

Προγραμματιστικές Ασκήσεις, Φυλλάδιο 3
Να ολοκληρωθούν μέχρι 11/1/12 στις 11:59 μ.μ.

Η σειρά ασκήσεων αυτή πρέπει να παραδοθεί μέχρι τις 11 Ιανουαρίου τα μεσάνυχτα. Η κάθε άσκηση μετράει για 5 μονάδες. Το άριστα είναι το 10, δηλαδή **μπορείτε να πάρετε το άριστα παραδίδοντας 2 από τις 3 ασκήσεις**. Οι μονάδες πάνω από το 10 μετράνε ως bonus. Για την υλοποίηση των ασκήσεων θα πρέπει να χρησιμοποιήσετε κλάσεις και αντικείμενα. Στις κλάσεις δεν πρέπει να χρησιμοποιείτε public μεταβλητές. Εκτός από σταθερές, δεν πρέπει να χρησιμοποιήσετε καθολικές μεταβλητές (global variables). Τα προγράμματα σας πρέπει να εξασφαλίζουν ότι αποδεσμεύτηκε όλη η μνήμη η οποία είχε δεσμευτεί.

- A.** Μια χρήσιμη δομή είναι η *ουρά προτεραιότητας (priority queue)*. Η δομή αυτή μας επιτρέπει να εισαγάγουμε στοιχεία και έχουμε γρήγορη πρόσβαση στο «μεγαλύτερο» στοιχείο, όπου το μεγαλύτερο ορίζεται ανάλογα με το τύπο δεδομένων που αποθηκεύουμε. Υλοποιήστε μια ουρά προτεραιότητας χρησιμοποιώντας τη δομή ένα Heap (σορός). Χρησιμοποιώντας class templates, η δομή που θα ορίσετε θα πρέπει να δουλεύει για οποιοδήποτε τύπο δεδομένων. Για τον πίνακα που χρησιμοποιεί εσωτερικά το heap θα πρέπει να χρησιμοποιήσετε ένα vector από την STL ώστε να μπορεί να αποθηκεύσει μεταβλητό αριθμό από δεδομένα. Η κλάση Heap που θα υλοποιήσετε θα πρέπει να υποστηρίζει τις εξής λειτουργίες:
 - Initialize:** Δημιουργεί το heap από ένα πίνακα με δεδομένα.
 - Max:** Επιστρέφει την τιμή του μέγιστου στοιχείου.
 - ExtractMax:** Αφαιρεί το μέγιστο στοιχείο από το heap, και επιστρέφει την τιμή του.
 - Insert:** Προσθέτει ένα στοιχείο στο heap.
 - Size:** Επιστρέφει τον αριθμό των στοιχείων στο heap.
 - IsEmpty:** Επιστρέφει true αν ο το heap είναι άδειο.

Υλοποιήστε μια κλάση employee, η οποία κρατάει το όνομα, το επίθετο και τον μισθό ενός employee. Η κλάση έχει μια μέθοδο για να τυπώνει τα στοιχεία του employee. Οι employees ταξινομούνται με βάση το μισθό τους. Στη main() αρχικοποιήστε το Heap με ένα πίνακα από 5 employees. Καλέστε την ExtractMax τρεις φορές και τυπώστε τα στοιχεία που εξάγετε, εισάγετε 2 employees, και τυπώστε το μέγιστο στοιχείο. Όλες οι πληροφορίες θα δίνονται από την είσοδο.

Ψευδοκώδικας (διαδικασιακός) για την υλοποίηση των αλγορίθμων ενός heap μπορεί να βρεθεί σε βιβλία δομών δεδομένων. Φωτοτυπίες από το βιβλίο Introduction to Algorithms μπορείτε να βρείτε στο γραφείο της κα. Σουλίου, και υπάρχουν και σκαναρισμένες στη σελίδα του μαθήματος.

B. Ορίστε μια *αφηρημένη* κλάση PriorityQueue που υποστηρίζει όλες τις παραπάνω λειτουργίες και κάνετε την κλάση Heap να κληρονομεί από αυτήν. Μετά ορίστε μια άλλη κλάση ListPQ η οποία επίσης κληρονομεί από την PriorityQueue και μας δίνει την ίδια λειτουργικότητα όπως το Heap, αλλά χρησιμοποιεί τον τύπο δεδομένων list από την STL, και η υλοποίηση των λειτουργιών Max και ExtractMax γίνεται διατρέχοντας τη λίστα μέχρι να βρούμε (και αν χρειαστεί να αφαιρέσουμε) το μέγιστο στοιχείο.

- Θα υλοποιήσουμε ένα παιχνίδι μεταξύ του χρήστη και του υπολογιστή. Ο χρήστης ελέγχει ένα παίχτη, και παίζει εναντίον COMPUTER_PLAYERS = 10 παιχτών που ελέγχει ο υπολογιστής. Ο κάθε παίχτης (υπολογιστή, ή παίχτη) έχει μια δύναμη (strength). Για τον παίχτη του χρήστη, η δύναμη είναι μια τυχαία τιμή μεταξύ του 0 και του MAX_USER_STRENGTH = 100, για τους παίχτες του υπολογιστή είναι μια τυχαία τιμή μεταξύ 0 και MAX_COMPUTER_STRENGTH = 20. Όλοι οι παίχτες κινούνται πάνω σε ένα πλέγμα 5x5 από 25 κόμβους. Αν ο παίχτης του χρήστη συναντηθεί σε ένα κόμβο του πλέγματος με ένα παίχτη του υπολογιστή, τότε γίνεται μία «μάχη» μεταξύ των δύο παιχτών, όπου αφαιρούμε την δύναμη του παίχτη του υπολογιστή από αυτή του παίχτη του χρήστη. Αν το αποτέλεσμα είναι αρνητικό τότε ο χρήστης χάνει, και το παιχνίδι τελειώνει. Αλλιώς ο παίχτης του υπολογιστή καταστρέφεται και το παιχνίδι συνεχίζεται. Αν γίνει συνάντηση με παραπάνω από ένα παίχτες του υπολογιστή, τότε ο παίχτης του χρήστη πολεμάει ένα-ένα με όλους τους παίχτες του υπολογιστή.

Υπάρχουν επίσης και τρία όπλα: η χειροβομβίδα (grenade), το δακρυγόνο (teargas), και ο ιός (virus). Αν ο χρήστης έχει στην κατοχή του ένα όπλο, μπορεί να το χρησιμοποιήσει σε ένα κόμβο. Η χειροβομβίδα καταστρέφει όλους τους παίχτες του υπολογιστή στον κόμβο χωρίς να χάσει δύναμη ο παίχτης-χρήστης. Το δακρυγόνο έχει ως αποτέλεσμα όλοι οι παίχτες του υπολογιστή να μετακινηθούν εκτός του κόμβου (σε κάποιο τυχαίο γειτονικό κόμβο) χωρίς να γίνει μάχη. Ο ιός έχει αποτέλεσμα η δύναμη του κάθε παίχτη-υπολογιστή στον κόμβο να μειωθεί κατά 10 μονάδες πριν γίνει η μάχη με τον παίχτη του χρήστη (αρνητική τιμή της δύναμης του παίχτη-υπολογιστή είναι αποδεκτή, θα προκαλέσει αύξηση της δύναμης του παίχτη-χρήστη). Με πιθανότητα 5% ένα όπλο είναι ελαττωματικό, οπότε η χρήση του δεν έχει κανένα αποτέλεσμα και εκτελείται η μάχη κανονικά (αυτή η πληροφορία δεν θα γίνεται γνωστή στο χρήστη παρά μόνο μετά τη χρήση του όπλου). Αν χρησιμοποιηθεί ένα όπλο, μετά καταστρέφεται. Αρχικά ο χρήστης δεν έχει στην κατοχή του όπλο. Τα τρία όπλα είναι τοποθετημένα (τυχαία) σε κόμβους του πλέγματος (ο κάθε κόμβος μπορεί να κρατάει το πολύ ένα όπλο). Ο χρήστης μπορεί να μαζέψει (collect) ένα όπλο από ένα κόμβο, να αφήσει (drop) ένα όπλο σε ένα κόμβο, ή να ανταλλάξει (swap) το όπλο που έχει με αυτό του κόμβου. Ανά πάσα στιγμή δεν μπορεί να έχει στην κατοχή του πάνω από ένα όπλο.

Στην αρχή του παιχνιδιού ο παίχτης του υπολογιστή τοποθετείται στη θέση (0,0), και οι παίχτες του υπολογιστή σε τυχαίες θέσεις. Σε κάθε γύρο του παιχνιδιού γίνονται τα εξής βήματα. Οι παίχτες του υπολογιστή κινούνται τυχαία σε ένα γειτονικό κόμβο από αυτόν που βρίσκονται. Τυπώνεται η νέα κατάσταση του πλέγματος με τις θέσεις των αντιπάλων και των όπλων. Ο χρήστης αποφασίζει αν θα μετακινήσει τον παίχτη του και προς τα πού. Μετά αποφασίζει αν θα μαζέψει, θα αφήσει, ή θα ανταλλάξει όπλο, αν υπάρχει τέτοια δυνατότητα στο νέο κόμβο. Αν έχει όπλο αποφασίζει αν θα το χρησιμοποιήσει ή όχι, και υλοποιείται η χρήση. Γίνεται η μάχη αν υπάρχουν παίχτες του υπολογιστή. Το παιχνίδι σταματάει όταν ο παίχτης-χρήστης ηττηθεί (νίκη υπολογιστή), ή σκοτωθούν όλοι οι παίχτες του υπολογιστή (νίκη χρήστη), και τυπώνεται το ανάλογο μήνυμα.

Στην υλοποίηση σας δημιουργήστε κλάσεις για τους παίχτες, μία κλάση Cell που κρατάει πληροφορίες για ένα κόμβο (αντίπαλους, όπλο), και μία αφηρημένη κλάση Weapon για το όπλο η οποία η παράγει τις κλάσεις Grenade, Teargas, και Virus. Η Weapon κρατάει τον τύπο του όπλου και αν είναι ελαττωματικό ή όχι, και μπορεί να τυπώνει αυτές τις πληροφορίες. Επίσης έχει μια εικονική συνάρτηση Use, που παίρνει σαν όρισμα μία αναφορά (ή δείκτη) σε Cell και υλοποιεί τη χρήση του όπλου.

Κατά την εκτέλεση του παιχνιδιού, τυπώνετε όση πληροφορία χρειάζεται για να μπορεί κάποιος να παρακολουθεί το παιχνίδι. Μπορείτε να δημιουργήσετε πιο εύκολες ή δύσκολες εκδοχές του παιχνιδιού αλλάζοντας τις σταθερές για τον αριθμό των παιχτών, τη μέγιστη δύναμη του παίχτη-χρήστη και παίχτη-υπολογιστή, και τον αριθμό των διαθέσιμων όπλων.

3. Γράψτε ένα πρόγραμμα διαχείρισης μιας εταιρίας λογισμικού (SoftwareCompany) η οποία έχει ένα όνομα, μία τοποθεσία και τρία τμήματα (departments): πωλήσεων (sales), marketing, και προγραμματισμού (development). Η εταιρία έχει υπαλλήλους (employees), οι οποίοι έχουν όνομα, επίθετο, ΑΦΜ και μισθό. Για τους υπαλλήλους του τμήματος πωλήσεων έχουμε επίσης τον αριθμό των πωλήσεων που έχει κάνει ο υπάλληλος, για τους υπαλλήλους marketing έχουμε τον αριθμό από τις διαφημιστικές καμπάνιες που έχει ολοκληρώσει ο υπάλληλος, και για τους υπάλληλους προγραμματιστές, τον αριθμό των γραμμών κώδικα που έχει γράψει ο υπάλληλος. Για κάθε υπάλληλο έχουμε επίσης πληροφορία για το τμήμα στο οποίο δουλεύει. Το κάθε τμήμα μπορεί να προσλάβει (Hire), και να απολύσει (Fire) έναν υπάλληλο.

Οι δραστηριότητες της εταιρίας οργανώνονται σε projects, τα οποία χαρακτηρίζονται από ένα project id, και την ημερομηνία που ξεκίνησε το project. Έχουμε τριών ειδών projects: sales projects, marketing projects, development projects. Για ένα sales project έχουμε την πόλη και τον αριθμό των πωλήσεων στον οποίο στοχεύει το project. Για ένα marketing project έχουμε την πόλη στην οποία θα γίνει η καμπάνια. Για ένα programming project έχουμε τον απαιτούμενο αριθμό από γραμμές κώδικα. Το κάθε τμήμα δημιουργεί ένα project και το αναθέτει σε ένα υπάλληλο. Ο κάθε υπάλληλος μπορεί δουλεύει μόνο σε ένα project. Όταν ολοκληρωθεί το project, ενημερώνονται οι πληροφορίες του αντίστοιχου υπαλλήλου. Για κάθε υπάλληλο έχουμε πληροφορία για το project το οποίο έχει, και για κάθε project για τον υπάλληλο που είναι υπεύθυνος.

Υλοποιήστε το παραπάνω πρόγραμμα δημιουργώντας κλάσεις για το τμήμα, τον υπάλληλο και το project, οι οποίες θα κληρονομούν σε παράγωγες κλάσεις ανάλογα με τον τύπο του τμήματος, υπαλλήλου και project. Ορίστε *εικονικές* μεθόδους για εκτύπωση των πληροφοριών, πρόσληψη και

απόλυση υπαλλήλου, δημιουργία και ολοκλήρωση project. Προσθέσετε και όποια άλλη μέθοδο χρειάζεστε. Οι πληροφορίες που θα εκτυπώνονται για κάθε υπάλληλο θα είναι οι προσωπικές του πληροφορίες και οι εξειδικευμένες πληροφορίες ανάλογα με τον τύπο του υπαλλήλου, καθώς και ο κωδικός και η ημερομηνία του project που έχει αναλάβει (αν έχει). Παρομοίως για το project θα τυπώνονται όλες τις πληροφορίες για το project, και το ονοματεπώνυμο, και ΑΦΜ του υπαλλήλου που το έχει αναλάβει. Για το τμήμα θα τυπώνουμε τις πληροφορίες του τμήματος, τον αριθμό των υπαλλήλων και των projects, τις πληροφορίες για τους υπαλλήλους και τα projects που έχει το τμήμα, καθώς και συγκεκριμένες πληροφορίες για την απόδοση του τμήματος (συνολικό αριθμό πωλήσεων, ή συνολικό αριθμό από επιτυχημένες καμπάνιες, ή αριθμό γραμμών κώδικα). Δημιουργήστε μια κλάση για την εταιρία, η οποία θα έχει ένα μόνο πίνακα με όλους τους υπαλλήλους και όλα τα projects, και μεθόδους για να τυπώνει τις πληροφορίες τους. Επίσης θα έχει και μία συνάρτηση που θα τυπώνει την απόδοση του κάθε τμήματος.

Δοκιμάστε την υλοποίησή σας δημιουργώντας δύο εταιρίες, και 12 συνολικά υπαλλήλους (4 κάθε ειδικότητας) οι οποίοι μοιράζονται ανάμεσα στις δυο εταιρίες. Δημιουργήστε projects για κάθε υπάλληλο και τυπώστε την κατάσταση όλων των τμημάτων. Ολοκληρώστε τα projects, απολύστε ένα υπάλληλο από τη μία εταιρία και προσλάβετε τον στην άλλη. Τυπώστε τις πληροφορίες και για τις δύο εταιρίες.

ΓΕΝΙΚΕΣ ΟΔΗΓΙΕΣ

Κάντε turnin τα προγράμματα σας στο lab3@cs435. Για διευκόλυνση ονομάστε τα αρχεία με τον κώδικα σας **lab3_x_dddd.cpp** και τα αρχεία των αποτελεσμάτων **res3_x_dddd.txt** όπου dddd είναι ο Α.Μ. σας και x ο αριθμός της άσκησης.

π.χ. turnin lab3@cs435 lab3_1_dddd.cpp res3_1_dddd.txt lab3_3_dddd.cpp res3_3_dddd.txt

Στον κώδικα να αναγράφονται σε σχόλια το όνομα το login και ο ΑΜ σας.