Π Ο Λ Υ Τ Ε Χ Ν Ε Ι Ο   Κ Ρ Η Τ Η Σ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

ΠΡΟΣΕΓΓΙΣΗ ΤΗΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΣΥΝΑΛΛΑΓΜΑΤΟΣ ΜΕ ΤΗ ΘΕΩΡΙΑ ΤΩΝ ΡΟΩΝ

Βασίλειος Αρβανίτης

ΧΑΝΙΑ, 1991

ΠΕΡΙΕΧΟΜΕΝΑ

ΣΕΛ.

ΣΕΛ.

Ευχαριστίες

Με την ολοκλήρωση της μελέτης θα ήθελα να ευχαριστήσω πρώτα τον καθηγητή μου κ. Μ. Μιχαλόπουλο για την αμέριστη βοηθειά του, τους συναδέλφους Αρη Χαλβατζή, Οδυσσέα Στεφάνου και τον φίλο Γιάννη Λυμπερόπουλο για την τεχνική υποστήριξη της μελέτης.

ΕΙΣΑΓΩΓΗ

Το αντικείμενο της μελέτης που ακολουθεί, είναι η δημιουργία ενός προγράμματος που θα επιλύει προβλήματα μεγιστοποίησης του κέρδους στην Αγορά Συναλλάγματος ή αλλιώς προβλήματα arbitrage. Ενας απο τους στόχος της εργασίας είναι και η ελαχιστοποίηση του χρόνου επίλυσης του προβλήματος. Για το σκοπό αυτό, χρησιμοποιήθηκε η θεωρία των Γραφημάτων και ειδικότερα οι Ροές με τόξα κέρδους. Θα μπορούσε βέβαια, η επίλυση του προβλήματος arbitrage να γίνει και με Γραμμικό Προγραμματισμό έχοντας κέρδος, στην εκπόνηση του κώδικα προγραμματισμού και με αντίτιμο όμως τον μεγαλύτερο χρόνο επίλυσής του. Αντικειμενικός στόχος λοιπόν, της εργασίας είναι πρώτον **η μεγιστοποίηση του κέρδους** και δεύτερο **η ελαχιστοποίηση του χρόνου εκτέλεσης**.

Στο πρώτο κεφάλαιο, λοιπόν, θα γίνει μια γενική παρουσίαση της θεωρίας Συναλλάγματος, και ειδικότερα στις δύο Αγορές Συναλλάγματος, Τρέχουσα Αγορά (ή Αγορά Οψης) και Προθεσμιακή Αγορά. Δίνονται οι απαραίτητοι βασικοί ορισμοί, όπως η έννοια του arbitrage, που είναι και η βασικότερη έννοια της εργασίας.

Στη συνέχεια , στο δεύτερο κεφάλαιο, γίνεται επίσης μια γενική παρουσίαση, της θεωρίας των Ροών. Παρουσιάζεται και ο βασικός αλγόριθμος του Νίκου Χριστοφίδη, που αναφέρεται σε δίκτυα ροών με τόξα κέρδους, καθώς και δύο άλλοι βοηθητικοί αλγόριθμοι, ο Γενικος και ο Αλγόριθμος του Dijkstra, που χρησιμοποιούνται από τον αλγόριθμο του Χριστοφίδη.

Στο τρίτο κεφάλαιο, παρουσιάζονται αρχικά, τα προβλήματα arbitrage που μπορούν να επιλυθούν με την χρήση αυτού του αλγορίθμου. Στη συνέχεια, γίνεται η μοντελοποίηση του βασικού προβλήματος arbitrage, και τέλος παρουσιάζεται ένα παράδειγμα πιλότος.

Η παρουσίαση του κώδικα και η εκτενής αναλυσή του γίνεται στο τέταρτο κεφάλαιο.

Τέλος η μελέτη θα κλείσει με τη διατύπωση ορισμένων συμπερασμάτων σχετικά με ό,τι έχει προηγηθεί.

Είναι, επίσης, απαραίτητο να τονιστεί σ'αυτό το σημείο ότι σε όλη την έκταση της μελέτης χρησιμοποιούνται όροι όπως "arbitrage" ή "cross exchange rate" χωρίς να μεταφραστούν. Αυτό γίνεται επειδή οι συγκεκριμένοι όροι έχουν επικρατήση διεθνώς.

ΚΕΦΑΛΑΙΟ 1

ΒΑΣΙΚΕΣ ΑΡΧΕΣ ΤΗΣ ΑΓΟΡΑΣ ΣΥΝΑΛΛΑΓΜΑΤΟΣ

1.1 Εισαγωγή

Η διεθνής ανταλλαγή αγαθών και υπηρεσιών απαιτεί, όπως ακριβώς και στην περίπτωση του εξωτερικού εμπορίου μιάς χώρας, τη χρησιμοποίηση χρηματικών μέσων. Οι διεθνείς όμως πληρωμές απαιτούν την χρήση "διεθνούς" χρήματος ή χρήματος που είναι αποδεκτό στις συναλλαγές των εμπορευόμενων χωρών. Ετσι για παράδειγμα ο εξαγωγέας ελληνικού καπνού στην Κορέα [2] θα απαιτεί να πληρωθεί σε δραχμές και όχι σε γουόνς που διαθέτει ο Κορεάτης εισαγωγέας. Πώς πραγματοποιούνται οι εμπορευματικές πληρωμές αυτού του είδους; Και γενικά πώς χρηματοδοτούνται οι διεθνείς ανταλλαγές αγαθών και υπηρεσιών;

Προκύπτει επομένως η ανάγκη μιας διεθνούς αγοράς χρήματος που διαμέσου μιας διαδικασίας κανονισμών, να επιτρέπει τη διευθέτηση των χρηματικών απαιτήσεων και υποχρεώσεων που απορρέουν απο τη διεθνή διακίνηση εμπορευμάτων ή τη μεταφορά κεφαλαίων για επενδυτικούς ή άλλους σκοπούς. Η αγορά αυτή είναι η **Αγορά Συναλλάγματος** [2]. Το διαπραγματευόμενο αγαθό στην αγορά αυτή είναι το συνάλλαγμα (foreign exchange). Ο όρος αναφέρεται γενικά σε ξένες χρηματικές κυκλοφορίες και σε διάφορα πιστωτικά μέσα σε ξένα νομίσματα.

Η Αγορά Συναλλάγματος, επομένως θα πρέπει να νοείται σαν τόπος συνάντησης αγοραστών και πωλητών ξένων νομισμάτων. Το χαρακτηριστικό που διακρίνει την αγορά αυτή είναι η πλήρη ανταγωνιστικότητα που τη διέπει, όσον αφορά την ομοιογένεια του προιόντος (χρήματος). Το γεγονός αυτό, φαινομενικά τουλάχιστον, έρχεται σε αντίθεση με το δεύτερο χαρακτηριστικό της Αγοράς Συναλλάγματος, δηλαδή ότι ο τόπος ανταλλαγής δεν είναι αυστηρά προκαθορισμένος [2]. Η αγοραπωλησία συναλλάγματος μπορεί να διεξαχθεί σε διάφορες πόλεις σε όλο τον κόσμο.

Οπως και στην περίπτωση του εσωτερικού εμπορίου, οι διεθνείς πληρωμές και η διακίνηση κεφαλαίων γίνεται με την μεσολάβηση των κεντρικών και εμπορικών τραπεζών των διαφορών χωρών. Οι τράπεζες, στη συγκεκριμένη περίπτωση, αποτελούν το εκτελεστικό όργανο των εντολών arbitrage και άλλων συναλλαγών, που συνεπάγονται την μεταφορά κεφαλαίων από χώρα σε χώρα. Η εύρυθμη λειτουργία της Αγοράς Συναλλάγματος, επομένως, εξαρτάται από την καλή οργάνωση και συνεργασία των τραπεζικών συστημάτων των διαφόρων χωρών [2].

Ο μεσολαβητικός ρόλος των τραπεζών, διευκολύνει την πραγματοποίηση των δύο βασικών λειτουργιών της Αγοράς Συναλλάγματος: α) τη μεταβίβαση αγοραστικής δύναμης από χώρα σε χώρα σε ξένο συνάλλαγμα και β) την εξασφάλιση πιστώσεων για τη διακίνηση των αγαθών στη διεθνή αγορά [2]. Μία τρίτη λειτουργία της Αγοράς Συναλλάγματος είναι η διευκόλυνση των ατόμων που πραγματοποιούν συναλλαγές στα πλαίσιά της, ώστε να ελαχιστοποιήσουν τον κίνδυνο χρηματικών ζημιών λόγω της διακύμανσης στην τιμή συναλλάγματος [2]. Παρακάτω, εξετάζεται συνοπτικά ο μηχανισμός των Διεθνών Πληρωμών καθώς και πώς πραγματοποιούνται οι λειτουργίες αυτές. Αρχικά όμως, παρατίθονται ορισμένες βασικές έννοιες σχετικά με τη Ζήτηση και την Προσφορά του συναλλάγματος.

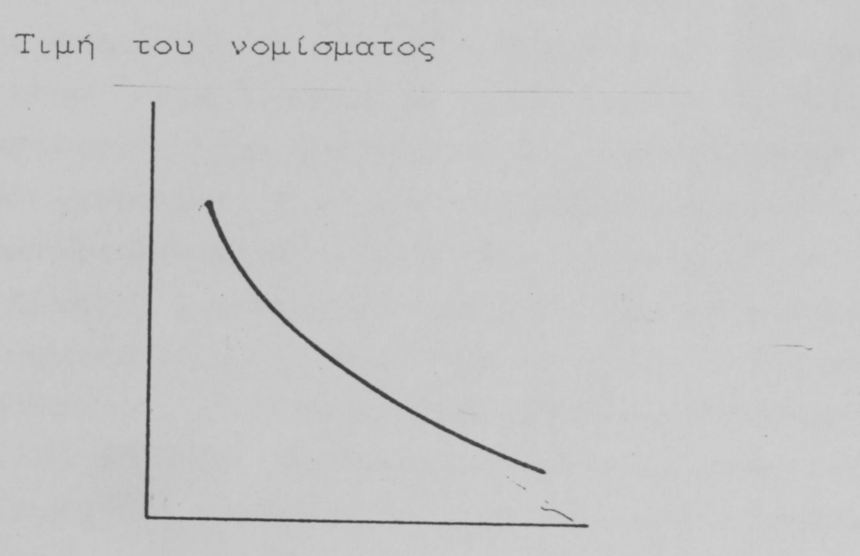
1.2 Ζήτηση και Προσφορά Συναλλάγματος

Οπως συμβαίνει και με κάθε άλλο αγαθό, η τιμή συναλλάγματος μπορεί να καθοριστεί απο τη προσφορά και τη ζητησή του. Αυτός όμως είναι ένας μόνο τρόπος για τον προσδιορισμό της τιμής του συναλλάγματος. Η παρεμβατική πολιτική του κράτους (κεντρικής τράπεζας) είναι συνήθως βασικός παράγοντας στην προκειμένη περίπτωση.

1.2.1 Ζήτηση Συναλλάγματος

Οι πηγές ζήτησης συναλλάγματος είναι ποικίλες, γενικά όμως, κατατάσονται σε δύο βασικές κατηγορίες. Η πρώτη κατηγορία είναι η διεθνής ανταλλαγή αγαθών και υπηρεσιών πού συνεπάγεται εμπορευματικές πληρωμές σε ξένα νομίσματα. Η δεύτερη κατηγορία περιλαμβάνει τις πηγές ζήτησης συναλλάγματος για μή εμπορευματικές πληρωμές, όπως η χρησιμοποίηση συναλλάγματος για την απόκτηση περιουσιακών τίτλων στο εξωτερικό, για κερδοσκοπικούς σκοπούς, για την εξόφληση δανείων κτλ. [1].

Η σχέση μεταξύ τιμής και ζητούμενης ποσότητας συναλλάγματος, όπως και στην περίπτωση αγοράς οποιουδήποτε άλλου προιόντος, εκφράζεται απο μιά καμπύλη ζήτησης που έχει "ορθόδοξη" μορφή και σχήμα [1] :



1.2.2 Προσφορά Συναλλάγματος

Οπως ελέχθει παραπάνω, η χρηματική κυκλοφορία μιάς χώρας αποτελεί συνάλλαγμα για τις άλλες χώρες. Αυτό σημαίνει ότι η ζήτηση απο τη χώρα Α του νομίσματος της χώρας Β αποτελεί στην πραγματικότητα προσφορά συναλλάγματος για τη χώρα Β. Παραπέρα, αφού η ζήτηση του νομίσματος μιας χώρας είναι παράγωγη της ζήτησης για εισαγωγές απο τη χώρα αυτή, φαίνεται λοιπόν, ότι η προσφορά συναλλάγματος στη σχετική χώρα είναι συνάρτηση των εξαγωγών της. Ετσι, για παράδειγμα, όταν η Γερμανία ζητά στερλίνες για την αγορά αγγλικών προιόντων, τα αγοράζει με αντάλλαγμα μάρκα που αποτελούν την προσφορά συναλλάγματος για την Αγγλία [2]. Η ανταλλαγή των νομισμάτων διεξάγεται στην Αγορά Συναλλάγματος και στις περισσότερες περιπτώσεις αποβλέπει στην πραγματοποίηση εμπορευματικών πληρωμών.

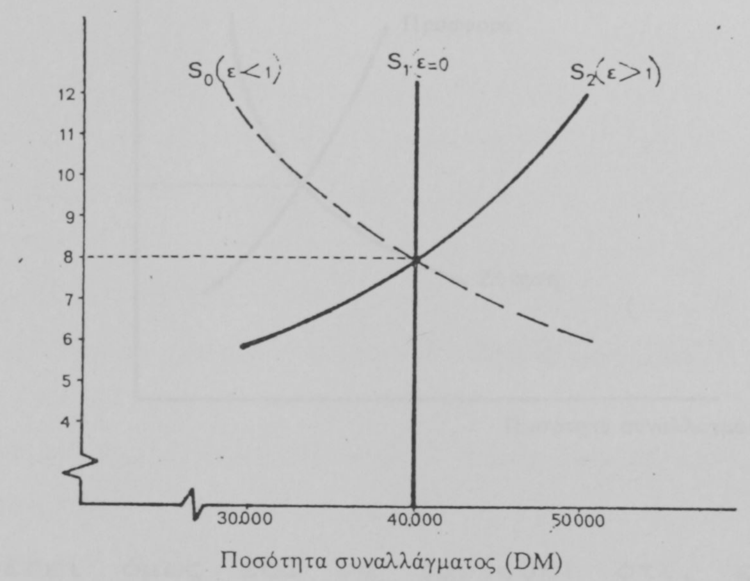
Ιδιαίτερη επομένως σημασία για τον καθορισμό της προσφοράς συναλλάγματος μιας χώρας είναι η ζήτηση απο το εξωτερικό για τα προιόντα της χώρας αυτής. Η ελαστικότητα της ζήτησης για τα εξαγωγικά προιόντα μιας χώρας θα καθορίσει την κλίση και την ακριβή θέση της καμπύλης προσφοράς. Οσο πιο ελαστική είναι η ζήτηση της Γερμανίας για αγγλικά προιόντα, τόσο μεγαλύτερη θα είναι η ποσότητα γερμανικών μάρκων (DM) που προμηθεύεται η Αγγλία απο τη διάθεση των προιόντων της σε χαμηλότερες τιμές. Η μείωση της τιμής της στερλίνας σε μάρκα θα οδηγήσει στη διάθεση μεγαλύτερου χρηματικού ποσού μάρκων για την αγορά αγγλικών προιόντων, αν η ελαστικότητα της ζήτησης είναι μεγαλύτερη απο τη μονάδα. Αυτό υπονοεί αυξημένη προσφορά συναλλάγματος για την Αγγλία, δηλαδή η καμπύλη προσφοράς μάρκων έχει θετική κλίση. Αν η ελαστικότητα της ζήτησης για αγγλικά προιόντα ήταν μικρότερη απο τη μονάδα, η ποσότητα μάρκων, μετά τη μείωση της τιμής της στερλίνας, θα μειωνόταν και η καμπύλη προσφοράς συναλλάγματος θα είχε αρνητική κλίση. Τέλος στην περίπτωση μοναδιαίας ελαστικότητας της ζήτησης, η ποσότητα του συναλλάγματος θα παραμένει σταθερή παρά τις αυξομειώσεις της τιμής του [2].

Για να κατανοηθεί καλύτερα η σχέση ανάμεσα στην τιμή και την προσφορά συναλλάγματος δίνεται ένα αριθμητικό παράδειγμα όπου συναλλασσόμενες χώρες είναι Ελλάδα και η Γερμανία [2] :

Ἡ Καμπύλη Προσφορᾶς Συναλλάγματος

Τιμὴ Ἐξαγωγῆς Καπνοῦ σὲ δρχ. (1)	Τιμὴ Γερμανικοῦ Μάρκου σὲ Δρχ. (2)	Τιμὴ Καπνοῦ σὲ Μάρκα Γερμανίας (DM) (3)	Ζητούμενη Ποσότητα (κιλά) Καπνοῦ (ἐλαστικότητα = 0) (4)	Ποσότητα Συναλλάγματος (DM) (3) × (4) (5)	Ζητούμενη Ποσότητα (κιλά) Καπνοῦ (ἐλαστικότητα > 1) (6)	Ποσότητα Συναλλάγματος (3) × (6) (7)
40	12	3,33	12.000	40.000	15.000	49.950
40	11	3,67	11.000	40.000	13.000	47.710
40	10	4,00	10.000	40.000	11.000	44.000
40	9	4,44	9.000	40.000	9.500	42.180
40	8	5,00	8.000	40.000	8.000	40.000
40	7	5,71	7.000	40.000	6.800	38.828
40	6	6,66	6.600	40.000	5.400	35.964
40	5	8,00	5.000	40.000	4.000	32.000
40	4	10,00	4.000	40.000	2.800	28.000

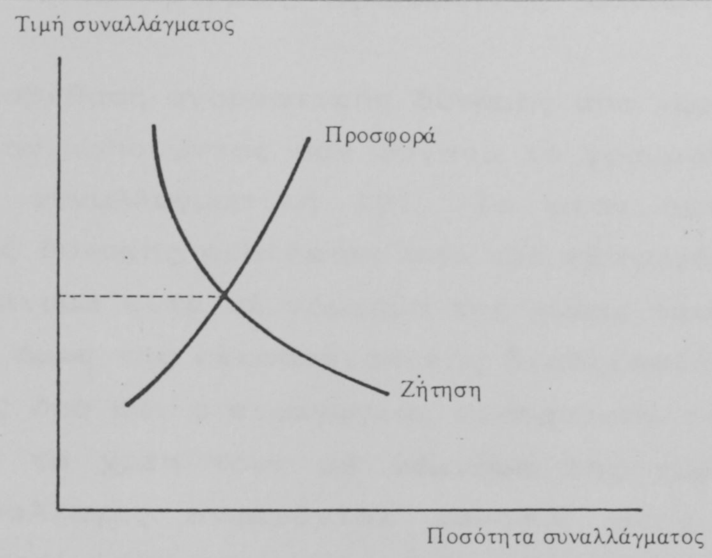
Η Ελλάδα, λοιπόν, είναι η χώρα που εξάγει και το εξαγόμενο προιόν είναι ο καπνός, ενώ η Γερμανία είναι η χώρα που ζητά το προιόν με αντάλλαγμα μάρκα, που αποτελούν συνάλλαγμα για την Ελλάδα. Η τιμή του προιόντος κατά μονάδα δίνεται σε δραχμές όπως φαίνεται στην στήλη (1) του παραπάνω πίνακα. Η στήλη (2) δίνει διάφορες ισοτιμίες δραχμής-μάρκου ενώ στην τρίτη στήλη η τιμή του προιόντος εκφράζεται σε μάρκα σύμφωνα με τις ισοτιμίες του. Στις στήλες (4) και (6) δίνονται οι ζητούμενες ποσότητες για ελαστικότητα ζήτησης του προιόντος μοναδιαία ή μεγαλύτερη απο τη μονάδα, αντίστοιχα. Η προσφορά συναλλάγματος για καθεμιά απο τις δυό περιπτώσεις ελαστικότητας δίνεται στις στήλες (5) και (7). Στην πρώτη περίπτωση η ποσότητα συναλλάγματος είναι σχεδόν σταθερή παρά τις αυξομειώσεις που σημειώνονται στην τιμή του μάρκου. Στο παρακάτω σχήμα η ποσότητα συναλλάγματος αντιστοιχεί στην καμπύλη προσφοράς S_1 που είναι τέλεια ανελαστική. Με ελλαστική ζήτηση εξάλλου η στήλη (7) δειχνει ότι η μείωση τής τιμής της δραχμής έναντι του μάρκου σημαίνει αύξηση της ποσότητας του προσφερόμενου συναλλάγματος. Στο σχήμα η περίπτωση αυτή απεικονίζεται απο την καμπύλη S_2. Τέλος η καμπύλη S_0 αντιστοιχεί σε ανελαστική ζήτηση του προιόντος και έχει αρνητική κλίση, σύμφωνα με τα παραπάνω [2] :



1.2.3 Συνολική Προσφορά, Ζήτηση και Τιμή Συναλλάγματος

Αφού κατασκευάσθηκαν οι καμπύλες ζήτησης και προσφοράς συναλλάγματος, που προκύπτουν από την ανταλλαγή προιόντων μεταξύ δύο χωρών, είναι δυνατό να παρθούν και οι συνολικές καμπύλες προσφοράς και ζήτησης του συναλλάγματος. Από την αντιπαράθεση των δύο αυτών καμπυλών λαμβάνεται η τιμή συναλλάγματος, όπως φαίνεται στο σχήμα που ακολουθεί.

Η τομή των δύο καμπυλών καθορίζει την τιμή του συναλλάγματος [2] :



Πρέπει όμως εδώ να τονισθεί ότι, ενώ ο μηχανισμός καθορισμού της τιμής συναλλάγματος είναι βασικά αυτός που απεικονίζεται παραπάνω, για να συζητηθεί πιο διεξοδικά το θέμα θα πρέπει να ληφθεί υπόψη το υφιστάμενο διεθνές νομισματικό σύστημα, αφού αυτό καθορίζει σε μεγάλο βαθμό την έκταση των μεταβολών της τιμής του συναλλάγματος.

1.3 Ο Μηχανισμός Διενέργειας Των Διεθνών Πληρωμών

Οπως αναφέρθηκε παραπάνω, η μεταφορά αγοραστικής δύναμης από χώρα σε χώρα, με τη μορφή ξένου συναλλάγματος, πραγματοποιείται με τη μεσολάβηση των τραπεζών που κάνουν συναλλαγές για τους πελάτες τους ή για δικό τους λογαριασμό. Οι καλά οργανωμένες τράπεζες διαθέτουν τμήμα ξένου συναλλάγματος που παρέχει υπηρεσίες εξειδικευμένου προσωπικού ή επαγγελματιών χρηματιστών της Αγοράς Συναλλάγματος ή όπως είναι γνωστοί σαν διαπραγματευτές συναλλάγματος (forex dealers) [2]. Για να κατανοηθούν καλύτερα οι λειτουργίες της Αγοράς Συναλλάγματος θα αναφερθεί περιληπτικά ο μηχανισμός πραγματοποιήσης των λειτουργιών αυτών.

1.3.1. Η Μεταβίβαση Αγοραστικής Δύναμης

Η μεταβίβαση αγοραστικής δύναμης απο χώρα σε χώρα γίνεται συνήθως χρησιμοποιώντας σαν όργανο το γραμμάτιο συναλλάγματος (drafts) ή συναλλαγματική [3]. Το μέσο αυτό μεταφοράς της αγοραστικής δύναμης εκδίδεται απο τον εξαγωγέα (πιστωτή) είτε σε εγχώριο νόμισμα είτε σε νόμισμα της χώρας του εισαγωγέα. Με τη μεσολάβηση όμως της εκκαθαριστικής διαδικασίας των τραπεζών, τόσο ο εξαγωγέας όσο και ο εισαγωγέας εισπράτουν τις απαιτήσεις τους ή εξοφλούν τα χρέη τους με νόμισμα της χώρας τους. Τέτοιου είδους συναλλαγές ανφέρονται συνήθως στην **Τρέχουσα Αγορά Συναλλάγματος** (spot foreign exchange market). Δηλαδή οι αγοραπωλησίες γίνονται με τις τρέχουσες ισοτιμίες των νομισμάτων οι οποίες καθορίζονται καθημερινά απο την κεντρική τράπεζα κάθε χώρας, αυτό βέβαια σημαίνει ότι η ισοτιμία, π.χ., δύο νομισμάτων είναι η ίδια τελικά σε όλες τις Αγορές Συναλλάγματος, πράγμα που επιτυγχάνεται με το arbitrage ή πρόκριση συναλλάγματος [2] που είναι και το βασικό αντικείμενο της μελέτης.

1.3.2 Arbitrage [2]

Ο όρος arbitrage έχει καθιερωθεί διεθνώς και χρησιμοποιήται τόσο πλατιά απο "ειδικούς" και μη έτσι ώστε να μην χρειάζεται μετάφραση, θα μπορούσε πάντως να μεταφραστή σαν πρόκριση συναλλάγματος. Θα πρέπει να τονιστεί απο την αρχή ότι το arbitrage δεν είναι κερδοσκοπία στην Αγορά Συναλλάγματος, δεδομένου ότι αυτός που κάνει arbitrage δεν διατρέχει κανένα κίνδυνο χρηματικής ζημίας εξαιτίας κάποιας μεταβολής στην ισοτιμία.

Εκείνο που εξασφαλίζει ότι οι ισοτιμίες θα είναι ενιαίες και ότι ανάμεσα στις τιμές των διαφόρων νομισμάτων θα υπάρχει πλήρης αντιστοιχεία είναι η διενέργεια arbitrage. Το arbitrage είναι η ταυτόχρονη αγορά και πώληση διαφόρων χρηματικών κυκλοφοριών στην περίπτωση που υπάρχουν διαφορές στις ισοτιμίες των νομισμάτων αυτών. Αυτός που ασκεί το arbitrage αποβλέπει στην πραγματοποίηση κέρδους, που τα περιθωριά του καθορίζονται απο την έκταση της διαφοράς των ισοτιμιών. Τα περιθώρια αυτά μικραίνουν και τελικά εξαφανίζονται, μετά απο αυξομειώσεις στην ζήτηση και την προσφορά των νομισμάτων που αποτέλεσαν το αντικείμενο ανταλλαγής στο arbitrage. Θα πρέπει να σημειωθεί εδώ, ότι στον πιό πάνω ορισμό του arbitrage, αυτός που το ασκεί αγοράζει και πουλά **ταυτόχρονα**, για να καλυφθεί απο πιθανές διακυμάνσεις των ισοτιμιών. Δεν παίρνει επομένως "position" ή ανοιχτή θέση όπως συμβαίνει στην περίπτωση της κερδοσκοπίας. Σε αυτό το σημείο φαίνεται η μεγάλη σημασία που έχει ο χρόνος για αυτόν που παίρνει αποφάσεις για την αγορά και την πώληση των νομισμάτων. Για την καλύτερη κατανόηση της διενέργειας arbitrage δίνεται το παρακάτω παράδειγμα.

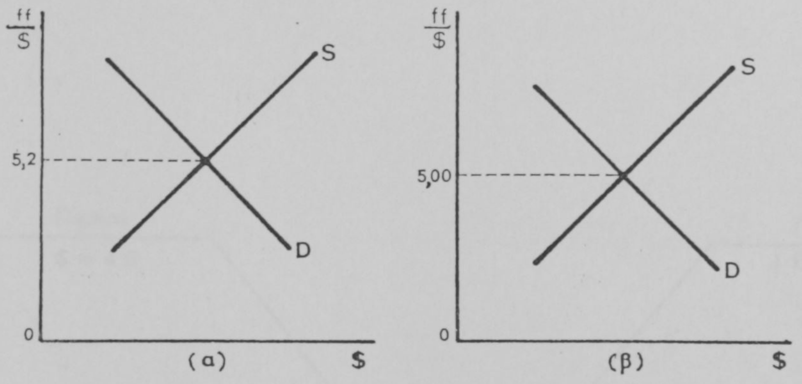
Υποτίθεται λοιπόν, η σχέση ανάμεσα στο δολάριο και στο γαλλικό φράγκο στις αγορές του Παρισίου και του Λονδίνου είναι

Παρίσι : 1$ = 5.2 Fr.

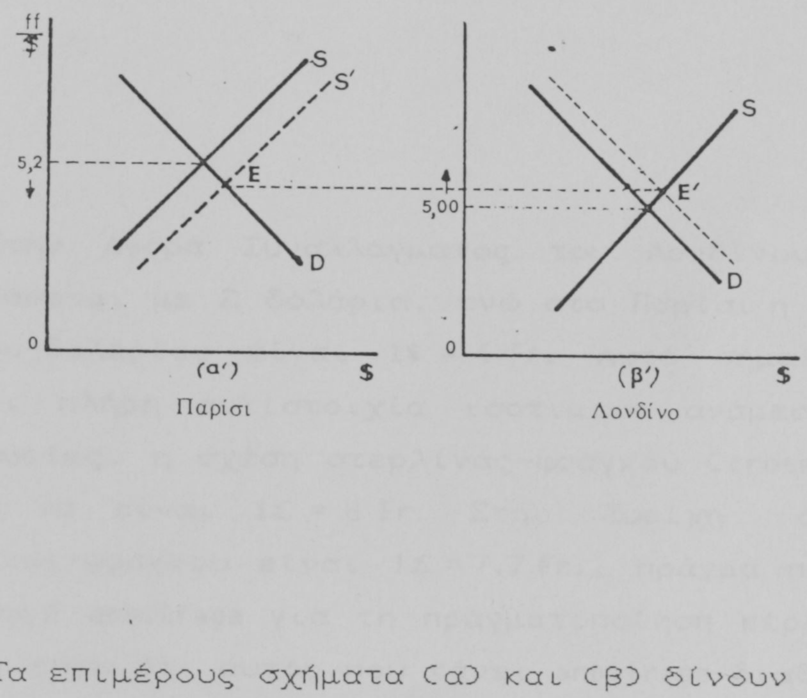
Λονδίνο : 1$ = 5,0 Fr.

Δεδομένου ότι υπάρχουν περιθώρια κέρδους, αυτός που διαθέτει δολάρια θα σπεύσει να αγοράσει γαλλικά φράγκα στο Παρίσι και να πουλήσει συγχρόνως το ίδιο νόμισμα στο Λονδίνο πραγματοποιώντας κέρδος 0.2 Fr. για κάθε δολάριο. Η ενέργεια τους αυτή θα ασκήσει εξισορροπητικές επιδράσεις στις ισοτιμίες των δύο αγορών. Η αύξηση της ζήτησης φράγκων στο Παρίσι θα ασκήσει μια ανοδική επίδραση στην τιμή του φράγκου, ενώ η προσφορά φράγκων στο Λονδίνο θα έχει μια πτωτική επίδραση στην ισοτιμία της αγοράς αυτής. Ετσι θα επικρατήσει τελικά ενιαία τιμή και στις δύο αγορές.

Η εξισωτική επίδραση, απεικονίζεται παρακάτω :



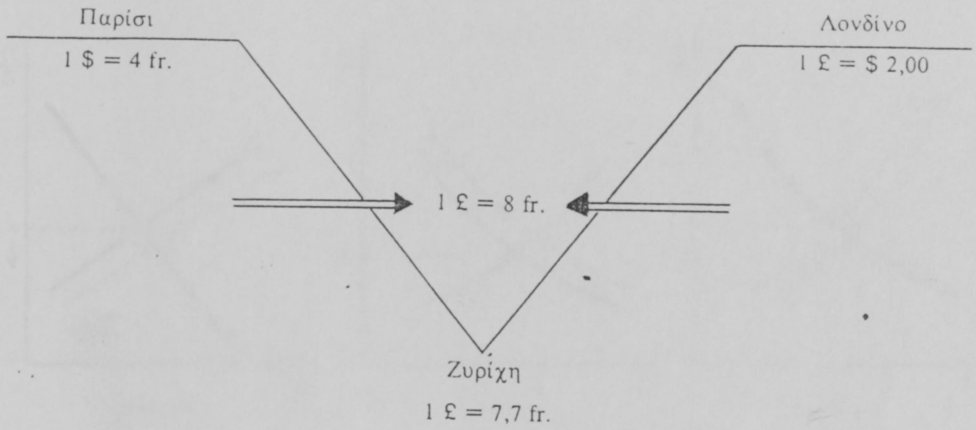
Παρίσι Λονδίνο



Παρίσι Λονδίνο

Τα επιμέρους σχήματα (α) και (β) δίνουν την αρχική ζήτηση και προσφορά του συναλλάγματος (δολαρίων), εκφρασμένη σε γαλλικά φράγκα. Στα διαγράμματα (α') και (β') φαίνεται ότι η αύξηση της προσφοράς δολαρίων (για την αγορά γαλλικών φράγκων) στο Παρίσι θα προκαλέσει μειωτικές τάσεις στην τιμή του δολαρίου. Απο την άλλη, η αγορά δολαρίων στο Λονδίνο (προσφορά φράγκων) θα έχει σαν συνέπεια την αύξηση της τιμής του αμερικάνικου νομίσματος, έτσι ώστε να επιτευχθεί η εξισορρόπηση της ισοτομίας, όπως δείχνει η γραμμή ΕΕ'.

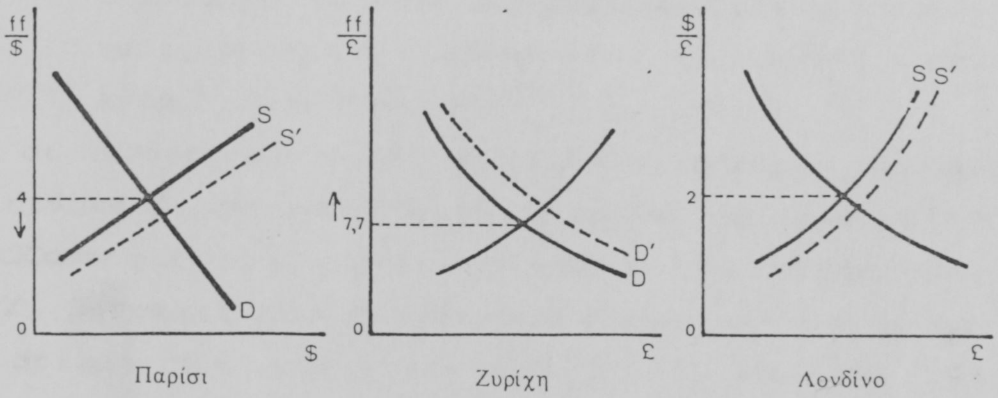
Φυσικά είναι δυνατό το arbitrage συναλλάγματος να αναφέρεται σε αγοραπωλησία περισσοτέρων απο δύο ξένων νομισμάτων, όπως φαίνεται στο παρακάτω παράδειγμα :



Στην Αγορά Συναλλάγματος του Λονδίνου η στερλίνα ανταλλάσεται με 2 δολάρια, ενώ στο Παρίσι η σχέση ανταλλαγής φράγκου-δολαρίου είναι 1$ = 4 Fr. Αυτό σημαίνει ότι για να υπάρχει πλήρη αντιστοιχία ισοτιμιών ανάμεσα στις τρείς κυκλοφορίες, η σχέση στερλίνας-φράγκου (cross exchange rate) θα πρέπει να είναι 1£ = 8 Fr. Στη Ζυρίχη όμως η σχέση στερλίνας-φράγκου είναι 1£ = 7.7 Fr., πράγμα που επιτρέπει την διεξαγωγή arbitrage για τη πραγματοποίηση κέρδους με τον εξής τρόπο: έστω ότι αυτός που κάνει arbitrage διαθέτει δολάρια, θα αγοράσει φράγκα στο Παρίσι, συγχρόνως θα αγοράσει στερλίνες στη Ζυρίχη και ταυτόχρονα θα τις πουλήσει στο Λονδίνο για να αγοράσει δολάρια σύμφωνα με την ισοτιμία 1£ = $2.00. Αυτό του δίνει την δυνατότητα να ανταλλάξει μετά τα 2 $ με 8 Fr. και να κερδίσει 0.3 Fr. για κάθε £. Αν υποτεθεί ότι διαθέτει 10000 $ με αυτά θα αγοράσει 40000 Fr. στο Παρίσι και με αυτά θα αγοράσει

5194.8 £ στη Ζυρίχη και τέλος αυτές, θα τις μετατρέψει σε $ 10389.6 στο Λονδίνο, έτσι θα έχει αποκτήσει κέρδος 389.6 δολαρίων.

Η εξισωτική λειτουργία του arbitrage είναι φανερή, στο επόμενο σχήμα φαίνονται οι επιμέρους αγορές, οι συναλλαγές του πιο πάνω arbitrage απεικονίζονται με τις διακεκομένες γραμμές :



Φαίνεται λοιπόν, ότι σε συνθήκες πλήρους ανταγωνισμού, η πλήρης αντιστοιχία των ισοτιμιών των διαφόρων νομισμάτων εξασφαλίζεται διαμέσου του arbitrage συναλλάγματος. Θα πρέπει όμως εδώ, να τονιστεί ότι στις περισσότερες περιπτώσεις επικρατούν συνθήκες μη-τέλειου ανταγωνισμού, εξαιτίας παρεμβατικής πολιτικής του κράτους ή γιατί η οργάνωση της αγοράς είναι ελλιπής ή ακόμα γιατί η γνώση των δεδομένων είναι ανεπαρκής, με συνέπεια να μην εξασφαλίζεται ενιαία ισοτιμία στις διάφορες αγορές.

1.3.3 Διενέργεια Arbitrage με Μηδενικό Κόστος Μετατροπής [3]

Εδώ πρόκειται για την περίπτωση όπου οι τιμές προσφοράς και ζήτησης των νομισμάτων είναι ίσες (είναι η περίπτωση που εξετάζει η μελέτη), δηλαδή δεν υπολογίζεται και κάποιο κόστος μετατροπής των νομισμάτων. Εστω λοιπόν, ότι υπάρχουν τέσσερα νομίσματα, θα υπάρχουν τότε, 12 διαφορετικές νομισματικές ισοτιμίες. Τα τέσσερα νομίσματα έστω ότι είναι δολάριο ($), στερλίνα (£), γερμανικό μάρκο (DM), και γιέν (¥). Για κάποιον που κρατάει δολάρια, οι σχετικές, με τα άλλα νομίσματα τρέχουσες ισοτιμίες είναι S($/£), S($/DM) και S($/¥) όπου [3]:

> S($/£) καθορίζεται σαν τον αριθμό των δολαρίων που απαιτούνται για την ανταλλαγή με μια στερλίνα στην Τρέχουσα Αγορα Συναλλάγματος. Και γενικότερα S(i/j) είναι ο αριθμός μονάδων του νομίσματος i που απαιτούνται για να αγοραστεί μια μονάδα του νομίσματος j στην Τρέχουσα Αγορά Συναλλάγματος.

Ολες οι πιθανές ισοτιμίες μεταξύ των τεσσάρων νομισμάτων δίνονται στο πίνακα I, που θα λέγεται πίνακας νομισματικών ισοτιμιών. Διαβάζοντας κατά μήκος τον πίνακα I, για παράδειγμα, στη σειρά για £, βρήσκεις τον αριθμό των στερλίνων ανα δολάριο S(£ / $), τον αριθμό των στερλίνων ανα μάρκο S(£ / DM) κλπ.

ΠΙΝΑΚΑΣ ΠΙΘΑΝΩΝ ΝΟΜΙΣΜΑΤΙΚΩΝ ΙΣΟΤΙΜΙΩΝ

Πώληση νομίσματος	Αγορά νομίσματος			
	$	£	DM	¥
$	1	S($/£)	S($/DM)	S($/¥)
£	S(£/$)	1	S(£/DM)	S(£/¥)
DM	S(DM/$)	S(DM/£)	1	S(DM/¥)
¥	S(¥/$)	S(¥/£)	S(¥/DM)	1

Πίνακας I

Παρόλα αυτά οι παραπάνω ισοτιμίες δεν είναι όλες ανεξάρτητες μεταξύ τους, στην πραγματικότητα μερικές είναι περιττές. Κια το πιο φανερό είναι ότι εάν είναι γνωστό το S(£/$), τότε αυτόματα είναι γνωστό και το S($/£). Στην πραγματικότητα όταν δεν υπάρχουν προμήθειες στις συναλλαγές ή κάποιο κόστος συναλλαγής ισχύει :

$$S(\$/£) = \frac{1}{S(£/\$)} \quad (1.1)$$

ή γενικότερα $S(i/j) = \frac{1}{S(j/i)}$.

Ετσι λοιπόν κάποιος που εφαρμόζει arbitrage και έχει δολάρια μπορεί να κινηθεί, π.χ., στον εξής δρόμο :

$$S(DM/\$) \cdot S(£/DM) \cdot S(\$/£) > 1 \quad (1.2)$$

όταν δηλαδή το παραπάνω γινόμενο είναι μεγαλύτερο της μονάδας τότε θα υπάρχει κέρδος, ξεκινώντας δηλαδή με \$1 θα φτάσει στο τέλος να έχει περισσότερα απο \$1.

Δυστυχώς όμως, η ευκαιρία για κέρδος μπορεί να εξαφανιστεί πολύ γρήγορα και αυτό γιατί, όπως αναφέρθηκε και πιο πρίν, υπάρχουν πολλοί διαπραγματευτές συναλλάγματος σε όλο τον κόσμο που είναι έτοιμοι να εκμεταλευτούν μιά τέτοια κατάσταση, πράγμα που όπως εξηγήθηκε πιο πάνω ο ανταγωνισμός που υπάρχει τείνει να δημιουργήσει ενιαίες ισοτιμίες σε όλες τις αγορές.

Στην πραγματικότητα κάποιος επαγγελματίας θα συνεχίσει μέχρι η σχέση (1.2) γίνει :

$$S(DM/\$) \cdot S(£/DM) \cdot S(\$/£) \leq 1 \quad (1.3)$$

Αυτό βέβαια δεν σημαίνει ότι δέν υπάρχουν άλλα περιθώρια κέρδους με τα συγκεκριμένα νομίσματα, θα μπορούσε κανείς π.χ., να ακολουθήσει τον εξής δρόμο :

$$S(£/\$) \cdot S(DM/£) \cdot S(\$/DM)$$

εάν λοιπόν το παραπάνω γινόμενο είναι μεγαλύτερο της μονάδας τότε θα έχει κέρδος. Εάν το γινόμενο είναι μικρότερο ή ίσο της μονάδας είναι δηλαδή :

$$S(£/\$) \cdot S(DM/£) \cdot S(\$/DM) \leq 1$$

πηγαίνοντας στο πίνακα I είναι $S(£/\$) = 1 / [S(\$/£)]$,

$S(DM/£) = 1 / [S(£/DM)]$, και $S(\$/DM) = 1 / [S(DM/\$)]$ και με αντικατάσταση είναι :

$$S(DM/\$) \cdot S(£/DM) \cdot S(\$/£) \geq 1 \qquad (1.4)$$

Αρα λοιπόν μπορεί κάποιος να συνεχίσει απο την μιά ή απο την άλλη κατεύθηνση παίρνοντας κάποιο κέρδος, και αυτό γιατί δεν υπάρχει κόστος μετατροπής, μέχρι να φτάσει στο σημείο όπου :

$$S(DM/\$) \cdot S(£/DM) \cdot S(\$/£) = 1 \qquad (1.5)$$

Αυτός ο κανόνας αλυσίδας που έδωσε την σχέση (1.5) μετά την διενέργεια arbitrage χωρίς κόστος μετατροπής δείχνει, πώς μερικές ισοτιμίες μπορούν να παραχθούν απο άλλες. Η ισοτιμία $S(\$/DM)$ μπορεί να παραχθεί βάζοντας $S(DM/\$) = 1 / [S(\$/DM)]$ η οποία απο την (1.5) δίνει :

$$S(\$/DM) = S(\$/£) \cdot S(£/DM) \qquad (1.6)$$

και στη συνέχεια

$$S(£/DM) = S(\$/DM) / S(\$/£) \qquad (1.7)$$

και

$$S(\$/£) = S(\$/DM) / S(£/DM) \qquad (1.8)$$

Και γενικότερα ισχύει :

$$S(i/j) = S(i/k) \cdot S(k/j)$$

ή

$$S(i/j) = S(k/j) / S(k/i)$$

ή

$$S(i/j) = S(i/k) / S(j/k)$$

όπου και τα $S(i/k)$, $S(k/j)$ μπορούν και αυτά να γραφτούν με διαφορετικούς τρόπους.

Θα πρέπει εδώ να τονιστεί ότι τα παραπάνω αποτελέσματα βγαίνουν εξαιτίας της δυνατότητας διενέργειας arbitrage και εξαιτίας του ότι η αγορά είναι "αποδοτική" με την έννοια του ότι αυτό συμβαίνει μέχρι οι ευκαιρίες απο το arbitrage εξαφανιστούν. Ετσι λοιπόν με βάση τις σχέσεις (1.6)-(1.8) βγαίνει ο πίνακας II. Είναι σημαντικό να τονιστεί ότι οι ισοτιμίες στο πίνακα II βγαίνουν απο τρείς νομισματικές ισοτιμίες : S($/£), S($/DM) και S($/¥). Δηλαδή γνωρίζοντας την ισοτιμία κάθε νομίσματος σε σχέση με το δολάριο, μπορεί κάποιος να γνωρίζει την ισοτιμία κάθε νομίσματος σε σχέση με οποιοδήποτε άλλο νόμισμα. Γενικότερα, εάν n είναι ο αριθμός των νομισμάτων, οι n(n-1) πιθανές ισοτιμίες μπορούν όλες να υπολογιστούν απο τις (n-1) πιθανές ισοτιμίες σε σχέση με το n-οστό νόμισμα. Οταν υπολογίζονται συναλλαγματικές ισοτιμίες απο άλλες ισοτιμίες, βάση των σχέσεων (1.6)-(1.8) οι ισοτιμίες που υπολογίστηκαν καλούνται "cross exchange rates"

ΤΡΕΧΟΥΣΕΣ ΝΟΜΙΣΜΑΤΙΚΕΣ ΙΣΟΤΙΜΙΕΣ ΜΕ ARBITRAGE

Πώληση νομίσματος	Αγορά νομίσματος			
	$	£	DM	¥
$	1	S($/£)	S($/DM)	S($/¥)
£	1/[S($/£)]	1	S($/DM)/S($/£)	S($/¥)/S($/£)
DM	1/[S($/DM)]	S($/£)/S($/DM)	1	S($/¥)/S($/DM)
¥	1/[S($/¥)]	S($/£)/S($/¥)	S($/DM)/S($/¥)	1

ΠΙΝΑΚΑΣ II

Στο σημείο αυτό θα δωθεί ένα παράδειγμα για το πως μπορεί να βγεί ενας πίνακας "cross exchange rates" εαν δωθούν οι ισοτιμίες σε σχέση με το δολάριο (οι ισοτιμίες αυτές δίνονται καθημερινά απο τις εμπορικές τράπεζες.

Εστω λοιπόν οτι η τράπεζα δίνει τις εξής ισοτιμίες [3] :

$ / £ = 1.8930

$ / DM = 0.4535

$ / ¥ = 0.004372

και εφαρμόζωντας τους τύπους του πίνακα II έπεται ο ακόλουθος πίνακας :

ΑΓΟΡΑ

	$	£.	DM	¥	
$	1	1.8930	0.4535	0.004372	ΠΩΛΗΣΗ
£	0.5282	1	0.2396	0.002309	
DM	2.2050	4.1742	1	0.009640	
¥	228.73	432.98	103.73	1	

Πίνακας cross exchange rates

Δεν υπάρχει κάποιος κανόνας που να λέει ότι οι ισοτιμίες πρέπει να αναφέρονται όλες σε σχέση με το δολάριο, κάθε νόμισμα μπορεί να είναι νόμισμα αναφοράς. Παρόλα αυτά, είναι σημαντικό το νόμισμα αναφοράς να είναι γενικά αποδεκτό. Στην αρχή του 20-στού αιώνα η στερλίνα ήταν το νόμισμα ανοφοράς, μετά όμως το 1944 με την Συμφωνία του Bretton Woods το δολάριο καθορίστηκε σαν το βασικό νόμισμα για τον καθορισμό των άλλων ισοτιμιών. Πρέπει απλά να τονιστεί, ότι μπορούν να υπολογιστούν οι ισοτιμίες όχι μόνο σε σχέση με το n-οστό νόμισμα, αλλά αντί αυτού μπορεί να χρησιμοποιηθεί η τιμή των νομισμάτων σε σχέση με κάποιο αγαθό (commodity) όπως είναι ο χρυσός. Με αυτό όμως δεν ασχολείται αυτή η μελέτη. Στη συνέχεια γίνεται παρουσίαση της Προθεσμιακής Αγοράς Συναλλάγματος [2].

1.4 Η Προθεσμιακή Αγορά Συναλλάγματος

Απο τα προηγούμενα φαίνεται ότι, στην χρηματοδότηση της διεθνούς ανταλλαγής αγαθών και υπηρεσιών διαμέσου της Αγοράς Συναλλάγματος, πολλές φορές παρεμβάλεται ένα χρονικό διάστημα ανάμεσα στη σύναψη της συμφωνίας της εμπορικής πληρωμής, και της πραγματοποιησής της. Αυτό συνεπάγεται την ανάληψη κινδύνου για τους συναλλασσόμενους, εξαιτίας των μεταβολών που μπορεί να σημειωθούν στις ισοτιμίες κατά τη διάρκεια αυτού του διαστήματος. Οι συναλλασσόμενοι, λοιπόν, για να διασφαλιστούν απο τούς κινδύνους που συνεπάγεται μιά μεταβολή των ισοτιμιών, μπορούν να χρησιμοποιήσουν τις ευκολίες του τμήματος εκείνου της Αγοράς Συναλλάγματος που είναι γνωστό σαν **Προθεσμιακή Αγορά** (forward Market). Η αγορά αυτή επιτρέπει να συναφθεί συμφωνία για την πώληση ή αγορά συναλλάγματος σε προκαθορισμένη ημερομηνία στο μέλλον και σε τιμή συναλλάγματος που καθορίζεται την στιγμή που συνάπτεται η συμφωνία.

Η Αγορά Συναλλάγματος λοιπόν, επιτελεί διαμέσου του τμήματος, της Προθεσμιακής Αγοράς και μια τρίτη λειτουργία, πέρα απο τις δύο πρώτες, δηλαδή τη μεταφορά αγοραστικής αξίας σε ξένο νόμισμα και της παροχής πιστώσεων. Η τρίτη αυτή λειτουργία, είναι η διασφάλιση απέναντι στους κινδύνους διακύμανσης των ισοτιμιών και είναι γνωστή διεθνώς hedging [3].

1.4.1 Καθορισμός της Προθεσμιακής Ισοτιμίας

Εκτός από το καθορισμό της προθεσμιακής ισοτιμίας συναλλάγματος θα αναφερθεί και ποια είναι η σχέση της με την ισοτιμία όψης. Για να καθοριστεί η σχέση αυτή θα πρέπει να εισαχθεί η έννοια μίας τρίτης ισοτιμίας, της ισοτιμίας επιτοκίων, όπως και η έννοια του arbitrage επιτοκίων (interest arbitrage) [2].

Ας υποτεθεί ότι, στα δύο μεγάλα χρηματοδοτικά κέντρα, αυτά της Νέας Υόρκης και του Λονδίνου τα επιτόκια δανεισμού, βραχυχρόνια είναι 9% και 10%, αντίστοιχα. Με δοσμένη την τρέχουσα ισοτιμία των δύο νομισμάτων, π.χ. 1 £ = $ 3 και προκειμένου να καρπωθεί κάποιος τη διαφορά των επιτοκίων, θα αγόραζε στερλίνες στην τρέχουσα αγορά για να τις επενδύσει στο Λονδίνο [2]. Παράλληλα, για να καλυφθεί από τις ενδεχόμενες μεταβολές στην τιμή συναλλάγματος, θα πουλούσε στερλίνες στην προθεσμιακή αγορά. Η συναλλαγή αυτή θα είχε σαν αποτέλεσμα να υψωθεί η τιμή του συναλλάγματος στην τρέχουσα αγορά και να μειωθεί η προθεσμιακή τιμή, εξαιτίας της αύξησης της προσφοράς προθεσμιακού συναλλάγματος. Η διαφορά μεταξύ των δύο ισοτιμιών, όταν εκφρασθεί σαν ετήσιο ποσοστό, πρέπει να ισούται, σε κατάσταση ισορροπίας, με τη διαφορά των επιτοκίων. Η διαφορά αυτή εξασφαλίζει την ισοτιμία, με την έννοια ότι μετά τις μεταβολές που σημειώθηκαν στις τιμές στην τρέχουσα και στην προθεσμιακή αγορά συναλλάγματος, η 1 £ στο Λονδίνο αποδίδει σε εισόδημα όσο και 3 $ στην Νέα Υόρκη. Η ευθυγράμμιση αυτής της τιμής του χρήματος στις δύο αγορές σημειώθηκε με το arbitrage επιτοκίων [2].

Στο παραπάνω παράδειγμα, εξαιτίας της διαφοράς του προεξοφλητικού επιτοκίου στις δύο αγορές (δηλαδή το 1%), η στερλίνα προθεσμίας πωλείται σε χαμηλότερη τιμή από εκείνη στην τρέχουσα αγορά. Η διαφορά αυτή ανάμεσα στις δύο τιμές δίνει το **προεξοφλητικό** περιθώριο και ισούται, όταν εκφράζεται σαν ποσοστό, με τη διαφορά των επιτοκίων. Αν το επιτόκιο του Λονδίνου είναι χαμηλότερο από εκείνο της Νέας Υόρκης, τότε η προθεσμιακή τιμή της στερλίνας θα είναι ψηλότερη από εκείνη στην τρέχουσα αγορά. Το πριμ αυτό της τιμής συναλλάγματος στην Προθεσμιακή Αγορά ποσοστιαία, θα ισούται επίσης, με τη διαφορά των επιτοκίων στις δύο αγορές χρήματος.

Περιληπτικά, εάν υπάρχουν διαφορές στα επιτόκια i, στις αγορές χρήματος δύο χωρών A (εγχώρια αγορά) και B (αλλοδαπή αγορά) τότε ισχύει η παρακάτω σχέση [2]:

Εάν $i_B > i_A$ όπου i το επιτόκιο

τότε $\tau_\pi < \tau_o$

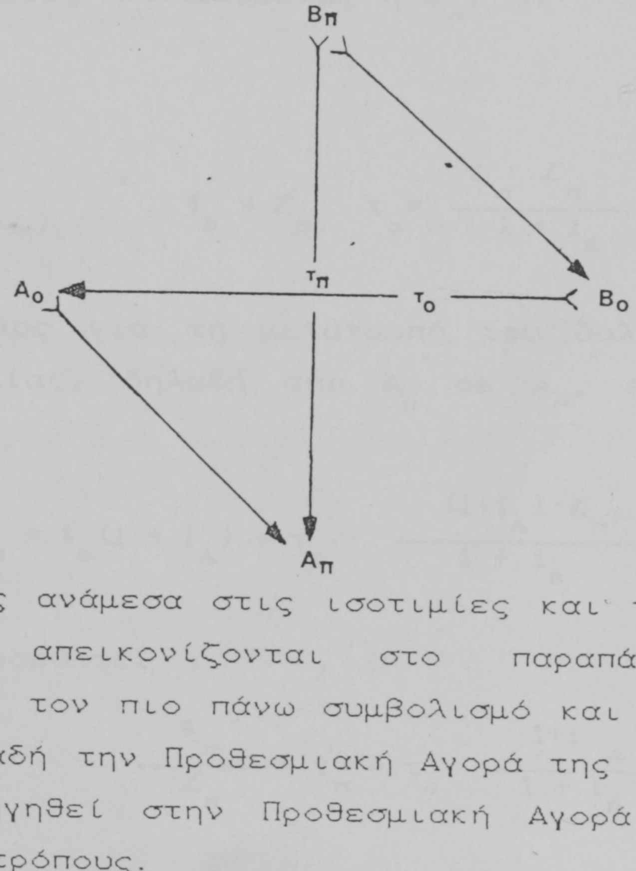
όπου τ η ισοτιμία και οι δείκτες **π** και **ο** είναι η προθεσμιακή αγορά και η τρέχουσα αγορά αντίστοιχα. Δηλαδή το προθεσμιακό συνάλλαγμα θα πουλιέται με προεξοφλητικό περιθώριο. Αντίθετα εάν $i_A > i_B$

τότε $\tau_{\pi} > \tau_{o}$,

δηλαδή το νόμισμα της χώρας με το χαμηλότερο επιτόκιο θα πουλιέται στην Προθεσμιακή Αγορά με πρίμ εναντι της τιμής του στην Τρέχουσα Αγορά [2].

1.4.2 Arbitrage Επιτοκίων και Ισοτιμία

Η διαφορά λοιπόν, ανάμεσα στην προθεσμιακή ισοτιμία και στην τρέχουσα ισοτιμία αντιστοιχεί με τη διαφορά των επιτοκίων των χωρών στις οποίες γίνεται η αναφορά. Σε περίπτωση μη αντιστοιχίας ανάμεσα στα διάφορα επιτόκια και στις ισοτιμίες θα διενεργηθεί arbitrage επιτοκίων. Είναι ευνόητο ότι αν δεν υπάρχει διαφορά επιτοκίων στις δύο χώρες, η προθεσμιακή ισοτιμία και η τρέχουσα ισοτιμία είναι ίδιες [2].



Οι σχέσεις ανάμεσα στις ισοτιμίες και τα επιτόκια δύο χωρών A και B απεικονίζονται στο παραπάνω διάγραμμα. Χρησιμοποιώντας τον πιο πάνω συμβολισμό και με αφετηρία το σημείο A_{π}, δηλαδή την Προθεσμιακή Αγορά της χώρας A, μπορεί κάποιος να οδηγηθεί στην Προθεσμιακή Αγορά της B με δύο διαφορετικούς τρόπους.

Ο πρώτος είναι η απευθείας σύνδεση των δύο χωρών διαμέσου της προθεσμιακής ισοτιμίας τ_{π}, όπως φαίνεται στο σχήμα. Ο δεύτερος τρόπος είναι έμμεσος και χρησιμοποιεί τις σχέσεις που υπάρχουν ανάμεσα στις ισοτιμίες και στα επιτόκια των δύο χωρών. Για παράδειγμα, μια στερλίνα στο σημείο B_{π}, στην Προθεσμιακή

Αγορά δηλαδή, ισούται με $1£\cdot(1 + i_B)$. Δηλαδή η προθεσμιακή στερλίνα $£_\pi$ ισούται με την στερλίνα όψης £, σύν τον τόκο για το υποτιθέμενο χρονικό διάστημα, δηλαδή $£_\pi = £\ (1 + i_B)$. Επιστρέφοντας στο σημείο B_π επομένως, και γνωρίζοντας ότι Α και Β είναι οι τρέχουσες αγορές της χώρας Α π.χ. \$ και της χώρας Β π.χ. £ αντίστοιχα, θα ισχύει [2]:

(α) Απο το σημείο B_π στο B_o με προεξόφληση μιας προθεσμιακής στερλίνας έπεται:

$$1£_o = \frac{1£_\pi}{(1 + i_B)} \qquad (1)$$

(β) Απο το σημείο B_o στο A_o η σχέση που υπάρχει ανάμεσα στις ισοτιμίες είναι αυτή της τρέχουσας ισοτιμίας τ_o. Επομένως η $£_o$, αν μετατραπεί σε $\$_o$ έπεται:

$$\$_o = £_o \cdot \tau_o = \frac{\tau_o \cdot £_\pi}{1 + i_B} \qquad (2)$$

(γ) Τέλος για τη μετατροπή του δολαρίου όψης, σε δολάριο προθεσμίας, δηλαδή απο A_o σε A_π, έπεται :

$$\$_\pi = \$_o(1 + i_A) = \tau_o \cdot \frac{(1+i_A)\cdot £_\pi}{1 + i_B} \qquad (3)$$

Απο την (3) προκύπτει :

$$\frac{\$_\pi}{£_\pi} = \tau_\pi = \tau_o \cdot \frac{1+i_A}{1 + i_B} \qquad (4)$$

που δείχνει τη σχέση μεταξύ προθεσμιακής ισοτιμίας και τρέχουσας. Απο την (4) προκύπτει ότι αν $i_B = i_A$ τότε και $\tau_\pi = \tau_o$. Αν το επιτόκιο της χώρας Β είναι 10% και της Α 8% και η τρέχουσα ισοτιμία, τ_o, είναι 1£ = \$3, τότε αντικαθιστώντας την (4) βρήσκεται η προθεσμιακή ισοτιμία ενός έτους ίση με 2,94 \$. Δηλαδή, με τα δωθέντα επιτόκια 10% και 8% και με τρέχουσα ισοτιμία 1£ = $3 η προθεσμιακή στερλίνα ενός έτους θα πρέπει να πουλιέται με προεξοφλητικό περιθώριο 2% στην τρέχουσα τιμή δηλαδή 2% × 3$ = 0.06 $. Αν το προεξοφλητικό αυτό περιθώριο, εκφρασμένο σε ποσοστό, ήταν μικρότερο, δηλαδή η προθεσμιακή στερλίνα πουλιόταν σε τιμή ψηλότερη απο 2.94$ θα υπήρχε εκροεί κεφαλαίου απο την χώρα Α στη χώρα Β. Οι χρηματιστές της χώρας Α θα αγόραζαν στερλίνες όψης στη χώρα Β και θα τις πουλούσαν στην προθεσμιακή αγορά, γιατί στην πργματικότητα θα έπαιρναν ποσοστό μεγαλύτερο απο 10%, εφόσον η προθεσμιακή στερλίνα θα πουλιόταν σε τιμή ψηλότερη απο 2.94 $ [2].

Πρέπει να τονιστεί εδώ, ότι στην πραγματικότητα δεν υπάρχει πλήρης αντιστοιχία ισοτιμίας επιτοκίων και κυκλοφοριών. Παρατηρούνται μικρές παρεκκλίσεις, που δικαιολογούνται απο την αμοιβή των τραπεζών για τη διενέργεια του **arbitrage** επιτοκίων. Στις αγορές που δεν επικρατούν πλήρως ανταγωνιστικές συνθήκες αναμένονται μεγαλύτερες παρεκκλίσεις.

1.4.3 Arbitrage και Ισοδύναμη Προθεσμιακή Ισοτιμία [2]

Απο τα προηγούμενα λοιπόν φαίνεται ότι η κατεύθυνση της ροής κεφαλαίων καθορίζεται απο τη διαφορά στα επιτόκια της αγοράς χρήματος στις διάφορες χώρες και τη διαφορά ανάμεσα στις ισοτιμίες όψης και προθεσμία.

Χρησιμοποιώντας τον διεθνή συμβολισμό είναι [2] :

r_f : ισοτιμία στην Προθεσμιακή Αγορά

r_s : ισοτιμία στην Τρέχουσα Αγορά (ή Αγορά Οψης)

i_a : το επιτόκιο στη χώρα Α (έστω ότι είναι η εγχώρια οικονομία)

i_b : το επιτόκιο στη χώρα Β (έστω ότι είναι η αλλοδάπη οικονομία)

οπότε η (4) γράφεται

$$r_f = \frac{1 + i_a}{1 + i_b} \cdot r_s \qquad (4\alpha)$$

αν υποτεθεί τώρα ότι i_a και i_b είναι τα επιτόκια τριών μηνών, απο την (4α) φαίνεται ότι μια χρηματική μονάδα αν τοκιστεί με επιτόκιο i_a θα είναι στο τέλος της τριμηνίας ίση με $1 + i_a$. Χρησιμοποιώντας τώρα την αλλοδαπή αγορά, η αξία της χρηματικής μονάδας, όταν μετατραπεί σε νόμισμα της χώρας Β και τοκιστεί με επιτόκιο i_b θα είναι :

$$(1 + i_b) \cdot \frac{1}{r_s}$$

Και για να γίνει κάλυψη για τον κίνδυνο υποτίμησης, θα πρέπει το ξένο νόμισμα να πουληθεί στην Προθεσμιακή Αγορά, ώστε τελικά η αξία της χρηματικής μονάδας να είναι :

$$r_f \cdot (1 + i_b) \cdot \frac{1}{r_s}$$

Για να αποφασίσει κάποιος σε πιά αγορά (εγχώρια ή αλλοδαπή) συμφέρει να τοποθετήση τα διαθέσημα κεφάλαια θα πρέπει να συγκρίνη την τελική αξία της χρηματικής μονάδας στις δύο αγορές. Ετσι αν ισχύει

$$1 + i_a > r_f \cdot (1 + i_b) \cdot \frac{1}{r_s} \qquad (4\beta)$$

η αγορά της χώρας Α θα έλξει την επένδυση κεφαλαίου. Αντίθετα αν ήταν

$$1 + i_a < r_f \cdot (1 + i_b) \cdot \frac{1}{r_s}$$

τότε συμφέρει η αγορά της Β. Τέλος είναι φανερό ότι σε περίπτωση ισότητας, δηλαδή :

$$1 + i_a = r_f \cdot (1 + i_b) \cdot \frac{1}{r_s} \qquad (5)$$

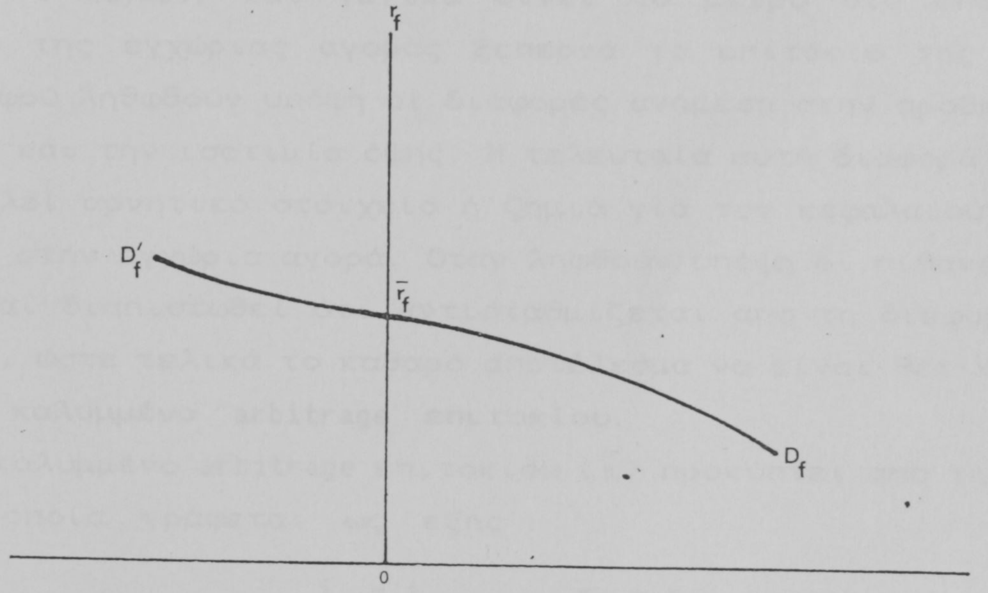
δεν θα γίνει μετακίνηση κεφαλαίων. Η τελευταία περίπτωση αντιστοιχεί στο προηγούμενο παράδειγμα, όπου η διαφορά των επιτοκίων μεταξύ εγχώριας αγοράς και αλλοδαπής (2%) είναι ίση με τη διαφορά, εκφρασμένη σαν ποσοστό, μεταξύ της προθεσμιακής ισοτιμίας και τις τρέχουσας ισοτομίας ($1\ £_f = \$\ 2.94$).

Παρατηρήται ότι η (5) είναι η (4α) σε διαφορετική μορφή.

Αλλά η (5) είναι ισοδύναμη με τη σχέση

$$r_f = \frac{i_a - i_b}{1 + i_b} \cdot r_s + r_s \qquad (6)$$

Η σχέση αυτή δίνει την ισοδύναμη προθεσμιακή ισοτιμία (parity forward exchange rate) ή ισοτιμία προθεσμιακού συναλλάγματος για την οποία, με δοσμένα επιτόκια i_a, i_b και δοσμένη τρέχουσα ισοτιμία r_s, δεν γίνεται arbitrage επιτοκίων. Και πάλι ο λόγος είναι ότι η διαφορά επιτοκίων αντισταθμίζεται ακριβώς απο τη διαφορά μεταξύ προθεσμιακής ισοτιμίας και τρεχουσας. Στο επόμενο σχήμα απεικονίζεται η πιο πάνω συναρτησιακή σχέση ζήτησης προθεσμιακού συναλλάγματος, στο παράδειγμα είναι η ζήτηση του προθεσμιακού νομίσματος της χώρας Β, απο αυτούς που διενεργούν arbitrage [2] :



Προθεσμιακό συνάλλαγμα Προθεσμιακό συνάλλαγμα

Εστω ότι με $\bar{r}_f$ συμβολίζεται η ισοδύναμη προθεσμιακή ισοτιμία, απο το σχήμα φαίνεται ότι η ζήτηση για προθεσμιακό συνάλλαγμα που αντιστοιχεί στην $\bar{r}_f$ είναι ίση με μηδέν. Για τιμές του r_f μικρότερες του $\bar{r}_f$, δηλαδή η σχέση (4β), αυτοί που διενεργούν arbitrage, αγοράζουν προθεσμιακό συνάλλαγμα όπως δείχνει το τμήμα $\bar{r}_f D_f$ της καμπύλης ζήτησης. Αυτό σημαίνει ότι η διαφορά των επιτοκίων $i_a - i_b$ υπεραντισταθμίζει τη διαφορά ισοτιμιών όψης και προθεσμίας ώστε να επενδύονται κεφάλαι στην οικονομία Α απο τους κεφαλαιούχους της χώρας Β. Το αντίθετο συμβαίνει όταν $r_f >$ $\bar{r}_f$ όπως δείχνει το τμήμα $D'_f \bar{r}_f$ της καμπύλης ζήτησηςπροθεσμιακού συναλλάγματος. Αυτοί που διενεργούν arbitrage θα πουλήσουν προθεσμιακό συνάλλαγμα για να καλυφθούν απο την πιθανότητα υποτίμησης της κυκλοφορίας της χώρας Β όπου επενδύουν στν τρέχουσα αγορά. Θα αυξηθεί λοιπόν, η προσφορά προθεσμιακού συναλλάγματος όπως δείχνει το τμήμα $D_f \bar{r}_f$ (αρνητική ζήτηση) στο σχήμα.

1.4.4 Καλυμμένο Arbitrage Επιτοκίου [2]

Την έννοια του καλυμμένου arbitrage επιτοκίου πρώτος ανέπτυξε ο Keynes, και γενικά δίνει το μέτρο στο οποίο το επιτόκιο της εγχώριας αγοράς ξεπερνά το επιτόκιο της ξένης αγοράς αφού ληθφθούν υπόψη οι διαφορές ανάμεσα στην προθεσμιακή ισοτιμία και την ισοτιμία όψης. Η τελευταία αυτή διαφορά μπορεί να αποτελεί αρνητικό στοιχείο ή ζημιά για τον κεφαλαιούχο που επενδύει στην εγχώρια αγορά. Οταν ληφθούν υπόψη οι πιθανές αυτές ζημιές και διαπιστωθεί ότι αντισταθμίζεται απο τη διαφορά στα επιτόκια, ώστε τελικά το καθαρό αποτέλεσμα να είναι θετικό τότε υπάρχει **καλυμμένο arbitrage επιτοκίου**.

Το καλυμμένο arbitrage επιτοκίου ($\bar{\kappa}$) προκύπτει απο τη σχέση (6) οι οποία γράφεται ως εξής :

$$\bar{\kappa} = \frac{i_a - i_b}{1 + i_b} - \frac{r_f - r_s}{r_s} \qquad (7)$$

Στη μορφή αυτή η σχέση (7) δείχνει και αλγεβρικά αυτό που ανφέρθηκε πιο πάνω σχετικά με τη σύγκριση της διαφοράς των επιτοκίων $i_a - i_b$, με τη διαφορά ισοτιμιών εκφρασμένη σε ποσοστά $\frac{r_f - r_s}{r_s}$. Πράγματι, λαμβάνοντας υπόψην ότι η διαίρεση του πρώτου όρου της (7) με $1 + i_b$ δεν επηρεάζει σημαντικά την αξία του, ο όρος αυτός μπορεί να γραφτεί κατα προσέγγιση σαν $i_a - i_b$, και δίνει τη διαφορά των επιτοκίων στις χώρες Α και Β. Ετσι στο μέτρο που ο πρώτος όρος είναι μεγαλύτερος απο το δεύτερο, θα είναι $\bar{\kappa} > 0$.

Η σχέση (7) μπορεί τώρα να χρησιμοποιηθεί για να ερμηνεύση τη ζήτηση για προθεσμικό συνάλλαγμα, που φαίνεται στο πρηγούμενο σχήμα. Φαίνεται δηλαδή όταν

$$r_f < \bar{r}_f$$

τότε θα είναι

$$(\bar{\kappa}) > 0$$

πράγμα που σημαίνει ότι η εγχώρια αγορά προσελκύει τα ξένα κεφάλαια για επένδυση. Για την κάλυψη των επενδύσεων, αυτοί που διενεργούν arbitrage θα αγοράσουν προθεσμιακά το νόμισμα της χώρας τους ώστε το τμήμα $\bar{r}_f D_f$ να είναι η ζήτηση προθεσμιακού συναλλάγματος.

Οταν

$$r_f > \bar{r}_f$$

θα είναι

$$(\bar{\kappa}) < 0$$

και η επένδυση εγχωρίων κεφαλαίων γίνεται στην αλλοδαπή αγορά. Σαν αποτέλεσμα η πώληση του προθεσμιακού συναλλάγματος είναι αναγκαία για κάλυψη και έτσι στην πραγματικότητα η (αρνητική) ζήτηση προθεσμιακού συναλλάγματος αποτελεί την προσφορά του.

ΒΙΒΛΙΟΓΡΑΦΙΑ ΚΕΦΑΛΑΙΟΥ 1

ΕΛΛΗΝΙΚΗ

1. ΓΕΩΡΓΑΚΟΠΟΥΛΟΣ , Θ. Εισαγωγή στην πολιτική οικονομία. Αθήνα , 1982.

2. ΠΟΥΡΝΑΡΑΚΗΣ , Ε. , Διεθνές οικονομικές σχέσεις Αθήνα , 1981

ΞΕΝΗ

3. LEVI, M. , International Finance , Mc Graw Hill, New York , 1983.

4. SALKIN G., CHROSTOFIDES N., HEWINS R., Graph theoretic approaches to foreign exchange operations

ΚΕΦΑΛΑΙΟ 2

ΘΕΩΡΙΑ ΡΟΩΝ

2.1 Εισαγωγή

Μιλώντας γενικά, μια *ροή* είναι ένας τρόπος να στέλνεις αντικείμενα απο ενα μέρος σε ένα άλλο. Γιά παράδειγμα, η αποστολή ετοίμων αγαθών απο την παραγωγή στη κατανάλωση, η μεταφορά ανθρώπων απο τα σπίτια τους στο μέρος που εργάζονται, ή η μεταφορά γραμμάτων απο το σημείο του ταχυδρομείου στον τελικό προορισμό τους μπορούν όλα να θεωρηθούν σαν προβλήματα ροών.

Αναλύοντας τις διάφορες παραλλαγές που μπορεί να αντμετωπίσει κάποιος, παίρνονται πολλά παραπλήσια προβλήματα ροών. Για παράδειγμα, μπορεί να θέλει κανείς να μεγιστοποιήσει το ποσό των αντικειμένων που μεταφέρεται απο ένα μέρος σε ένα άλλο απο ένα σύστημα παράδοσης, ή να θέλει να υπολογίσει το δρόμο με το ελάχιστο κόστος για να στείλει ενα δωθέντα αριθμό απο αντικείμενα απο ένα μέρος σε ένα άλλο διά μέσου ενός συστήματος, ή να θέλει να υπολογίσει το γρηγορότερο δρόμο για να παραδόσει μια αποστολή αγαθών δια μέσου του συστήματος διανομής [3].

Εδώ θα αναφερθούν προβλήματα ροών σε ένα γράφημα, γενικά μια ροή σε ένα γράφημα είναι ένας τρόπος να στέλνεις αντικείμενα απο μια κορυφή σε μια άλλη δια μέσου των τόξων. Η κορυφή απο την οποία τα αντικείμενα ξεκινούν καλείται *πηγή* και συνήθως συμβολίζεται με s. Η κορυφή στην οποία καταλήγουν τα αντικείμενα καλείται *αποδέκτης* και συμβολίζεται με t. Τα αντικείμενα που ταξιδεύουν ή ρέουν απο την πηγή στον αποδέκτη καλούνατι *μονάδες ροής* ή *μονάδες*. Οπως ανάφερθηκε και πιο πάνω, οι μονάδες ροής μπορούν να είναι έτοιμα προιόντα, άνθρωποι γράμματα ή σχεδόν οτιδήποτε [2].

2.2 Ροές με τόξα κέρδους

Στίς περισσότερες περιπτώσεις η μονάδα ροής που εισέρχεται σε ένα τόξο, η ίδια ροή εξέρχεται χωρις να έχει αλλάξει. Σε ένα, όμως, μεγάλο αριθμό προβλημάτων αυτό δεν είναι αποδεκτό. Για παράδειγμα σε δίκτυα που γίνεται μεταφορά υγρών (π.χ. πετρελαιαγωγός) ή ηλεκτρικά δίκτυα απώλειες του συστήματος επιφέρουν χάσιμο ροής κατά μήκος ενός τόξου του συστήματος. Σε μια διαδικασία παραγωγής η οποία μπορεί να παρουσιαστή απο ένα γράφημα με τα τόξα να αντιπροσωπεύουν τις λειτουργίες, η τιμή των υλικών που αφήνουν ένα τόξο είναι μεγαλύτερη απο την τιμή των υλικών που εισέρχεται στο τόξο δηλ. υπάρχει ένα κέρδος που σχετίζεται με ένα τόξο που αντιπροσωπεύει την 'προσθετική αξία' τις λειτουργίας. Στο πρόβλημα των νομισματικών συναλλαγών (δηλ. την αγορά και πώληση νομισμάτων), το κερδος τόξου αντιπροσωπεύει την νομισματική ισοτιμία που μετετρέπει την εισερχόμενη ροή (που μετριέται σε ένα νόμισμα) σε εξερχόμενη ροή (που μετριέται σε ενα διαφορετικό νόμισμα) [4].

Ετσι λοιπόν απο εδω και στο εξής θα συμβολίζεται με g_{ij} ο συντελεστής κέρδους του τόξου (x_i,x_j) (θα είναι πάντα μη αρνητικός αριθμός) επ ίσης με q_{ij} θα συμβολίζεται η χωρητικότητα του ίδιου τόξου (δηλ. η μέγιστη τιμή ροής που μπορεί να περάσει απο αυτό το τόξο). Εαν συμβολίζεται με [3] ξ^{θ}_{ij} την εισερχόμενη ροή στο τόξο (x_i,x_j) και με ξ^{o}_{ij} την εξερχόμενη ροή του ίδιου τόξου τότε θα ισχύει [3] :

$$\xi^{o}_{ij} = \xi^{\theta}_{ij} \cdot g_{ij} \qquad (2.1)$$

Επ ίσης πρέπει να σημειωθεί οτι η χωρητικότητα τόξου αναφέρεται στην εισερχόμενη ροή, έτσι λοιπόν για οποιοδήποτε τόξο ισχύει [3] :

$$\xi^{\theta}_{ij} \leq q_{ij} \qquad (2.2)$$

χωρίς να λαμβάνεται υπόψιν η εξερχόμενη ροή του τόξου.

Ετσι γίνεται αναφορά σε ένα γράφημα G (όπου με x_i συμβολίζεται μιά κορυφή του γραφήματος) με τόξα κέρδους, όπου κάθε τόξο (x_i,x_j) σχετίζεται με ένα συντελεστή g_{ij}, με μιά χωρητικότητα

q_{ij} και με μια υπάρχουσα εφικτή ροή $\Xi = (\xi^{\theta}, \xi^{o})$ (απο εδώ και στο εξής όταν αναφέρεται η ροή ξ_{ij} θα εννοείται η εισερχόμενη ροή του τόξου (x_i, x_j)). Το γράφημα αυτό θα συμβολίζεται με $G(\Xi)$. Σκοπός λοιπόν, της μελέτης είναι να βρεθεί η μέγιστη εφικτή ροή απο την κορυφή πηγή s μέχρι την κορυφή αποδέκτη t [3].

2.2.1 Αυξητικές αλυσίδες.

Μια αλυσίδα απο τόξα, που ξεκινάει απο την κορυφή x_{i_1} και καταλήγη στη κορυφή x_{i_k} καλείται αυξητική αλυσίδα απο την x_{i_1} στη x_{i_k} εάν μπορεί να σταλεί κάποια μονάδα ροής μέσα στο $G(\Xi)$ κατα μήκος αυτής της αλυσίδας. Εάν η αλυσίδα δίνεται απο ένα σύνολο κορυφων $x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ τότε με F θα συμβολίζεται το σύνολο όλων των "πρός τα μπρός" τόξων αυτής της αλυσίδας δηλ. τα τόξα $(x_{i_p}, x_{i_{p+1}})$ με $x_{i_{p+1}} \in \Gamma(x_{i_p})$ και με B θα συμβολίζεται το σύνολο όλων των "προς τα πίσω" τόξων δηλ. τα τόξα με $x_{i_p} \in \Gamma(x_{i_{p+1}})$. Μιά αλυσίδα είναι τότε αυξητική αλυσίδα εάν για κάθε "προς τα μπρός" τόξο $(x_i, x_j) \in F$ είναι $\xi^{\theta}_{ij} < q_{ij}$ και για κάθε "πρός τα πίσω" τόξο $(x_j, x_i) \in B$ έχουμε $\xi^{\theta}_{ji} > 0$ [3].

A. *Κέρδος αλυσίδας* [3]

Το κέρδος μιάς αλυσίδας $S = x_{i_1}, x_{i_2}, \ldots, x_{i_k}$ δίνεται απο :

$$g(S) = \prod_{(x_i, x_j) \in F} g_{ij} \cdot \prod_{(x_j, x_i) \in B} 1 / g_{ji} \qquad 2.3$$

Ε. Χωρητικότητα αλυσίδας [3]

Η προσαυξημένη χωρητικότητα μιας αυξητικής αλυσίδας S είναι η μέγιστη εισερχόμενη ροή της x_{i_p} η οποία μπορεί να σταλεί κατά μήκος της αλυσίδας πρός την x_{i_k}, μέχρι όπου ή η ροή σε κάποιο "προς τα μπρός" τόξο κορεστή (γίνει δηλ. ίση με την χωρητικότητα του τόξου) ή η ροή σε κάποιο "προς τα πίσω τόξο" γίνει ίση με μηδέν.

Εάν S είναι μια απλή αλυσίδα, και μια ροή δ εισέλθει στην x_{i_1}, τότε η ροή που θα εισέλθει στην x_{i_p} στο τόξο $(x_{i_p}, x_{i_{p+1}})$ του S είναι :

$$\delta_{i_p i_{p+1}} = \delta \cdot \prod_{(x_i, x_j) \in F_p} g_{ij} \cdot \prod_{(x_j, x_i) \in B_p} 1 / g_{ji} \equiv \delta \cdot (S_p) \qquad 2.4$$

όπου F_p και B_p αντιπροσωπεύουν τα "πρός τα μπρός" και "τα πρός τα πίσω" τόξα, αντίστοιχα, της υπο-αλυσίδας $S_p = x_{i_1}, x_{i_2}, \ldots\ldots, x_{i_p}$ και $g(S_p)$ είναι το κέρδος της S_p. Εάν $(x_{i_p}, x_{i_{p+1}})$ είναι ένα "προς τα μπρός" τόξο του S που μεταφέρει ροή $\xi^{\theta}_{i_p i_{p+1}}$ το τόξο θα κορεστή όταν $\xi^{\theta}_{i_p i_{p+1}} + \delta_{i_p i_{p+1}} = q_{i_p i_{p+1}}$. Εάν το τόξο είναι ένα "πρός τα πίσω" τόξο θα έχει την ροή του ίση με 0 όταν

$\delta_{i_p i_{p+1}} = \xi^{o}_{i_{p+1} i_p}$, όταν δηλαδή $\delta_{i_p i_{p+1}} = g_{i_{p+1} i_p} \cdot \xi^{\theta}_{i_{p+1} i_p}$. Η προσαυξημένη χωρητικότητα της αλυσίδας S δίνεται από :

$$
q(S) = \min \left[\min_{(x_{i_p}, x_{i_{p+1}}) \in F} \left(\frac{q_{i_p i_{p+1}} - \xi^{\theta}_{i_p i_{p+1}}}{g(S_p)} \right), \right.
$$

$$
\left. \min_{(x_{i_{p+1}}, x_{i_p}) \in B} \left(\frac{g_{i_{p+1} i_p} \cdot \xi^{\theta}_{i_{p+1} i_p}}{g(S_p)} \right) \right] \qquad 2.5
$$

Εάν η S δέν είναι μια απλή αλυσίδα και μερικά τόξα εμφανίζονται περισσότερο απο μιά φορά, τότε ο τύπος που δίνει την προσαυξημένη χωρητικότητα είναι παρόμοιος με τον 2.5.

2.2.2 Ενεργητικοί κύκλοι

Απο την στιγμή που ένας κύκλος μπορεί να αναφερθεί σαν μια αλυσίδα με ίδια αρχική και τελική κορυφή, τότε το κέρδος του κύκλου (και η χωρητικοτητά του) μπορεί να καθοριστεί με τον ίδιο τρόπο σαν κι αυτό μιας αλυσίδας. Ενας κύκλος Φ λέγεται *ενεργητικός* σε ένα γράφημα G(Ξ) με αναφορά την κορυφή t (αποδέκτης) εάν [3] :

(i) Το κέρδος του είναι μεγαλύτερο της μονάδας.

(ii) Η προσαυξημένη χωρητικότητα είναι διάφορη του μηδενός.

(iii) Υπάρχει κάποια κορυφή x_i στον Φ έτσι ώστε μια αυξητική αλυσίδα απο την x_i, στη t να υφίσταται.

Ομοίως μπορεί να καθοριστεί ένας ενεργητικός κύκλος με αναφορά στην αρχική κορυφή s.

Απο την παραπάνω διευκρίνηση είναι φανερό οτι εάν κάποια ροή κυκλοφορήση γύρω απο ένα ενεργητικό κύκλο, τότε κάποια παραπανήσια ροή μπορεί να δημιουργηθεί (φαίνεται απο την (i)) και αυτή η παραπανήσια ροή μπορεί να μεταφερθεί στην t (φαίνεται απο την (iii)).

Ενας ενεργητικός κύκλος με αναφορά την t λέγεται οτι είναι απο-ενεργοποιημένος εαν κάποια ροή επιβληθεί στο G έτσι ώστε, ή η προσαυξημένη χωρητικότητα του γίνει μηδέν ή οι χωρητικότητες όλων των αυξητικών αλυσίδων απο κάθε κορυφή του κύκλου στην t γίνουν μηδέν (δηλ. δεν υπάρχουν άλλες αυξητικές αλυσίδες στο καινούργιο G(Ξ)) [3].

2.3 Αλγόριθμος βέλτιστης ροής για γραφήματα με τόξα κέρδους.

Εδώ λοιπόν θα παρουσιάστει ο αλγόριθμος εύρεσης της μέγιστης ροής σε ένα γράφημα με τόξα κέρδους. Ο αλγόριθμος αυτός παρουσιάστηκε απο τον Νίκο Χριστοφίδη το 1975.

2.3.1 Παρουσίαση Αλγόριθμου Χριστοφίδη[3]

Βήμα 1. ΞΕΚΙΝΗΣΤΕ με κάποια εφικτή ροή Ξ στο γράφημα G. (Μηδενική ροή γιά όλα τα τόξα είναι αποδεκτή).

Βήμα 2. ΒΡΕΙΤΕ έναν ενεργητικό κύκλο μέσα στο G(Ξ), με αναφορά την κορυφη t.

Βήμα 3. ΕΣΤΩ x_i είναι η κορυφή του κύκλου Φ έτσι ώστε η αυξητική αλυσίδα είναι απο την x_i μέχρι την t(σημειώστε οτι η x_i μπορεί να είναι η ίδια η t). Ξεκινώντας απο την x_i κυκλοφορήστε κάποια ροή δ μέσα στον ενεργητικό κύκλο και μετά μεταβιβάστε την εξερχόμενη ροή του x_i κατά μήκος της αυξητικής αλυσίδας μέχρι την t. ΔΙΑΛΕΞΤΕ δ έτσι ώστε (μαζί με την ίδη υπάρχουσα ροή Ξ) ή η προσαυξημένη χωρητικότητα του Φ ή αυτή της αυξητικής αλυσίδας μηδενιστή.

Βήμα 4. ΕΝΗΜΕΡΩΣΤΕ την Ξ και επιστρέψτε στο βήμα 2 μέχρι που όλοι οι ενεργητικοί κύκλοι απο-ενεργοποιηθούν και δεν μπορούν να βρεθούν πλέον άλλοι ενεργητικοί κύκλοι, οπότε σε αυτή την περίπτωση προχωρήστε στο βήμα 5.

Βήμα 5. ΒΡΕΙΤΕ την αυξητική αλυσίδα απο την s στη t με το μεγαλύτερο κέρδος. Στείλτε κάποια ροή κατά μήκος της αλυσίδας μέχρι η προσαυξημένη χωρητικοτητά της μηδενιστή.

Βήμα 6. ΕΝΗΜΕΡΩΣΤΕ την Ξ και επιστρέψτε στο βήμα 5 μέχρι όπου ή η εξερχόμενη ροή απο την t είναι η απαιτούμενη τιμή της βέλτιστης ροής, ή μέχρι όπου δέν μπορεί να βρεθεί περαιτέρω άλλη αυξητική αλυσίδα οπότε σε αυτή την περίπτωση, η ροή είναι η μέγιστη-βέλτιστη ροή.
ΤΕΛΟΣ του αλγορίθμου.

Τα βήματα του παραπάνω αλγορίθμου είναι πολύ απλά και η μέθοδος είναι εφικτή. Τα βήματα 2 και 5 μπορούν, να εκτελεστούν απο ένα αλγόριθμο έυρεσης του ελαχίστου δρόμου και για αυτό το σκοπό θα χρησιμοποιηθεί ο ΓΕΝΙΚΟΣ αλγόριθμος. Ετσι λοιπόν, αντίστοιχα με το γράφημα $G(\Xi)$ μπορεί να καθοριστεί το προσαυξημένο γράφημα $G^{\mu}(\Xi) = (X^{\mu}, A^{\mu})$ με κορυφές $X^{\mu} = X$ και με τόξα A^{μ} έτσι ώστε [3]:

$$(x_i^{\mu}, x_j^{\mu}) \in A^{\mu} \qquad \text{εάν} \qquad 0 \le \xi_{ij}^{\theta} < q_{ij},$$

με προσαυξημένο "κόστος"

$$c_{ij}^{\mu} = -\log(q_{ij})$$

και χωρητικότητα

$$q_{ij}^{\mu} = q_{ij} - \xi_{ij}^{\theta}$$

και

$$(x_j^{\mu}, x_i^{\mu}) \in A^{\mu} \qquad \text{εάν} \qquad 0 < \xi_{ij}^{\theta} \le q_{ij},$$

με προσαυξημένο " κόστος "

$$c^{\mu}_{ij} = -\log (1/g_{ij})$$

και χωρητικότητα

$$q^{\mu}_{ij} = \xi^{\theta}_{ij} \cdot g_{ij} .$$

Ενας κύκλος Φ στο G(Ξ) με κέρδος μεγαλύτερο της μονάδας είναι αντίστοιχα ένα κύκλωμα στο $G^{\mu}(\Xi)$ με αρνητικό "κόστος". Αυτό μπορεί να φανεί παίρνοντας αλγορίθμους και στις δύο πλευρές της σχέσης 2.3 αντίστοιχα για ένα κύκλο Φ. Ετσι λοιπόν είναι [3] :

$$\log [g(\Phi)] = \sum_{(x_i, x_j) \in F} \log(g_{ij}) + \sum_{(x_j, x_i) \in B} \log(1/g_{ji})$$

$$= - \left[\sum_{F} c^{\mu}_{ij} + \sum_{B} c^{\mu}_{ij} \right]$$

όπου F και B είναι όπως έχει αναφερθεί το σύνολο των "πρός τα μπρός" και των "πρός τα πίσω" τόξων του κύκλου Φ.
Εάν $g(\Phi) > 1$ τότε $\log [g(\Phi)] > 0$ πράγμα που σημαίνει οτι

το κόστος του κύκλου Φ στο $G^{\mu}(\Xi)$, δηλ. το $\sum_{F} c^{\mu}_{ij} + \sum_{B} c^{\mu}_{ij}$

πρέπει να είναι αρνητικό. Επίσης, εάν $g(\Phi) \leq 1$ το κόστος του Φ στο G(Ξ) είναι τότε μη-αρνητικό.
Ετσι λοιπόν, κύκλοι στο G(Ξ) -στο βήμα 2 του παραπάνω αλγορίθμου -μπορούν να υπολογιστούν με το να βρεθούν όλα τα συντομότερα (ελάχιστου "κόστους") μονοπάτια απο τις κορυφές του $G^{\mu}(\Xi)$ μέχρι την τερματική κορυφή t χρησιμοποιώντας τον ΓΕΝΙΚΟ αλγόριθμο, και δοκιμάζοντας για κυκλώματα αρνητικού κόστους. Παρόλα αυτά, εάν η ροή δ που εφαρμόζεται στον ενεργητικό κύκλο Φ στο βήμα 3 και μετά μεταφέρεται κατά μήκος της αυξητικής αλυσίδας στη t δεν μηδενίζει την χωρητικότητα του Φ, αλλά μηδενίζει την χωρητικότητα της αυξητικής αλυσίδας, τότε στην επόμενη εκτέλεση του βήματος 2 είναι μόνο απαραίτητο να βρεθεί μια άλλη αυξητική αλυσίδα που ξεκινάει απο μια κορυφή του Φ μέχρι την t. Και συνεχίζοντας έτσι μέχρι ο κύκλος απο-ενεργοποιηθεί και κάποιοι άλλοι

ενεργητικοί κύκλοι μπορούν να υπολογιστούν απο το βήμα 2 [3].
Το βήμα 5 του παραπάνω αλγορίθμου μπορεί επείσης να υπολογιστεί χρησιμοποιώντας τον Γενικό αλγόριθμο και βρήσκοντας το συντομότερο μονοπάτι από την s μέχρι την t [3].

2.3.2 Παρουσίαση Γενικού Αλγορίθμου

Μέ X θα συμβολίζεται το σύνολο των κορυφών, με A το σύνολο των τόξων του γραφήματος G που περιέχονται στις διαδρομές με το μικρότερο βάρος [1].

Βήμα 0. ΠΡΟΣΔΙΟΡΙΣΤΕ ένα μερικό γράφημα του αρχικού γραφήματος που να είναι ένας δενδρισμός με ρίζα την s.
Για το σκοπό αυτό μπορεί να χρησιμοποιηθεί ο αλγόριθμος του DIJKSTRA.
Εάν s δέν είναι ρίζα, ΤΕΛΟΣ.
Εάν s είναι ρίζα του γραφήματος, έστω A το σύνολο των τόξων του δενδρισμού και
Π(x) = βάρος της διαδρομής απο την s στην x μέσα στο δενδρισμό G.
ΠΗΓΑΙΝΕ στο βήμα 1.

Βήμα 1. ΑΝΑΖΗΤΗΣΕ ένα τόξο u τέτοιο ώστε

$$\delta = -\Pi(T(u)) + \Pi(I(u)) + W_u < 0$$

όπου T(u) : το τέλος του τόξου u.
I(u) : η αρχή του τόξου u.
W_u : το κόστος του τόξου u.

Εάν ένα τέτοιο τόξο δέν υπάρχει, τότε G είναι ένας δενδρισμός με το μικρότερο κόστος. ΤΕΛΟΣ
Εάν ένα τέτοιο τόξο υπάρχει, τότε :
Εάν (X, A ∪ {u}) περιέχει ένα κύκλωμα τότε το κύκλωμα είναι αρνητικό. ΤΕΛΟΣ
Εάν (X, A ∪ {u}) δέν περιέχει κύκλωμα, τότε πήγαινε στο βήμα 2.

Βήμα 2. *Εστω* x = T(u), v τόξο του G τέτοιο ώστε T(v) = x

ΘΕΣΕ $A = A \cup \{u\} - \{v\}$

$X' = \{ y \in X \,/\, y = x$ ή η y είναι ένας απόγονος της x μέσα στο $G \}$.

$\Pi(y) = \Pi(y) + \delta \quad \forall y \in X'$

ΠΗΓΑΙΝΕ στο βήμα 1.

2.3.3 Παρουσίαση Αλγορίθμου Dijkstra.

Με S θα συμβολίζεται το σύνολο των κορυφών για τις οποίες έχουν ίδη υπολογιστεί οι διαδρομές με το μικρότερο μήκος. Σε κάθε επανάληψη το $\Pi(x)$ συμβολίζει το βάρος μιάς διαδρομής με το μικρότερο βάρος απο την s στην x μέσα στην κλάση των διαδρομών των οποίων όλες οι ενδιάμεσες κορυφές ανήκουν στο S. Σε κάθε επανάληψη η τελευταία κορυφή που έχει εισέλθει στο S παίζει ένα ιδιαίτερο ρόλο και θα συμβολίζεται με $\hat{x}$. Ομοίως κι εδώ όπως και στον Γενικο αλγόριθμο A είναι το σύνολο των τόξων με το μικρότερο βάρος, και W το βάρος κάθε τόξου [1].

Βήμα 0. ΘΕΣΕ $S = \{s\}$, $\Pi(s) = 0$, $\Pi(x) = \infty \quad \forall x \in X - \{s\}$

$A(x) = \emptyset \quad \forall x \in X, \quad \hat{x} = s$

Βήμα 1. ΕΞΕΤΑΣΕ διαδοχικά όλα τα τόξα u των οποίων το αρχικό άκρο είναι $\hat{x}$ και το τελικό άκρο y δεν ανήκει στο S.

Εαν $\Pi(\hat{x}) + Wu < \Pi(y)$ τότε

ΘΕΣΕ $\Pi(y) = \Pi(\hat{x}) + Wu$

$A(y) = u$

ΠΗΓΑΙΝΕ στο βήμα 2.

Βήμα 2. ΔΙΑΛΕΞΕ μια κορυφή $z \notin S$ τέτοια ώστε :

$$\Pi(z) = \min_{y \notin S} [\, \Pi(y) \,]$$

(σε περίπτωση ισοτιμίας γίνεται αυθαίρετη επιλογή)

Εάν $\Pi(z) = \infty$, η s δεν είναι ρίζα. ΤΕΛΟΣ

Εαν όχι : ΘΕΣΕ $\hat{x} = z$, $S = S \cup \langle \hat{x} \rangle$

Εαν $S = X$ ΤΕΛΟΣ

Εαν $S \neq X$ ΠΗΓΑΙΝΕ στο βήμα 1.

ΒΙΒΛΙΟΓΡΑΦΙΑ ΚΕΦΑΛΑΙΟΥ 2

ΕΛΛΗΝΙΚΗ

1. ΜΙΧΑΛΟΠΟΥΛΟΣ Μ., Θεωρία Ροών
Χανιά, 1989

ΞΕΝΗ

2. CHRISTOFIDES, N.,
MINGOZZI, A., Combinatorial Optimization

3. CHRISTOFIDES, N., Graph Theory
An Algorithmic Approach
Department of Managment Science
Imperial College
London, 1975

4. PHILLIGS, T.,
GARSIA-DIAZ, A., Fundamentals of Network Analysis

ΚΕΦΑΛΑΙΟ 3

ΠΡΟΣΕΓΓΙΣΗ ΤΗΣ ΛΕΙΤΟΥΡΓΙΑΣ ΤΟΥ ΣΥΝΑΛΛΑΓΜΑΤΟΣ ΜΕ ΤΗΝ ΘΕΩΡΙΑ ΤΩΝ ΡΟΩΝ

3.1 Εισαγωγή

Ξεκινώντας, θα πρέπει να τονιστεί ότι το κεφάλαιο αυτό είναι βασισμένο στο άρθρο των Νίκου Χριστοφίδη, Robin Hewins, Gerald Salkin απο το Department of Management Science του Imperial College στο Λονδίνο με τον τίτλο Graph Theoretic Approaches to Foreign Exchange Operations.

Η νομισματική συναλλαγή μέ σκοπό νά επιτευχθεί η καλύτερη ισοτιμία όπως αναφέρθηκε και στο κεφ. 1 είναι γνωστή σάν arbitrage ή πρόκριση συναλλάγματος. Το **arbitrage**, λοιπόν σύμφωνα και με την θεωρία του κεφ. 1 μπορεί να χωριστεί σε τρείς κατηγορίες :

(1) **Arbitrage με βάση το τόπο συναλλαγής** : δηλαδή η απόκτηση πλεονεκτήματος με την συναλλαγή , απο την διαφορά μεταξύ των ισοτιμιών που καθορίζονται την ιδια χρονική στιγμή σε διαφορετικές αγορές.

(2) **Arbitrage με βάση το χρόνο** : η απόκτηση πλεονεκτήματος με την συναλλαγή απο την διαφορά μεταξύ των διαφορετικών περιθωρίων κλεισίματος που υπαρχουν μέχρι την συναλλαγματική λήξη.

(3) **Arbitrage με βάση το επιτόκιο** : το πλεονέκτημα που μπορείς να αποκτήσεις απο την διαφορά της απόδοσης των προεξωφλητικών επιτοκίων (βραχυπρόθεσμων) τών διαφόρων νομισμάτων. Αυτή η μορφή του αρμπιτράζ μπορεί να διαχωρισθεί σε α) **καλυμένο αρμπιτράζ επιτοκιου** και **β) ακάλυπτο αρμπιτράζ επιτοκίου**. Η πρώτη μορφή χρησιμοποιεί τη σημερινή προθεσμιακή

αναλογία για την μετατροπή (προθεσμιακά) πίσω ξανά στο υπάρχων νόμισμα η δεύτερη μορφή επιτρέπει στον dealer να χρησιμοποιήσει την τρέχουσα αναλογία που θα υπάρχει στο μέλλον.

Οι μορφές (1) και (3α) είναι ντετερμινιστικές και με αυτές θα ασχοληθεί αυτή η εργασία οι άλλες μορφές είναι στοχαστικές και απαιτούν πρόβλεψη των μελλοντικών συναλλαγματικών αναλογιών για να καθοριστούν τα πιθανά αποτελέσματα.

Εδώ λοιπόν η μελέτη αυτή ασχολείται με τη τρέχουσα και τη προθεσμιακή αγορά και για τις μορφές arbitrage (1) και (3α). Οπως αναφέρθηκε και στην αρχή βασικος σκοπός της εργασίας είναι να επιτευχθεί η μέγιστη δυνατή ταχύτητα υπολογισμού τού αποτελέσματος. Τα προβλήματα που αντιμετωπίζονται εδώ μπορούν να λυθούν και με γραμμικό προγραμματισμό αλλά όμως η θεωρία των γραφημάτων είναι η πλέον κατάλληλη γιά να επιτευχθεί όσο το δυνατόν μικρότερο υπολογιστικό χρόνο.

Υποτίθεται οτι το σύνολο των νομισμάτων που συναλλάσονται δίνεται μαζί με τις επικρατούσες συναλλαγματικές ισοτιμίες απο κάθε νόμισμα για κάθε νόμισμα (και για τις δυο μορφές , το τρέχων και το προθεσμιακο).

Μη διαθέσιμες , αμφίβολες ή αζήτητες αναλογίες μπορεί να θεωρηθούν μηδέν . Για κάθε ζεύγος νομισμάτων (i,j) τα πάνω όρια (δηλ. το ανώτερο ποσό) του ποσού του νομίσματος i που μπορούν να ανταλλαγούν με το νόμισμα j παραδιδόμενα σε κάποια συγκεκριμένη ημερομηνία θεωρούνται γνωστά. Επίσης τα επιτόκια είναι καθορισμένα.

3.2 Προβλήματα arbitrage

Α. Arbitrage κατευθείαν απο τις ισοτιμίες

Α1. Ο τύπος Α1 εμπλέκει τα ποιό επικερδεί συναλλάγματα απο μια δωθούσα απλή νομισματική κατάσταση σε μια άλλη νομίσματική κατάσταση για την ίδια ημέρα παράδοσεις.

(α) Δωθέντος ενος συγκεκριμένου ποσού a_k νομίσματος k ,ποιο είναι το μέγιστο ποσό του νομίσματος i που μπορεί να αποκτηθεί;

(β) Δωθέντος ενος νομίσματος k, ποιο ποσό απο αυτό το νόμισμα μπορεί να μαζευτεί απο τη κυκλοφοριακή ροή χρημάτων μεταξύ των άλλων νομισμάτων;
Το πρόβλημα (β) είναι ενα καθαρό πρόβλημα **arbitrage**.

Α2. Ο τύπος Α2 ειναι μια επέκταση του τύπου Α1 :
(α) Δωθέντος ενος συγκεκριμένου ποσού a_k νομίσματος **k** για να πωληθεί, τι άλλα νομίσματα μπορούν να αγοραστούν ;
(β) Δωθέντος ενος συγκεκριμένου ποσού ενος νομίσματος **i** για να αγορασθεί, βρήτε τι ποσά απο άλλα νομίσματα θα πρέπει να πωληθούν.

Α3. Ο τύπος Α3 είναι ο γενικός τύπος Α προβλημάτων **arbitrage** :
(α) Δωθέντος κάποιου ποσού χρημάτων κρατημένα σε διάφορα νομίσματα βρήτε το βέλτιστο σύνολο των συναλλαγών για να βελτιωθεί το υπάρχων πακέτο.

Οι παραπάνω τύποι του **arbitrage** είναι πιθανοί και για τρέχων και για προθεσμιακό συνάλλαγμα.

Β. **Arbitrage** απο ανταλλαγή των ισοτιμιών

Ενώ όπως φαίνονται οι τύποι των προβλημάτων Α1-Α3 αναφέρονται στην αλλαγή ενός νομίσματος σε ένα άλλο ή σε πολλά άλλα, όλα όμως για παράδοση στην ίδια ημερομηνία, με τον τύπο Β του **arbitrage** μπορούν να καθοριστούν δύο συμφωνίες οι οποίες επιτρέπουν την ανταλλαγή του υπάρχων νομίσματος σε άλλα νομίσματα στη τρέχουσα αγορά και μετά την επιστροφή στο αρχικό πακέτο των νομισμάτων μετά απο μια καθορισμένη περίοδο και σε μια ισιτιμία που καθορίστηκε στο παρόν. Τέτοιες συμφωνίες ονομάζονται "**τρέχων** *εναντίων* **προθεσμίας**" ανταλλαγές και είναι ένα σπουδαίο μέρος του συστήματος εξωτερικού συναλλάγματος απο τη στιγμή τέτοιες συμφωνίες αναφέρονται σε **arbitrage** χρόνου και επιτοκίου μαζί.

Επίσης όπως και στο Α πρόβλημα υπάρχουν κι εδώ τύποι Β1, Β2, Β3 που είναι αντίστοιχοι με τα Α1, Α2, Α3 με τον αρχικό όμως όρο οτι όλες ο προθεσμιακές συμφωνίες που καθορίστηκαν θα συνεπάγουν και μια αντίστοιχη τρέχουσα συναλλαγή κατά την αντίστροφη κατεύθυνση.

Γ. Arbitrage Επιτοκίου.

Οι παραπάνω περιπτώσεις δεν ασχολούνται καθόλου με το επιτόκιο - αυτή είναι η γενική περίπτωση όπου το συνάλλαγμα εξαρτάται απο μια τράπεζα. Παρόλα αυτά όταν το εξωτερικό νόμισμα μπορεί να κερδίσει απο την απόδοση του επιτοκίου το πρόβλημα πλέον που αντιμετωπίζεται είναι να αποφασιστεί σε ποιά αγορά (δηλ. σε ποιό νόμισμα) να τοποθετηθεί το υπάρχων κεφαλαιό. Εδώ όπως αναφέρθηκε και στην αρχή τού κεφαλαίου θα γίνει αναφορά στο καλυμμένο arbitrage επιτοκίου, δηλ. όπου επιστρέφουμε στο αρχικό νόμισμα (ή πακέτο νομισμάτων) σε μια προθεσμιακή ισοτιμία, που καθορίζεται τώρα, μετά την περίοδο της τοποθέτησεί μας.

3.3 Διατύπωση της θεωρίας των Γραφημάτων

Ενα νόμισμα i αντιπροσωπεύται απο ένα σύνολο κορυφών x_{im} με $m = 0,1,\dots,n$, του γραφήματος G ,οπου n είναι ο αριθμός των προθεσμιακών επιτοκίων που υπάρχουν στην αγορά (π.χ. συνήθως οταν μιλάμε για 1,2,3,και 6 μήνες είναι $n = 4$). Μια πιθανή συναλλαγή απο το νόμισμα i την χρονική περίοδο p σε ενα νόμισμα j την χρονική περίοδο r αντιπροσωπεύεται απο ένα τόξο (x_{ip}, x_{jr}). Το γράφημα G θα παριστάνεται απο την τριάδα $\langle X, A, N \rangle$ οπου X είναι το σύνολο των κορυφών (νομίσματα) ,A είναι το σύνολο των τόξων (πιθανές συναλλαγές) , και N το σύνολο των προθεσμιακών ημερομηνιών με τις οποίες σχετίζεται η επένδυση.
Κάθε τόξο $(x_{ip}, x_{jr}) \in A$ συσχετίζεται με ενα κέρδος g_{ijpr} (εξαίρεση υπάρχει στο τύπο Γ του arbitrage οταν επιδέχεται ανταλλαγή τρέχων εναντίων προθεσμιακού) το οποίο παριστάνει:

(1) Την ισοτιμία μεταξύ του νομίσματος i και του νομίσματος

j για προθεσμιακή ημερομηνία p οταν r = p. Ετσι η τρέχουσα αναλογία του i σε σχέση με το j είναι q_{ijoo}.

(2) Την τιμή της μονάδος του νομίσματος i που προσφέρθηκε για την περίοδο r οταν i = j και p = 0.

Τα άλλα πιθανά τόξα είναι στοχαστικά ή χωρις σημασία.

Επιπρόσθετα κάθε τόξο (x_{ip}, x_{jr}) συσχετίζεται με μια χωρητικότητα q_{ijpr} που παριστά το μέγιστο δυνατό ποσό του νομίσματος i για μια τέτοια μετατροπή ή τοποθέτηση χωρίς να επηρεάζει τις συναλλαγματικές ισοτιμίες ή τα επιτόκια αντίστοιχα. Πιθανές συμφωνίες πέρα απο αυτά τα όρια μπορούν να περιληφθούν στη μεθολογία με λίγη δυσκολία.

Ορίζοντας σωστά την κορυφή πηγή s, και την κορυφή αποδέκτη t στο δίκτυο που περιγράφηκε ποιό πάνω, η χρήση του γενικού αλγόριθμου για ροές στα γραφήματα με τόξα κέρδους (δηλ. ο Αλγόριθμος του Χριστοφίδη) μπορεί να επιλύσει τα προβλήματα arbitrage που παρουσιάστηκαν.

3.3.1 Το μοντέλο του βασικού προβλήματος arbitrage Παράδειγμα πιλότος.

Εδώ λοιπόν θα παρουσιαστεί το μοντέλο του τύπου A1 καθώς θα δωθεί και ένα παράδειγμα για αυτού του είδους το πρόβλημα, το οποίο θα είναι και το βασικό παράδειγμα στην παρουσίαση του προγραμματός.

Το πρόβλημα (α) είναι επακριβώς το πρόβλημα τού να βρεθεί η βέλτιστη ροή με εισαγώμενη τιμή $v_s = a_k$ μεταξύ των κορυφών $s = x_{kp}$ και $t = x_{ip}$ του γραφήματος $G = \{X, A, N = p\}$ οπου οι κορυφές x_{kp} και x_{ip} αντιστοιχούν στη προθεσμιακή ημερομηνία p της κατάστασης των νομισματων k και i και ολες οι άλλες κορυφές $x_{lp} \in X$ του γραφήματος G αντιστοιχούν στα άλλα νομίσματα για αυτή την ημερομηνία.

Αυτή η βέλτιστη μορφή ροής απο το $x_{kp} = s$ στο $x_{ip} = t$ θα μπορεί να παρουσιάση την καλύτερη σειρά των συναλλαγών για να μετατρέπη το ποσό a_k του νομίσματος k στο μεγαλύτερο ποσό του νομίσματος i την προθεσμιακή ημερομηνία p.

Το πρόβλημα (β) είναι μια ειδική περίπτωση του (α) η κορυφή πηγής s είναι ιδια με την κορυφή τελους t π.χ. $s = t = x_{kp}$.

Ας διευκρινηστεί εδώ η έννοια του προβλήματος (β) δίνοντας ενα παράδειγμα:

ΑΓΟΡΑ

	DM	$	Fr.	£	Sw.Fr.	
DM	-	0.3765	1.8400	0.1640	1.1756	
$	2.5925	-	4.8237	0.4256	3.0460	
Fr .	0.5422	0.2068	-	0.08826	0.6310	ΠΩΛΗΣΗ
£	6.0775	2.3425	11.2600	-	7.1375	
Sw.Fr.	0.8485	0.3275	1.5740	0.1399	-	

Πίνακας 1: Cross-exchange rates

Εστω οπως φαίνεται και στο πίνακα 1 υπάρχουν τα εξης νομίσματα German Mark (DM) , US Dollar ($) , French Franc (Fr.) , Pound Sterling (£), και Swiss Franc (Sw.Fr.).Το τρέχων cross-exchange rate μεταξύ αυτών των νομισμάτων είναι το q_{ijoo} και δίνεται στο πίνακα 1.

Περαιτέρω q_{ijoo} θα είναι το μέγιστο ποσό του νομίσματος i που μπορεί να πουληθεί τρέχων για οποιοδήποτε άλλο νόμισμα j χωρίς να επειρεάζονται οι παραπάνω συναλλαγματικές ισοτιμίες και δίνεται απο : για i να αντιπροσωπεθει dollar- 5 million; sterling-1 million ; mark-8.133 million ; french franc-15 million ; swiss franc-10 million. Ετσι q_{ijoo} = 5 million ,j=2,...,5 κλπ.

Εδώ ζητήται να βρεθεί το μέγιστο ποσο απο δολάρια που μπορούν να παραχθούν απο τα υπάρχοντα νομίσματα μέσα στη τρέχουσα αγορά και τις βέλτιστες συναλλαγές που περιέχονται.

Εαν εφαρμοσθεί ο αλγόριθμος για τις ροές στα γραφήματα με τόξα κέρδους σε ενα 5-κορυφών γράφημα με τους παραπάνω συντελεστές τόξων, και τις δεδομένες χωρητικότητες και με $s = t = x_{20}$, το βέλτιστο σύνολο συναλλαγών περιέχει τις παρακάτω διακυμάνσεις, πρέπει να σημειωθεί ότι οι χωρητικότητες δύνονται σε εκατομμύρια :

Πουλώ	\$	2331000	Αγοράζω Fr	11246000
Πουλώ	Fr	11246000	Αγοράζω DM	6098000
Πουλώ	DM	6098000	Αγοράζω £	1000000
Πουλώ	£	1000000	Αγοράζω \$	2342500

ΚΑΘΑΡΕΣ ΕΙΣΡΟΕΣ ΣΕ \$	2342500
ΚΑΘΑΡΕΣ ΕΚΡΟΕΣ ΣΕ \$	2331000
ΚΑΘΑΡΟ ΚΕΡΔΟΣ ΣΕ \$	11500

ΚΕΦΑΛΑΙΟ 4

ΑΝΑΠΤΥΞΗ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ

4.1 Εισαγωγή

Στο κεφάλαιο αυτό θα γίνει μια λεπτομερή παρουσίαση και ανάπτυξη του κώδικα. Καταρχήν παρουσιάζονται οι μεταβλητές που χρησιμοποιούνται σε όλο το πρόγραμμα καθώς δίνεται και μια επεξήγηση για τον ρόλο της κάθε μίας. Στη συνέχεια παρουσιάζεται το βασικό πρόγραμμα και τα υπόλοιπα προγράμματα που χρησιμοποιούνται απο αυτό, και τέλος δίνονται οι διαδικασίες του κάθε προγράμματος και οι εργασίες που κάνει οι κάθε μία. Πρέπει να σημειωθεί εδώ ότι το πρόγραμμα έχει γραφτεί σε Turbo Pascal Version 5.5 .

4.2 Μεταβλητές του προγράμματος

Οπως ειπώθηκε και στην εισαγωγή του κεφαλαίου θα γίνει αναφορά και επεξήγηση των μεταβλητών που χρησιμοποιούνται στο κώδικα. Ετσι λοιπόν είναι :

inf : ακέραιος μεγαλύτερος απο το μεγαλύτερο αριθμό. έχει το ρόλο του άπειρου.

n : ακέραιος αριθμός, δείχνει των αριθμό των κορυφών του δικτύου.

s : ακέραιος αριθμός, δείχνει την κορυφή πηγή.

t : ακέραιος αριθμός, δείχνει την κορυφή αποδέκτη.

v : ακέραιος αριθμός, δείχνει την κορυφή που ενώνει το κύκλωμα με την αυξητική αλυσίδα. (στην περίπτωση όπου το κύκλωμα περιέχει την τερματική κορυφή t τότε είναι v = t).

d : πραγματικός αριθμός, χρησιμοποιείται δύο φορές, στην πρώτη περίπτωση χρησιμοποιήται στον Γενικό αλγόριθμο και δείχνει την διαφορά βάρους ανάμεσα σε δυό κορυφές, στη δεύτερη περίπτωση το d δείχνει την ροή που εισέρχεται σε μια κορυφή.

e : πραγματικός αριθμός, δείχνει την ελάχιστη τιμή ροής που πρέπει να κυκλοφορήσει στο κύκλωμα και στην αυξητική αλυσίδα ή μόνο στην αυξητική αλυσίδα (εάν δεν υπάρχει κύκλωμα).

path : λογική μεταβλητή, δείχνει εάν υπάρχει μονοπάτι απο την κορυφή s μέχρι την κορυφή t. Εάν υπάρχει παίρνει την τιμή TRUE.

circuit, cycle : λογικές μεταβλητές, δείχνουν εάν υπάρχει κύκλωμα αρνητικού κόστους, στην περίπτωση αυτή παίρνουν την τιμή TRUE.

dist : μονοδιάστατος πίνακας μεγέθους n απο πραγματικούς αριθμούς, δείχνει το βάρος κάθε κορυφής μετά τον Γενικό αλγόριθμο.

pred : μονοδιάστατος πίνακας μεγέθους n απο ακέραιους αριθμούς, δείχνει το δενδρισμό που βρέθηκε μετά τον Γενικό αλγόριθμο. Εάν μια κορυφή, έστω i, περιέχεται στον δενδρισμό τότε το pred[i] δείχνει την κορυφή απο όπου έρχεται το τόξο δηλαδή το τόξο είναι (pred[i],i), στην περίπτωση όπου η i δεν περιέχεται στον δενδρισμό τότε pred[i] = -1.

circ : μονοδιάστατος πίνακας μεγέθους n απο ακέραιους αριθμούς, δείχνει το κύκλωμα αρνητικού κόστους που έχει βρεθεί. Για τις κορυφές που δεν ανήκουν στο κύκλωμα το circ παίρνει την τιμή -1, διαφορετικά κι εδώ το circ[i] είναι η κορυφή πού ξεκινάει το τόξο και καταλήγει στην κορυφή i.

chain : μονοδιάστατος πίνακας μεγέθους n απο ακέραιους αριθμούς, δείχνει την αυξητική αλυσίδα απο τό κύκλωμα μέχρι την t, ή απο την s μέχρι την t (εάν δεν υπάρχει κύκλωμα), όπως και παραπάνω εάν μια κορυφή δεν ανήκει στην αυξητική αλυσίδα το chain παίρνει την τιμή -1, επίσης κι εδώ το chain[i] είναι η κορυφή που ξεκινάει το τόξο και καταλήγει στην κορυφή i.

Scirc : μονοδιάστατος πίνακας μεγέθους n απο πραγματικούς αριθμούς, δείχνει το κέρδος κάθε κορυφής του κυκλώματος εάν μια ροή δ εισέλθει στην κορυφή v.

Schain : μονοδιάστατος πίνακας μεγέθους n απο πραγματικούς αριθμούς, δείχνει το κέρδος κάθε κορυφής της αυξητικής αλυσίδας εάν μια ροή εξέλθει απο την κοινή κορυφή με το κύκλωμα, v, ή εάν μια ροή δ εισέλθει στην κορυφή πηγή s (όταν δεν υπάρχει κύκλωμα).

g : δισδιάστατος πίνακας μεγέθους n x n απο πραγματικούς αριθμούς, δείχνει τον συντελεστή κέρδους κάθε τόξου, εάν δεν υπάρχει το τόξο τότε το αντίστοιχο g παίρνει την τιμή inf.

q : δισδιάστατος πίνακας μεγέθους n x n απο πραγματικούς αριθμούς, δείχνει την χωρητικότητα κάθε τόξου, εάν δεν υπάρχει το τόξο τότε το αντίστοιχο q παίρνει την τιμή inf.

F : δισδιάστατος πίνακας μεγέθους n x n απο πραγματικούς αριθμούς, δείχνει την ροή που εισέρχεται σε κάθε τόξο (δηλ. το ξ^{e}_{ij}) προφανώς η ροή που εξέρχεται απο το τόξο (i, j) είναι F[i,j] x g[i,j], εάν δεν υπάρχει τόξο τότε το αντίστοιχο F παίρνει την τιμή inf.

c : δισδιάστατος πίνακας μεγέθους n x n απο πραγματικούς αριθμούς, δείχνει το προσαυξημένο κόστος στο γράφημα G^{μ}, επίσης κι εδώ εάν δεν υπάρχει τόξο το αντίστοιχο c παίρνει την τιμή inf.

qm : δισδιάστατος πίνακας μεγέθους n x n απο πραγματικούς αριθμούς, δείχνει την προσαυξημένη χωρητικότητα στο γράφημα G^{μ}, ομοίως εάν δεν υπάρχει τόξο το αντίστοιχο qm είναι inf.

4.3 Ανάπτυξη του προγράμματος

Το κυρίως πρόγραμμα ονομάζεται OSFP απο εκεί και μετά υπάρχουν άλλα τρία προγράμματα το INDATA, το FLOW και το GENERAL που το καθένα περιέχει διαφορετικές procedures και καλούνται απο το OSFP με την εντολή μεταγλωτιστή include ({$ I όνομα προγράμματος}). Απο το OSFP επίσης, γίνεται η έξοδος των αποτελεσμάτων, καθώς και δίνονται και οι πρώτες τιμές στον πίνακα F.

Το INDATA περιέχει την procedure InputData με την οποία γίνεται η είσοδος των δεδομένων. Τα δεδομένα που εισάγωνται είναι το inf, το n, το s, το t, το g, και το q.

Το General περιέχει τις διαδικασίες General και Dijkstra. Η procedure Dijkstra χρησιμοποιήται απο την General και είναι ο αλγόριθμος του Dijkstra ο οποίος επεξηγήται στο κεφάλαιο 2 τα input δεδομένα είναι τα n, s, t, inf, c και τα output είναι τα path, dist, pred. Η διαδικασία General είναι ο Γενικός αλγόριθμος και όπως έχει αναφερθεί, με αυτόν γίνεται ανίχνευση για αρνητικού κόστους κύκλωμα εάν κάτι τέτοιο υπάρχει τότε δίνεται, καθώς επίσης δίνεται και ο βέλτιστος δενδρισμός του δικτύου. Τα input της General είναι τα n, pred, dist, path και δίνει σαν output τα circ και cycle.

Το FLOW περιέχει τις procedures IncrGraph, AugChain, IncrCapacity, και UpdateFlow.

Η procedure IncrGraph δίνει το προσαυξημένο γράφημα G^{μ} πέρνει σαν input τα n, F, q, g και δίνει σαν oputput τα c και qm.

Η procedure AugChain βρίσκει την αυξητική αλυσίδα απο το αρνητικό κύκλωμα μέχρι την τερματική κορυφή t ή απο την κορυφή πηγή s μέχρι την t (όταν δεν υπάρχει κύκλωμα). Τα input είναι τα n, s, t, pred, circ, cycle, και δίνει σαν output τα chain και v.

Η procedure IncrCapacity βρίσκει την ελάχιστη ροή δ που πρέπει να κυκλοφορήσει στο κύκλωμα και στην αλυσίδα ή στην περίπτωση που δεν υπάρχει κύκλωμα βρίσκει την ελάχιστη ροή δ που πρέπει να κυκλοφορήσει στην αυξητική αλυσίδα απο την s στην t. Σαν input παίρνει τα inf, n, t, v, circ, chain, qm, g και cycle και σαν output δίνει τα e, Scirc και Schain.

Τέλος με την procedure UpdateFlow γίνεται ενημέρωση του πίνακα της ροής F στο τέλος κάθε επανάληψης, τα input αυτής της διαδικασίας είναι τα inf, n, g, F, circ, chain, Scirc, Schain και e, και δίνει output τον ενημερωμένο πίνακα ροής F.

Τελευταίο αυτού του κεφαλαίου είναι η παρουσίαση της λίστας του προγράμματος η οποία δίνεται στο παράρτημα Α, στο τέλος της μελέτης, επίσης στο παράρτημα Β, παρατίθεται η λίστα του προγράμματος Gener το όποιο είναι το πρόγραμμα του Γενικού αλγόριθμου και στο παράρτημα Γ παρατίθεται η λίστα του προγράμματος Dijkstra που είναι το πρόγραμμα του αλγόριθμου του Dijkstra, στο παράρτημα Δ παρουσιάζονται οι οθόνες εισαγωγής των στοιχείων και εξαγωγής των αποτελεσμάτων του κυρίως προγράμματος osfp, τέλος στο παράρτημα Ε παρουσιάζεται το λογικό διάγραμμα του osfp.

ΚΕΦΑΛΑΙΟ 5

ΣΥΜΠΕΡΑΣΜΑΤΙΚΕΣ ΕΠΙΣΗΜΑΝΣΕΙΣ

Σ' αυτή τη μελέτη έγινε όπως λέει και ο τίτλος της, μια προσέγγιση της βασικής λειτουργίας του συναλλάγματος, δηλαδή του arbitrage, με την θεωρία των ροών και ειδικότερα με τις ροές με τόξα κέρδους.

Οπως αναφέρθηκε λοιπόν, και στην αρχή της μελέτης, οι διεθνείς συναλλαγές διέπονται απο μια διαδικασία κανονισμών τους οποίους καθορίζει η Αγορά Συναλλάγματος. Η Αγορά Συναλλάγματος επιτελεί, διαμέσου των τμημάτων της Αγοράς Οψης και της Προθεσμιακής Αγοράς, τρείς βασικές λειτουργίες, πρώτα επιτυγχάνει τη μεταφορά αγοραστικής αξίας σε ξένο νόμισμα, δεύτερη λειτουργία είναι η παροχή πιστώσεων και τελευταία είναι η διασφάλιση απέναντι στους κινδύνους διακύμανσης των ισοτιμιών.

Το βασικό όμως αντικείμενο της μελέτης είναι η διενέργεια του arbitrage δηλαδή η ταυτόχρονη αγορά και πώληση διαφόρων νομισμάτων στην περίπτωση που υπάρχουν διαφορές στις ισοτιμίες τους. Βασικός σκοπός αυτού που διενεργεί το arbitrage και επομένως και αυτής της μελέτης είναι η πραγματοποίηση κέρδους, επειδή όμως υπάρχουν πολλοί που προσπαθούν ταυτόχρονα να επιτύχουν κι αυτοί κέρδος βασικός παράγοντας είναι ο χρόνος που παίρνονται οι αποφάσεις, και αυτός πρέπει να είναι όσο το δυνατόν μικρότερος. Ετσι λοιπόν, τα δύο βασικά σημεία της μελέτης ήταν η δημιουργία ενός προγράμματος που να δίνει κέρδος με το μικρότερο υπολογιστικό χρόνο.

Για το λόγο αυτό επιλέχθηκαν οι ροές με τόξα κέρδους του Νικου Χριστοφίδη, σαν τον αλγόριθμο επίλυσης του συγκεκριμένου προβλήματος, όπως αναφέρθηκε και στην αρχή το συγκεκριμένο πρόβλημα μπορούσε να επιλυθεί και με γραμμικό προγραμματισμό έχοντας όμως πολύ μεγαλύτερο υπολογιστικό χρόνο.

Στο σημείο αυτό πρέπει να τονιστεί η μεγάλη χρήση που έχουν σήμερα τά δίκτυα και αξίζει να σημειωθεί ενα παράδειγμα που δείχνει τη χρήση του συγκεκριμένου αλγορίθμου σε ενα διαφορετικό πρόβλημα.

Εστω λοιπόν ένας οικονομικός αναλυτής μιάς επιχείρησης που πρέπει να αποφασίσει πως θα διανείμει τα επενδυτικά ποσά της επιχείρησης μεταξύ συναγωνιζόμενων επενδύσεων. Ο αναλυτής λοιπόν μπορεί κάθε πιθανή επένδυση να την θεωρήση σαν ένα τόξο που ξεκινάει απο μια κορυφή που αντιστοιχεί με την αρχική ημερομηνία και καταλήγη στην ημερομηνία λήξης της επένδυσης. Εαν λοιπόν η επένδυση έχει απόδοση 8% τότε ένας συντελεστής κέρδους 1,08 αντιστοιχεί στο συγκεκριμένο τόξο. Επείσης η χωρητικότητα του τόξου θα είναι ίση με με το μέγιστο ποσό που μπορεί να τοποθετηθεί στην συγκεκριμένη επένδυση. Εαν κάθε π.χ. δολάριο που διατείθεται να επενδυθή μπορεί να θεωρηθεί σαν μια μονάδα ροής τότε το πρόβλημα απόφασης του αναλυτή για τις βέλτιστες επενδύσης μπορέι να θεωρηθεί σαν ένα πρόβλημα του να πάρθει όσο είναι δυνατόν μεγαλύτερη εξερχόμενη ροή απο μια δωθέντα κορυφή αποδέκτη με όσο το δυνατόν μικρότερη εισερχόμενη ροή στη κορυφή πηγή σε ένα αναπαραγώμενο γράφημα απο της πιθανές επενδύσης. (Η πηγή είναι η κορυφή που αντιπροσωπεύει την ημερομηνία έναρξης των επενδύσεων και ο αποδέκτης είναι η κορυφή που αντιπροσωπεύει την ημερομηνία που όλες οι επενδύσεις πρέπει να λήξουν).

Το βασικό πρόγραμμα λοιπόν είναι το **osfp** και εκτός αυτού δώθηκαν δύο ακόμα σημαντικά προγράμματα το **Gener** (ο Γενικός αλγόριθμος) και το **Dijkstra** (ο αλγόριθμος του **Dijkstra**).

Τέλος δώθηκε ένα αρκετά διευκρινηστικό παράδειγμα με κάποιες πιθανές συναλλαγματικές ισοτιμίες.

ΠΑΡΑΡΤΗΜΑ Α

```
program  Osfp;
  uses  Crt;

  type  arrnn         = array[1..20,1..20] of real;
        arrn1         = array[1..20] of integer;
        arrn2         = array[1..20] of real;
        boolarrn      = array[1..20] of boolean;
  var   c,
        g,
        q,
        f,
        qm                     : arrnn;
  var   path,
        cycle                  : boolean;
  var   Final                  : Boolarrn;
  var   dist,
        Schain,
        Scirc                  : arrn2;
  var   pred,
        circ,
        chain                  : arrn1;
  var   e,
        d,
        z,
        x                      : real;
  var   i,
        j,
        inf,
        s,
        t,
        n,
        v                      : integer;

{$I C:\LN\TPC\DIPLOMA\FLOW.PAS}
{$I C:\LN\TPC\DIPLOMA\GENERAL.PAS}
{$I C:\LN\TPC\DIPLOMA\INDATA.PAS}
```

```
begin
  clrscr;
  InputData;
  for i := 1 to n do
    for j := 1 to n do
    begin
      if (g[i,j] < inf) then
        F[i,j] := 0;
      if (g[i,j] >= inf) then
        F[i,j] := inf;
    end;
  repeat
    z := 0;
    for i := 1 to n do
    begin
      if (F[i,t] < inf) then
        z := z + F[i,t] * g[i,t];
    end;
    IncrGraph(n,inf,c,qm,F,q,g);
    Dijkstra(inf,s,t,n,c,path,final,dist,pred);
    General(n,pred,path,dist,circ,cycle);
    if (path = true) then
    begin
      Augchain(n,s,t,pred,circ,cycle,chain,v);
      IncrCapacity(n,inf,t,v,circ,chain,qm,g,cycle,e,Schain,Scirc);
      UpdateFlow(n,inf,g,F,circ,chain,Scirc,Schain,e);
      x := 0;
      for i := 1 to n do
      begin
        if (F[i,t] < inf) then
          x := x + F[i,t] * g[i,t];
      end;
    end;
  until((path = FALSE) or (z = x));
  clrscr;
  for i := 1 to n do
    for j := 1 to n do
```

```
    if ((F[i,j] < inf) and (F[i,j] > 0)) then
    begin
      write('Πουλήστε ');
      highvideo;
      write(F[i,j]:8:3);
      lowvideo;
      write(' του νομίσματος ');
      highvideo;
      write(i);
      lowvideo;
      write(' αγοράστε ');
      highvideo;
      write((F[i,j] * g[i,j]):8:3);
      lowvideo;
      write(' του νομίσματος ');
      highvideo;
      writeln(j);
      lowvideo;
      writeln;
      readln;
    end;
  end;
end.
```

```
program InData;

procedure InputData;      (**** ΕΙΣΟΔΟΣ ΔΕΔΟΜΕΝΩΝ******)

  Var  i,j      :Integer;
  var  inchr    :char;

begin
  write('Δώστε ένα πολύ μεγάλο αριθμό (θα αναγνωρίζεται σαν άπειρο : ');
  readln(inf);
  writeln;
  write('Δώστε τον αριθμό των νομισμάτων (όχι μεγαλύτερο του 20) : ');
  readln(n);
  writeln;
  write(' Δώστε το αρχικό νόμισμα : ');
  readln(s);
  writeln;
  write(' Δώστε το τελικό νόμισμα : ');
  readln(t);
  writeln;
  for i := 1 to n do
  begin
    for j := 1 to n do
      g[i,j] := inf;
  end;
  writeln;
  repeat
    writeln('Θέλεται να κάνεται διορθώσεις στον πίνακα g; y/n : ');
    inchr := readkey;
    if ( inchr = 'y') then
    begin
      i := 0;
      j := 0;
      write('Δώστε την αρχική κορυφή του τόξου : ');
      readln(i);
      write('Δώστε την τελική κορυφή του τόξου : ');
      readln(j);
      write('g[',i,j,'] = ');
```

```
      end;
    until (inchr = 'n');
    writeln;
    for i := 1 to n do
    begin
      for j := 1 to n do
        q[i,j] := 0;
    end;
    writeln;
    repeat
      writeln('Θέλεται να κάνεται διωρθώσεις στον πίνακα q; y/n : ');
      inchr := readkey;
      if ( inchr = 'y') then
      begin
        i := 0;
        j := 0;
        write('Δώστε την αρχική κορυφή του τόξου : ');
        readln(i);
        write('Δώστε την τελική κορυφή του τόξου : ');
        readln(j);
        write('q[',i,j,'] = ');
        readln(q[i,j]);
      end;
    until (inchr = 'n');
    writeln;
end;
```

```
program General;

Procedure Dijkstra(                                  (*** ΑΛΓΟΡΙΘΜΟΣ DIJKSTRA ***)
                    inf,s,t,n           : integer;
                    var c               : arrnn;
                    var path            : boolean;
                    var final           : boolarrn;
                    var dist            : arrn2;
                    var pred            : arrn1    );

var u,v,y,recent       : integer;
var newlabel, temp     : real;

begin
  for v := 1 to n do
  begin
    dist[v]  := inf;
    final[v] := false;
    pred[v]  := -1;
  end;
  dist[s]  := 0;
  final[s] := true;
  path     := true;
  recent   := s;
  while not final[t] do
  begin
    for v := 1 to n do
    if (c[recent,v] < inf) and (not final[v]) then
    begin
      newlabel := dist[recent] + c[recent,v];
      if newlabel < dist[v] then
      begin
        dist[v] := newlabel;
        pred[v] := recent;
      end;
    end;
    temp := inf;
    for u := 1  to n do
```

```
    if (not final[u]) and (dist[u] < temp) then
    begin
      y    := u;
      temp := dist[u];
    end;
    if temp < inf then
    begin
      final[y] := true;
      recent   := y;
    end
    else
    begin
      path     := false;
      final[t] := true;
    end;
  end;   (*  while    *)
  if (s = t) then
  pred[s] := pred[t];
end;   (*  DIJKSTRA  *)


Procedure General(                              (***** ΓΕΝΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ *****)
                          n          :Integer;
                     var  pred       :arrn1;
                     var  path       :Boolean;
                     var  dist       :arrn2;
                     var  circ       :arrn1;
                     var  cycle      :Boolean   );

var    circuit             : boolean;
Var    i,
       j,
       k,
       l                   :Integer;


begin
```

```
if path = TRUE then
begin
  repeat
    for i := 1 to n do
    circ[i] := -1;
    circuit := TRUE;
    d       := 0;
    i       := 1;
    j       := 0;
    repeat
      if (j < n) then
        j := j + 1
      else
      if (i < n) then
      begin
        j := 1;
        i := i + 1;
      end;
      d := dist[i] + c[i,j] - dist[j];
    until ((d < 0) or ((i=n) and (j=n)));
    if ( d < 0 ) then
    begin
      circ[j] := i;
      l       := i;
      repeat
        k       := pred[l];
        circ[l] := k;
        l       := k;
      until((k=j) or (k<0));
      if (k = j) then
      begin
        circuit := circuit;                    { έχουμε δηλ. κύκλωμα }
        cycle   := TRUE;
      end
      else
      begin
        circuit := not circuit;              { δεν έχουμε δηλ. κύκλωμα }
        cycle   := FALSE;
```

```
        end;
        if (circuit = FALSE) then
        begin
          pred[j] := i;
          dist[j] := dist[j] + d;
          i := 0;
          repeat
            i := i + 1;
            if (pred[i] = j) then
              dist[i] := dist[i] + d;
          until ( i = n );
        end;
      end;
    until ( circuit = TRUE );
  end;
end;  (***** GENERAL *****)
```

```
program Flow;

procedure IncrGraph                               (**** ΠΡΟΣΑΥΞΗΜΕΝΟ ΓΡΑΦΗΜΑ Gμ ******)
                 (        n,inf   :integer;
                    var   c       :arrnn;
                    var   qm      :arrnn;
                    var   F       :arrnn;
                    var   q       :arrnn;
                    var   g       :arrnn );

var i,j     : integer;

begin
  for i := 1 to n do
    for j := 1 to n do
    begin
      c[i,j]  := inf;
      qm[i,j] := inf;
    end;
    for i := 1 to n do
      for j := 1 to n do
      begin
        if (F[i,j] < inf) then
        begin
          if ((F[i,j] >= 0) and (F[i,j] < q[i,j])) then
          begin
            c[i,j]  := -(ln(g[i,j])/ln(10));
            qm[i,j] := q[i,j] - F[i,j];
          end;
          if (F[i,j] = q[i,j]) then
          begin
            c[j,i]  := -(ln(1/g[i,j])/ln(10));
            qm[j,i] := F[i,j] * g[i,j];
          end;
```

```
      end;
    end;
end;

procedure AugChain(                    { Βρίσκει την ΑΥΞΗΤΙΚΗ ΑΛΥΣΙΔΑ }
                  n,s,t           :Integer;
                  Var  pred       :arrn1;
                  Var  circ       :arrn1;
                  Var  cycle      :Boolean;
                  Var  Chain      :arrn1;
                  Var  V          :Integer    );

Var  i,j,k,l        :Integer;


begin
  for i := 1 to n do
  chain[i] := -1;
  if (cycle = TRUE) then
  begin
    i := 1;
    repeat
      k := circ[i];
      i := i + 1;
    until ((k=t) or (i=n+1));
    if (k = t) then
      v := t
    else
    begin
      v := t;
      repeat
        i := 1;
        k := pred[v];
        l := circ[i];
        chain[v] := k;
        repeat
          i := i + 1;
          l := circ[i];
```

```
        until ((l = k) or (i = n));
        v := k;
      until (l = k);
    end;
  end;
  if (cycle = FALSE) then
  begin
    i := t;
    repeat
      k := pred[i];
      chain[i] := k;
      i := k;
    until (k = s);
    v := s;
  end;
end;


procedure IncrCapacity(                    { Βρίσκει την ΕΛΑΧΙΣΤΗ ΡΟΗ δ }
                       n,inf,t        :Integer;
                       Var  v         :Integer;
                       Var  circ      :arrn1;
                       Var  chain     :arrn1;
                       Var  qm        :arrnn;
                       var  g         :arrnn;
                       var  cycle     :boolean;
                       var  e         :real;
                       var  Schain    :arrn2;
                       var  Scirc     :arrn2   );

var  d      : real;
var  i,
     j,
     l,
     k      : integer;
```

```
begin
  for i := 1 to n do
    Schain[i] := -1;
  for i := 1 to n do
    Scirc[i] := -1;
  if (cycle = TRUE) then
  begin
    Scirc[v] := 1;
    l := v;
    repeat           (**** Φτιάχνω πρώτα το κέρδος του κύκλου ****)
      i := 0;
      repeat
        i := i + 1;
        k := circ[i];
      until(k = l);
      if (g[l,i] = inf) then
        g[l,i] := 1/g[i,l];
      Scirc[i] := Scirc[l] * g[l,i];
      l := i;
    until(i = circ[v]);
    if (v <> t) then
    begin
      if (g[circ[v],v] = inf) then
        g[circ[v],v] := 1/g[v,circ[v]];
      Schain[v] := Scirc[i] * g[circ[v],v] - Scirc[v];
      l := v;
      repeat           (**** Εδώ παίρνω το κέρδος της αλυσίδας *****)
        i := 0;
        repeat
          i := i + 1;
          k := chain[i];
        until(k = l);
        if (g[l,i] = inf) then
          g[l,i]  := 1/g[i,l];
        Schain[i] := Schain[l] * g[l,i];
        l := i;
      until(i = t);
```

```
      end;
      d := 0;
      e := inf;
      l := v;
      repeat
        i := 0;
        repeat
          i := i + 1;
          k := circ[i];
        until(k = l);
        d := qm[k,i]/Scirc[k];
        if (d < e) then
          e := d;
        l := i;
     until(i = v);
     if ( v <> t ) then
     begin
       repeat
         i := 0;
         repeat
           i := i +1;
           k := chain[i];
         until(k = l);
         d := qm[k,i]/Schain[k];
         if (d < e) then
           e := d;
         l := i;
       until(i = t);
     end;
   end;
   if (cycle = FALSE) then
   begin
     Schain[v] := 1;
     l := v;
     repeat
       i := 0;
       repeat
         i := i + 1;
         k := chain[i];
```

```
        until(k = 1);
        if (g[l,i] = inf) then
          g[l,i] := 1/g[i,l];
        Schain[i] := Schain[l] * g[l,i];
        l := i;
      until(i = t);
      d := 0;
      e := inf;
      l := v;
      repeat
        i := 0;
        repeat
          i := i + 1;
          k := chain[i];
        until(k = l);
        d :=qm[k,i]/Schain[k];
        if (d < e) then
          e := d;
        l := i;
      until(i = t);
    end;
  end;


procedure UpdateFlow(                (**** ΕΝΗΜΕΡΩΣΗ ΤΗΣ ΡΟΗΣ F ********)
                      n,inf      : integer;
                      var  g     : arrnn;
                      var  F     : arrnn;
                      var  circ  : arrn1;
                      var  chain : arrn1;
                      var  Scirc : arrn2;
                      var  Schain: arrn2;
                      var  e     : real     );

Var  i,j          :integer;


begin
  for i := 1 to n do
```

```
    begin
      if (circ[i] > 0) then
      begin
        if (F[circ[i],i] >= inf) then
          F[i,circ[i]] := F[i,circ[i]] - (e * Scirc[circ[i]]);
        if (F[circ[i],i] < inf) then
          F[circ[i],i] := F[circ[i],i] + (e * Scirc[circ[i]]);
      end;
    end;
    for i := 1 to n do
    begin
      if (chain[i] > 0) then
      begin
        if (F[chain[i],i] >= inf) then
          F[i,chain[i]] := F[i,chain[i]] - (e * Schain[chain[i]]);
        if (F[chain[i],i] < inf) then
          F[chain[i],i] := F[chain[i],i] + (e * Schain[chain[i]]);
      end;
    end;
  end;
```

ΠΑΡΑΡΤΗΜΑ Β

```
program Gener;
  uses Crt;
  type      arrnn     = array[1..20,1..20] of real;
            arrn1     = array[1..20] of integer;
            arrn2     = array[1..20] of real;
            boolarrn = array[1..20] of boolean;
  var       inf,
             s,
             t,
             n,
             i,
             j,
             k,
             l,
             u,
             v,
             y,
             recent    : integer;
  var       newlabel,
            temp,
            d          : real;
  var       w          : arrnn;
  var       pred,
            circ       : arrn1;
  var       dist       : arrn2;
  var       final      : boolarrn;
  var       path,
            cycle,
            circuit    : boolean;
  var       inchr      : char;

Procedure Dijkstra(                          (*** ΑΛΓΟΡΙΘΜΟΣ  DIJKSTRA ***)
            inf,s,t,n       : integer;
            var w           : arrnn;
            var path        : boolean;
            var final       : boolarrn;
            var dist        : arrn2;
            var pred        : arrn1 );
```

```
begin
  for v := 1 to n do
  begin
    dist[v]  := inf;
    final[v] := false;
    pred[v]  := -1;
  end;
  dist[s]  := 0;
  final[s] := true;
  path     := true;
  recent   := s;
  while not final[t] do
  begin
    for v := 1 to n do
    if (w[recent,v] < inf) and (not final[v]) then
    begin
      newlabel := dist[recent] + w[recent,v];
      if newlabel < dist[v] then
      begin
        dist[v] := newlabel;
        pred[v] := recent;
      end;
    end;
    temp := inf;
    for u := 1  to n do
    if (not final[u]) and (dist[u] < temp) then
    begin
      y    := u;
      temp := dist[u];
    end;
    if temp < inf then
    begin
      final[y] := true;
      recent   := y;
    end
    else
    begin
```

```
      path     := false;
      final[t] := true;
    end;
  end;   (*  while    *)
end;   (*  DIJKSTRA  *)



Procedure General;                              (***** ΓΕΝΙΚΟΣ ΑΛΓΟΡΙΘΜΟΣ ****)
begin
  Dijkstra(inf,s,t,n,w,path,final,dist,pred);
  if path = TRUE then
  begin
    repeat
      for i := 1 to n do
      circ[i] := -1;
      circuit := TRUE;
      d       := 0;
      i       := 1;
      j       := 0;
      repeat
        if (j < n) then
          j := j + 1
        else
        if (i < n) then
          begin
            j := 1;
            i := i + 1;
          end;
        d := dist[i] + w[i,j] - dist[j];
      until ((d < 0) or ((i=n) and (j=n)));
      if ( d < 0 ) then
        begin
          circ[j] := i;
          l       := i;
          repeat
            k       := pred[l];
            circ[l] := k;
```

```
          l        := k;
        until((k=j) or (k<0));
        if (k = j) then
        begin
          circuit := circuit;              { έχουμε  δηλ.  κύκλωμα }
          cycle   := TRUE;
        end
        else
        begin
          circuit := not circuit;         { δεν έχουμε δηλ. κύκλωμα }
          cycle   := FALSE;
        end;
        if (circuit = FALSE) then
        begin
          pred[j] := i;
          dist[j] := dist[j] + d;
          i := 0;
          repeat
            i := i + 1;
            if (pred[i] = j) then
              dist[i] := dist[i] + d;
          until ( i = n );
        end;
      end;
    until ( circuit = TRUE );
  end;
end;  (***** GENERAL *****)

begin              (**** Κυρίως  πρόγραμμα  ****)
  clrscr;
  writeln('Δώστε ενα αριθμό ετσι ώστε να είναι πάντα ο μεγαλύτερος');
  write(' (Θα αναγνωρίζεται σαν άπειρο) : ');
  readln(inf);
  writeln;
  write('Δώστε τον αριθμό των κορυφών (όχι μεγαλύτερο του 20) : ');
  readln(n);
  writeln;
  write('Δώστε την κορυφή πηγή : ');
```

```
readln(s);
writeln;
write('Δώστε την κορυφή αποδέκτη : ');
readln(t);
writeln;
for i := 1 to n do
  for j := 1 to n do
    w[i,j] := inf;
repeat
  write('Θέλεται να δώσεται τιμή ή ');
  writeln('να κάνεται διόρθωση στον πίνακα των βαρών; y/n ');
  inchr := readkey;
  if (inchr = 'y') then
  begin
    write('Δώστε την αρχική κορυφη του τόξου : ');
    readln(i);
    write('Δώστε την τελική κορυφή του τόξου : ');
    readln(j);
    write('w[',i,j,'] : ');
    readln(w[i,j]);
  end;
until (inchr = 'n');
general;
clrscr;
if (cycle = true) then
begin
  writeln('Υπάρχει κύκλωμα αρνητικού κόστους, αυτό είναι το : ');
  write('{');
  for i := 1 to n do
  begin
    if (circ[i] > 0) then
      write('(',circ[i],',',i,') ,');
  end;
  writeln('}');
  readln;
end;
if (cycle = false) then
begin
```

```
      writeln('Δεν υπάρχει κύκλωμα αρνητικού κόστους');
      writeln;
      writeln('Το σύνολο των τόξων του δενδρισμού είναι :');
      write('{');
      for i := 1 to n do
        write('(',pred[i],',',i,') ,');
      writeln('}');
      writeln;
      writeln('Τα βάρη των κορυφών του δενδρισμού είναι :');
      writeln;
      for i := 1 to n do
      writeln('Το βάρος της κορυφής ',i,' είναι : ',dist[i]:8:3);
      readln;
    end;
  end.
```

ΠΑΡΑΡΤΗΜΑ Γ

```
program DIJKSTRA;
  uses  crt;
  type  arrnn    = array[1..20,1..20] of integer;
        arrn1    = array[1..20] of integer;
        arrn2    = array[1..20] of real;
        boolarrn = array[1..20] of boolean;

  var i,
      j,
      n,
      s,
      t,
      inf,
      u,
      v,
      y,
      recent     : integer;
  var newlabel,
      temp       : real;
  var inchr      : char;
  var w          : arrnn;
  var path       : boolean;
  var final      : boolarrn;
  var dist       : arrn2;
  var pred       : arrn1;


begin
  clrscr;
  writeln('Δώστε ένα αριθμό έτσι ώστε να είναι πάντα ο μεγαλύτερος ');
  write('(θα αναγνωρίζεται σαν άπειρο) : ');
  readln(inf);
  writeln;
  write('Δώστε τον αριθμό των κορυφών (όχι μεγαλύτερο του 20): ');
  readln(n);
  writeln;
  write('Δώστε την κορυφή πηγή : ');
```

```
readln(s);
writeln;
write('Δώστε την κορυφή αποδέκτη : ');
readln(t);
writeln;
for  i := 1 to n do
begin
  for j := 1 to n do
    w[i,j] := inf;
end;
repeat
  write('Θέλεται να δώσεται τιμή ή ');
  writeln('να κάνεται διόρθωση στον πίνακα των βαρών; y/n : ');
  inchr := readkey;
  if (inchr = 'y') then
  begin
    write('Δώστε την αρχική κορυφή του τόξου : ');
    readln(i);
    write('Δώστε την τελική κορυφή του τόξου : ');
    readln(j);
    write('w[',i,j,'] = ');
    readln(w[i,j]);
    writeln;
  end;
until (inchr = 'n');
for v := 1 to n do
begin
  dist[v]  := inf;
  final[v] := false;
  pred[v]  := -1;
end;
dist[s]  := 0;
final[s] := true;
path     := true;
recent   := s;
while not final[t] do
begin
  for v := 1 to n do
```

```
    if (w[recent,v] < inf) and (not final[v]) then
    begin
      newlabel := dist[recent] + w[recent,v];
      if newlabel < dist[v] then
      begin
        dist[v] := newlabel;
        pred[v] := recent;
      end;
    end;
    temp := inf;
    for u := 1 to n do
    if (not final[u]) and (dist[u] < temp) then
    begin
      y    := u;
      temp := dist[u];
    end;
    if temp < inf then
    begin
      final[y] := true;
      recent   := y;
    end
    else
    begin
      path     := false;
      final[t] := true;
    end;
end; (*while*)
clrscr;
writeln('Το σύνολο των τόξων του δενδρισμού είναι :');
write('{');
for i := 1 to n do
  write('(',pred[i],i,') ,');
writeln('}');
writeln;
writeln('Τα βάρη των κορυφών του δενδρισμού είναι :');
for i := 1 to n do
begin
  write('Το βάρος της κορυφής ',i,' είναι : ');
```

```
    writeln(dist[i]:8:3);
  end;
  readln;
end.
```

ΠΑΡΑΡΤΗΜΑ Δ

```
ώστε ενα πολύ μεγάλο αριθμό (θα αναγνωρίζεται σαν άπειρο) : 1000

ώστε τον αριθμό των νομισμάτων (όχι μεγαλύτερο του 20) : 5

ώστε το αρχικό νόμισμα  : 2

ώστε το τελικό νόμισμα  : 2

έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 1
ώστε την τελική κορυφή  του τόξου : 2
[12] = 0.3765
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 1
ώστε την τελική κορυφή  του τόξου : 3
[13] = 1.8400
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 1
ώστε την τελική κορυφή  του τόξου : 4
[14] = 0.1640
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 1
ώστε την τελική κορυφή  του τόξου : 5


[15] = 1.1756
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 1
[21] = 2.5925
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 3
[23] = 4.8237
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 4
[24] = 0.4256
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 5
[25] = 3.0460
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 1
[31] = 0.5422
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 2
[32] = 0.2068
```

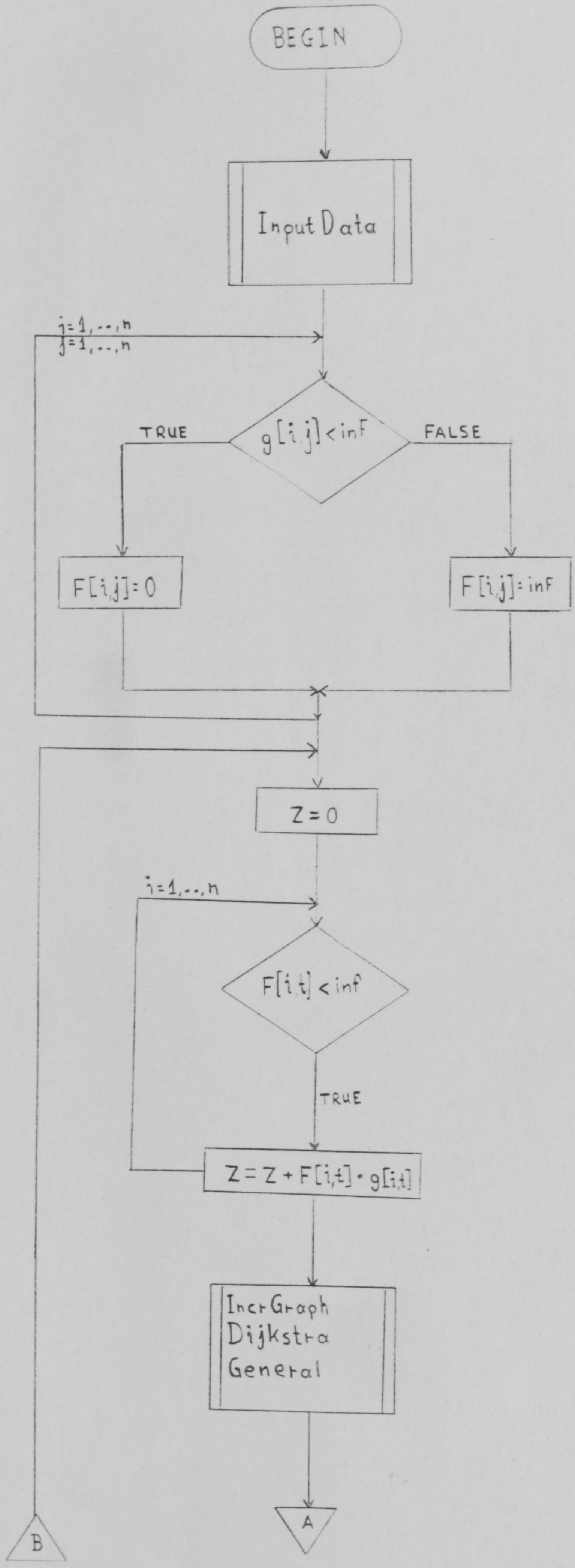
```
Δώστε την αρχική κορυφή  του τόξου : 3
Δώστε την τελική κορυφή  του τόξου : 4
g[34] = 0.08826
θέλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
Δώστε την αρχική κορυφή  του τόξου : 3
Δώστε την τελική κορυφή  του τόξου : 5
g[35] = 0.6310
θέλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
Δώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 1
g[41] = 6.0775
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 2
[42] = 2.3425
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 3
[43] = 11.2600
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 5
[45] = 7.1375
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :


ώστε την αρχική κορυφή  του τόξου : 5
ώστε την τελική κορυφή  του τόξου : 1
[51] = 0.8485
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 5
ώστε την τελική κορυφή  του τόξου : 2
[52] = 0.3275
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 5
ώστε την τελική κορυφή  του τόξου : 3
[53] = 1.5740
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :
ώστε την αρχική κορυφή  του τόξου : 5
ώστε την τελική κορυφή  του τόξου : 4
[54] = 0.1399
έλεται να κάνεται διορθώσεις στον πίνακα g; y/n :


λεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ωστε την αρχική κορυφή  του τόξου : 1
ωστε την τελική κορυφή  του τόξου : 2
12] = 8.133
λεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ωστε την αρχική κορυφή  του τόξου : 1
ωστε την τελική κορυφή  του τόξου : 3
```

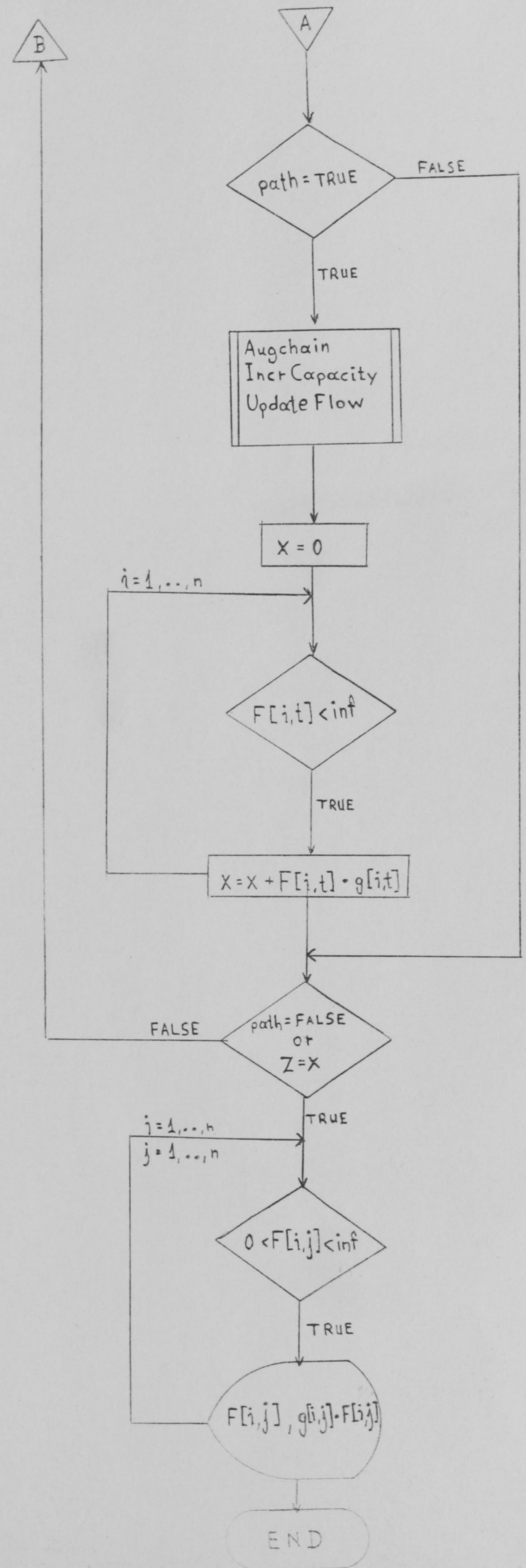
```
q[13] = 8.133
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 1
Δώστε την τελική κορυφή  του τόξου : 4
q[14] = 8.133
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 1
Δώστε την τελική κορυφή  του τόξου : 5
q[15] = 8.133
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 2
Δώστε την τελική κορυφή  του τόξου : 1
q[21] = 5
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 2
Δώστε την τελική κορυφή  του τόξου : 3
q[23] = 5
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 4
[24] = 5
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 2
ώστε την τελική κορυφή  του τόξου : 5
[25] = 5

ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 1
[31] = 15
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 2
[32] = 15
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 4
[34] = 15
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 3
ώστε την τελική κορυφή  του τόξου : 5
[35] = 15
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 1
[41] = 1
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
ώστε την αρχική κορυφή  του τόξου : 4
ώστε την τελική κορυφή  του τόξου : 2
[42] = 1
έλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
```

```
Δώστε την αρχική κορυφή  του τόξου : 4
Δώστε την τελική κορυφή  του τόξου : 3
q[43] = 1
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 4
Δώστε την τελική κορυφή  του τόξου : 5
q[45] = 1
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 5
Δώστε την τελική κορυφή  του τόξου : 1
q[51] = 10
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 5
Δώστε την τελική κορυφή  του τόξου : 2
q[52] = 10
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 5
Δώστε την τελική κορυφή  του τόξου : 3
q[53] = 10
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
Δώστε την αρχική κορυφή  του τόξου : 5
Δώστε την τελική κορυφή  του τόξου : 4
q[54] = 10
θέλεται να κάνεται διορθώσεις στον πίνακα q; y/n :
```

```
Πουλήστε    6.098  του νομίσματος 1 αγοράστε     1.000  του νομίσματος 4

Πουλήστε    2.331  του νομίσματος 2 αγοράστε    11.246  του νομίσματος 3

Πουλήστε   11.246  του νομίσματος 3 αγοράστε     6.098  του νομίσματος 1

Πουλήστε    1.000  του νομίσματος 4 αγοράστε     2.342  του νομίσματος 2

ΤΕΛΟΣ πατήστε <Enter>
```

ΠΑΡΑΡΤΗΜΑ Ε


BEGIN
Input Data
i=1,...,n
j=1,...,n
g[i,j]<inf
TRUE
FALSE
F[i,j]=0
F[i,j]=inf
Z=0
i=1,...,n
F[i,t]<inf
TRUE
Z=Z+F[i,t]·g[i,t]
IncrGraph
Dijkstra
General
B
A


A
B
path = TRUE
FALSE
TRUE
Augchain
Incr Capacity
Update Flow
X = 0
i = 1, .., n
F[i,t] < inf
TRUE
X = X + F[i,t] · g[i,t]
FALSE
path = FALSE
or
Z = X
TRUE
i = 1, .., n
j = 1, .., n
0 < F[i,j] < inf
TRUE
F[i,j] , g[i,j]·F[i,j]
END