διαδρομή αυθαίρετα.

• A-1-2-E

ότι και να στείλω από το A στο E θα πάνω 2, προσθέτουμε και αφαιρούμε, τέλειωσε γιατί δεν μπορεί να στείλει τίποτε άλλο.

• A-1-4-E

Το καλύτερο που θα στείλω 2 γιατί μόνο αυτό περνάει απ' το αρχικό.

• A-3-4-E το καλύτερο 4.

Scanned by CamScanner

Διαδρομή	A	A-3	1-2	1-4	3-4	2-E	4-E
A-1-2-E	2		2			2	
A-1-4-E	2			2			2
A-3-4-E	0	4			4		4
Σύνολο	4	4	2	2	4	2	6