

3 ΕΡΓΑΣΙΑ

Παράδοση μεσάνυχτα της 12/6/2022

1. Να δημιουργήσετε πρόγραμμα σε C το οποίο να δημιουργεί **απλό συνεκτικό¹ γράφο** με N κορυφές των οποίων το όνομα θα είναι διαδοχικά $1, 2 \dots N-1, N$. Στη συνέχεια με τυχαίο τρόπο να δημιουργήσετε τις διασυνδέσεις μεταξύ των κορυφών. Δηλαδή η πρώτη κορυφή θα επιλέγεται τυχαία με πόσες από τις υπόλοιπες $N-1$ κορυφές θα συνδεθεί άμεσα και στη συνέχεια, πάλι τυχαία, με ποιες από αυτές. Μετά για τη δεύτερη κορυφή το ίδιο, θα επιλέγεται τυχαία με πόσες από τις υπόλοιπες $N-2$ θα συνδεθεί άμεσα και στη συνέχεια με ποιες από αυτές και ούτω καθεξής.

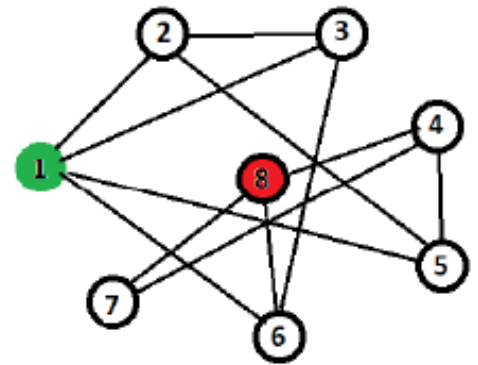
Το πρόγραμμα να χειρίζεται τον γράφο με **πίνακα γειτνίασης**. Να δημιουργήσετε συνάρτηση που να προσδιορίζει την διαδρομή που ξεκινά από την κορυφή 1 και καταλήγει στην κορυφή N και διέρχεται από τον **ελάχιστο** αριθμό κορυφών του γράφου.

Για την αρχικοποίηση της συνάρτησης τυχαιότητας θα χρησιμοποιήσετε τον AEM σας, δηλαδή `srand(AEM)`;

Το πρόγραμμα θα το τρέξετε για τυχαίο αριθμό κορυφών N , που κυμαίνονται από 6 μέχρι 10. Ο τυχαίος αυτός αριθμός N προκύπτει από τον μηχανισμό τυχαιότητας που ήδη έχει αρχικοποιηθεί.

Το πρόγραμμα θα εμφανίζει στην οθόνη τον πίνακα γειτνίασης του γράφου σας και την ή τις διαδρομές με την μορφή π.χ. $1 \rightarrow 3 \rightarrow 6 \rightarrow 8$ για $N=8$. Μπορεί οι διαδρομές αυτές να είναι περισσότερες από μία, τότε εμφανίζει όλες τις ισοδύναμες.

Για παράδειγμα στον γράφο του σχήματος έχουμε 8 συνολικά κορυφές οι οποίες διασυνδέονται μεταξύ τους με τυχαίο τρόπο. Από την κορυφή 1, όπως φαίνεται για να μεταβεί κάποιος στην 8 πρέπει να περάσει από ενδιάμεσους κόμβους. Υπάρχουν πολλές διαδρομές π.χ. $1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 8$, $1 \rightarrow 3 \rightarrow 6 \rightarrow 8$, $1 \rightarrow 6 \rightarrow 8$ ή $1 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 8$ και άλλες. Από όλες αυτές η διαδρομή που προτιμάται είναι η $1 \rightarrow 6 \rightarrow 8$.



Υπόδειξη : Για να βρείτε τις διαδρομές αυτές μπορείτε να σκεφτείτε ως εξής: Στον πίνακα γειτνίασης ελέγχετε αν στην γραμμή που αντιστοιχεί η κορυφή 1 υπάρχει η κορυφή 8. Αν υπάρχει σημαίνει ότι υπάρχει απευθείας σύνδεση μεταξύ των κορυφών και δεν υπάρχει άλλη συντομότερη. Αν δεν υπάρχει η γραμμή 8 τότε μπορείτε να πάτε στην γραμμή που αντιστοιχεί στην κορυφή 8 και να εξετάσετε αν υπάρχει κοινή κορυφή η κορυφές που υπάρχουν με την γραμμή της 1. Αν υπάρχουν αυτές είναι οι ενδιάμεσες κορυφές για τη σύνδεση της 1 με την 8. Αν είναι μόνο μία τότε μία είναι και η ζητούμενη διαδρομή, με ενδιάμεση αυτή τη μία κορυφή, αν είναι περισσότερες τότε τόσες είναι και οι ισοδύναμες διαδρομές.

Αν πάλι δεν υπάρχει καμία τότε πρέπει να προχωρήσει κανείς ένα ακόμα βήμα. Τι να κάνει δηλαδή; Να εξετάσει αν υπάρχουν κορυφές του γράφου της γραμμής 1 του πίνακα γειτνίασης που συνδέονται άμεσα με κορυφές του γράφου που υπάρχουν στην γραμμή της κορυφής 8. Αν υπάρχουν τότε οι διαδρομές που προκύπτουν έχουν τη μορφή $1 \rightarrow$ κορυφή της γραμμής 1 \rightarrow κορυφή της γραμμής 8 $\rightarrow 8$ π.χ.

¹ **Συνεκτικός** γράφος λέγεται ο γράφος του οποίου κάθε ζεύγος κορυφών συνδέεται με μια τουλάχιστον διαδρομή

στο σχήμα 1->3->6->8. Στην απίθανη ατυχή περίπτωση που δεν υπάρχει και τώρα διαδρομή συνεχίζει κανείς με παρόμοιο τρόπο.

2. Μια εταιρία πρατηρίων καυσίμων για τη διαχείριση των «καλών» πελατών της χρησιμοποιεί εταιρική ηλεκτρονική κάρτα μέλους. Κάθε πελάτης που καταχωρείται ως μέλος λαμβάνει έναν **επταψήφιο αριθμό μητρώου** τα δύο πρώτα ψηφία αφορούν την περιοχή (Καστοριά, Ηράκλειο, Θεσσαλονίκη κλπ) υπάρχουν 60 τέτοιες περιοχές (10-69), και τα πέντε επόμενα αφορούν στον πελάτη (π.χ. 5200067 αναφέρεται στον Σινάτκα που βρίσκεται στην Καστοριά περιοχή 52 και είναι ο 67 πελάτης της περιοχής). Η εταιρία ευελπιστεί να χειρίζεται περίπου 100000 πελάτες και διάλεξε ο προγραμματιστής της να διαχειρίζεται τις πληροφορίες μέσω **πίνακα κατακερματισμού** ο οποίος αντιμετωπίζει τις πιθανές συγκρούσεις με τη μέθοδο του **διπλού κατακερματισμού**.

Για το λόγο αυτό επέλεξε να χρησιμοποιεί πίνακα **διάστασης 200003** και ως συνάρτηση **δεύτερου κατακερματισμού** την συνάρτηση που δίνει **βήμα=30011-K%30011** Κ το ~~κ~~ειδί ο επταψήφιος αριθμός μητρώου του μέλους.

Κατασκευάστε πρόγραμμα σε C το οποίο θα δημιουργεί με τη χρήση της τυχαίας συνάρτησης rand() τους 100000 τυχαίους αριθμούς μητρώου των πελατών και παράλληλα να καταχωρεί για τον καθένα τη συνολική κατανάλωση σε ευρώ που έχει κάνει μέχρι τώρα.

Για να καταχωρήσετε την κατανάλωση κάθε πελάτη δημιουργείτε τυχαίο δεκαδικό αριθμό από 500.0 μέχρι 15000.0€. Για την αρχικοποίηση της σειράς των τυχαίων να χρησιμοποιήσετε την συνάρτηση srand(AEM) όπου AEM ο αριθμός μητρώου σας.

Θα ζητείται ο αριθμός μητρώου του πελάτη, ο οποίος θα πληκτρολογείται και το πρόγραμμα θα τον εντοπίζει και θα εμφανίζει τη συνολική κατανάλωση που έχει κάνει. Αν δεν τον εντοπίσει, τότε να τον καταχωρεί με αρχικό λογαριασμό τα 537.5€. Να εμφανίζονται στην οθόνη όλοι οι αριθμοί μητρώου των πελατών και τα ποσά κατανάλωσής τους με τους οποίους ο αριθμός μητρώου που πληκτρολογήσατε έρχεται **σε σύγκρουση**.

Μπορείτε με την διαχείριση που επέλεξε ο προγραμματιστής να βρείτε εύκολα τη συνολική κατανάλωση που έκαναν οι πελάτες σε συγκεκριμένη περιοχή π.χ. στην περιοχή της Καστοριάς 52;

Υπόδειξη : Τους λογαριασμούς μπορείτε να τους δημιουργήσετε με $B = 500.0 + (\text{rand}() \% 14500) / 10.0$
Τους αριθμούς μητρώου με την έκφραση $N = 100000 * (10 + \text{rand}() \% 60) + 1000 * (\text{rand}() \% 100) + \text{rand}() \% 1000$.

Υπενθύμιση : Οποιαδήποτε προσθήκη, διευκρίνιση, διόρθωση, απορία θα αντιμετωπίζεται στις Συνομιλίες.