

ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΜΑΙΟΥ-ΙΟΥΝΙΟΥ 2018

Μάθημα: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τάξη: Β΄

ΗΜΕΡΟΜΗΝΙΑ: 31/05/2018

ΧΡΟΝΟΣ: 2.5 ΩΡΕΣ

ΩΡΑ: 7:45-10:15

ΟΔΗΓΟΣ ΔΙΟΡΘΩΣΗΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ

ΟΔΗΓΙΕΣ

- Το εξεταστικό δοκίμιο αποτελείται από δεκαοκτώ (18) σελίδες.
- Το εξεταστικό δοκίμιο αποτελείται από τρία μέρη Α΄, Β΄ και Γ΄.
- Το μέρος Α΄ αποτελείται από έξι (6) ερωτήσεις και κάθε ερώτηση βαθμολογείται με πέντε (5) μονάδες.
- Το μέρος Β΄ αποτελείται από τέσσερις (4) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δέκα (10) μονάδες.
- Το μέρος Γ΄ αποτελείται από δύο (2) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δεκαπέντε (15) μονάδες.
- Όλες οι απαντήσεις σας να γραφούν στο τετράδιο που θα σας δοθεί.
- Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.
- Τα Λογικά Διαγράμματα, τα Διαγράμματα Ροής Δεδομένων, τα Διαγράμματα Σχέσεων Οντοτήτων και τα περιεχόμενά τους, μπορούν να γίνουν με μολύβι.
- Κατοχή κινητού τηλεφώνου = Δολίευση.
- Απαγορεύεται η χρήση διορθωτικού υγρού.

ΜΕΡΟΣ Α'

Άσκηση 1

Ο Ζαχαρίας θέλησε να φτιάξει στο μάθημα της πληροφορικής πρόγραμμα το οποίο να υπολογίζει τον όγκο της σφαίρας, ο οποίος όπως έχει μάθει από τα μαθηματικά δίνεται από τον πιο κάτω τύπο:

$$V = \frac{4}{3} * 3.14 * r^3, \text{ όπου } r \text{ η ακτίνα της σφαίρας (ακέραιος αριθμός)}$$

Σας δίνεται το πρόγραμμα που έφτιαξε ο Ζαχαρίας στη C++ για τον πιο πάνω σκοπό.

```
#include<iostream>
using namespace std;
int main(){
    cout<<"Dose tin aktina ";
    cin>>r;
    V=4/3*3.14*r*r*r;
    cout<<V;
    return 0;
}
```

Να ξαναγράψετε το πιο πάνω πρόγραμμα, έτσι ώστε να περιλαμβάνει τις διορθώσεις – προσθήκες που να απαντούν τα πιο κάτω ερωτήματα:

Ερώτημα (α) (Βαθμός 1)

Να δηλώσετε τις απαραίτητες μεταβλητές για το πιο πάνω πρόγραμμα.

Ερώτημα (β) (Βαθμοί 2)

Να δηλώσετε την απαραίτητη βιβλιοθήκη και να τροποποιήσετε τον υπολογισμό $r*r*r$ χρησιμοποιώντας την κατάλληλη συνάρτηση.

Ερώτημα (γ) (Βαθμοί 2)

Να γράψετε τις μεταβλητές και τις εντολές που χρειάζονται, ώστε το πρόγραμμα να υπολογίζει και να τυπώνει και το εμβαδόν της επιφάνειάς της σφαίρας όπως αυτό δίνεται από τον τύπο: $4*3.14* r^2$.

Λύση

Ερώτημα (β)	{	<pre>#include<iostream> #include<cmath> using namespace std;</pre>	1
Ερώτημα (α) – 1β	{	<pre>int main(){ int r; double V,E;</pre>	1
Ερώτημα (β) – 2β	{	<pre> cout<<"Dose tin aktina "; cin>>r; V=4/3*3.14*pow(r,3);</pre>	1 -pow
Ερώτημα (γ) – 2β	{	<pre> E=4*3.14*pow(r,2); cout<<V<<endl<<E; return 0; }</pre>	1.5 0.5 - Παρουσίαση E

Άσκηση 2

Σας δίνεται το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++ το οποίο ζητά δύο ακέραιους αριθμούς (N1, N2, θεωρείστε ότι $N1 < N2$). Το πρόγραμμα παρουσιάζει όλα τα πολλαπλάσια του 20 μεταξύ των δύο αριθμών.

```
#include<iostream>
using namespace std;
int main(){
    int i, N1, N2;
    cout<<"Dose 2 arithmous:";
    cin>>N1>>N2;
    for (i = 1; i <= N2; i++){
        if ( N1 % 20 == 0)
            cout << i << endl;
    }
    return 0;
}
```

Να ξαναγράψετε το πρόγραμμα, έτσι ώστε να απαντάτε τα πιο κάτω ερωτήματα:

Ερώτημα (α) (Βαθμοί 2)

Να εντοπίσετε και να διορθώσετε δύο (2) λογικά λάθη που υπάρχουν στο πρόγραμμα.

Άσκηση 3

Σας δίνεται το πιο κάτω πρόγραμμα:

```
#include<iostream>
using namespace std;
int main(){
int i=1, num, sum=0;

while ( i <= 3 ){
    cin>>num;
    if ( num >= 10)
        sum += num;
    else
        sum -= num;
    cout<<i<<endl;
    i++;
}
cout<<"sum="<<sum;

return 0;
}
```

Ερώτημα (α) (Βαθμοί 3)

Χρησιμοποιώντας τη μέθοδο της προκαταρκτικής εκτέλεσης να παρουσιάσετε τα αποτελέσματα του πιο πάνω προγράμματος αν δοθούν με την σειρά οι αριθμοί 5, 20, 10.

ΜΕΤΑΒΛΗΤΕΣ			ΑΠΟΦΑΣΕΙΣ				ΠΑΡΟΥΣΙΑΣΗ
I	Num	Sum	I<=3	T/F	Num>=10	T/F	1 2 3 Sum=25
1	5	0	1<=3	T	5>=10	F	
		-5					
2	20	15	2<=3	T	20>=10	T	
3	10	25	3<=3	T	10>=10	T	
4			4<=3	F			

5 στήλες * 0.2 =1

4 αποτελέσματα * 0.5 = 2

Ερώτημα (β) (Βαθμοί 1)

Πόσες φορές θα εκτελεστεί η επανάληψη αν αλλάξουμε την συνθήκη `i++` σε `i--` ;

Άπειρες – Ατέρμονος Βρόγχος

Ερώτημα (γ) (Βαθμοί 1)

Ποιο θα είναι το αποτέλεσμα του πιο πάνω προγράμματος αν δοθεί αρχικά η τιμή `i = 5`.

sum=0

Άσκηση 4

Να γράψετε το πρόγραμμα που να δέχεται πραγματικούς θετικούς αριθμούς και να υπολογίζει και να εμφανίζει τον μεγαλύτερο αριθμό. Το πρόγραμμα να τερματίζει μόλις δοθεί ως είσοδος ο αριθμός 0. Να θεωρήσετε πως θα δοθεί τουλάχιστον ένας αριθμός.

Ερώτημα (α) (Βαθμός 2)

Να φτιάξετε το λογικό διάγραμμα που να υλοποιεί το πιο πάνω σενάριο.

Ερώτημα (β) (Βαθμοί 2)

Να υλοποιήσετε σε πρόγραμμα στη C++ το λογικό διάγραμμα που δημιουργήσετε στο ερώτημα (α).

Ερώτημα (γ) (Βαθμοί 1)

Να προσθέσετε στο πρόγραμμά σας την απαραίτητη βιβλιοθήκη και να γράψετε την εντολή ούτως ώστε να τυπωθεί ο μεγαλύτερος αριθμός α) μέσα σε διάστημα 8 χαρακτήρων και β) με ακρίβεια 2 δεκαδικών.

Λύση

Ερώτημα (α) - 2 βαθμοί	Ερώτημα (β) - 2 βαθμοί, Ερώτημα (γ) - 1 βαθμοί
<pre>graph TD Start([ΑΡΧΗ]) --> Init[Max = 0] Init --> Read[/ΔΩΣΕ N/] Read --> Cond1{N != 0} Cond1 -- T --> Cond2{N > max} Cond1 -- F --> Print[/ΤΥΠΩΣΕ max/] Cond2 -- T --> Assign[Max = N] Cond2 -- F --> Read Assign --> Read Read --> Cond1 Print --> End([ΤΕΛΟΣ])</pre>	<pre>#include<iostream> #include<iomanip> using namespace std; int main(){ int N, max=0, i; cin>>N; while (N != 0){ if (N>max) max=N; cin>>N; } cout<<setw(8)<<fixed<<setprecision(2)<<max<<endl; return 0; }</pre> 0.25 0.25 0.5 0.25 0.5 (αν FOR -0.5) 0.25 0.25 0.25 0.25 0.5

Άσκηση 5

Ερώτημα (α) (Βαθμοί 2)

Κατά την ανάλυση ενός πληροφοριακού συστήματος εκτελούνται διάφορες εργασίες σε διαφορετικές φάσεις. Να αναφέρετε την ονομασία της φάσης κατά την οποία διενεργούνται τα πιο κάτω:

4 * 0.5

- A) Ετοιμασία εγγράφου προδιαγραφών Καθορισμός Προδιαγραφών
- B) Σχεδίαση Διαγραμμάτων Ροής Δεδομένων Καθορισμός Προδιαγραφών
- Γ) Καθορισμός ΠΩΣ θα υλοποιηθούν οι προδιαγραφές του συστήματος Σχεδίαση
- Δ) Καθορισμός των λειτουργικών χαρακτηριστικών ενός συστήματος

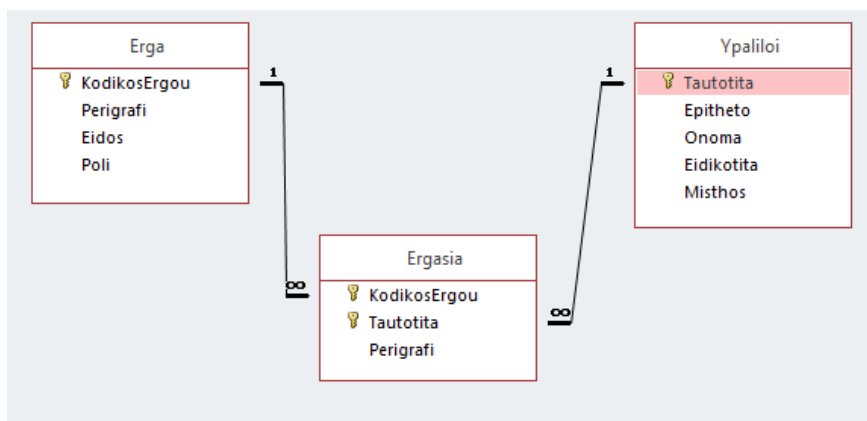
Εξακρίβωση Αναγκών και Καθορισμός Απαιτήσεων

Ερώτημα (β) (Βαθμοί 1)

Μια εργοληπτική εταιρεία αναλαμβάνει πολλά οικοδομικά έργα σε διάφορες πόλεις, στα οποία εργάζονται ένας ή περισσότεροι υπάλληλοι. Στον κάθε υπάλληλο ανατίθενται εργασίες σε ένα ή περισσότερα έργα.

Έχοντας υπόψη τα πιο πάνω να σχεδιάσετε τις σχέσεις μεταξύ των πινάκων του συστήματος που εμφανίζονται πιο κάτω και να σημειώσετε την πληθυκότητα (cardinality).

0. 5β η κάθε σχέση



Ερώτημα (γ) (Βαθμοί 2)

Να σχεδιάσετε ένα ερώτημα (query), με το οποίο θα παρουσιάζονται:

(α) η περιγραφή του έργου και το ονοματεπώνυμο όλων των υπαλλήλων που εργάζονται στα έργα που εκτελούνται στη ΛΕΜΕΣΟ.

(β) Τα στοιχεία πρέπει να εμφανίζονται με αλφαβητική σειρά με βάσει το επίθετο των υπαλλήλων.

Το ερώτημα να σχεδιαστεί συμπληρώνοντας το πλέγμα ερωτήματος στο επιπλέον φύλλο απαντήσεων που σας δόθηκε.

4 στήλες * 0.25

Field:	Epitheto	Onoma	Perigrafi	Poli
Table:	Ypaliloi	Ypaliloi	Erga	Erga
Sort:	Ascending			
Show:	☒	☒	☒	☐
Criteria:				"ΛΕΜΕΣΟΣ"
or:				

Ταξινόμηση: 0.5

Κριτήριο: 0.5

Άσκηση 6

Ερώτημα (α) (Βαθμός 1)

Τι δηλώνει η πιο κάτω εικόνα στη συσκευασία ενός παιχνιδιού;



PEGI 18

Η ταξινόμηση στην κατηγορία για ενήλικες εφαρμόζεται όταν το επίπεδο της βίας είναι τέτοιο, ώστε να αφορά στην απεικόνιση ακραίας βίας και/ή να περιλαμβάνει στοιχεία συγκεκριμένων τύπων βίας. Ο προσδιορισμός της ακραίας βίας είναι πολύ δύσκολος, καθώς σε πολλές περιπτώσεις μπορεί να είναι κάτι αρκετά υποκειμενικό, αλλά γενικά μπορεί να προσδιοριστεί ως οι απεικονίσεις βίας, που δημιουργούν ένα αίσθημα αποστροφής στον θεατή.

Ερώτημα (β) (Βαθμοί 2)

Οι περιγραφικές ενδείξεις στο πίσω μέρος της συσκευασίας ενός παιχνιδιού αναφέρουν τους κύριους λόγους για τους οποίους ένα παιχνίδι έλαβε μια ορισμένη ηλικιακή διαβάθμιση. Υπάρχουν οχτώ τέτοιες περιγραφικές ενδείξεις: βία, χυδαία γλώσσα, φόβος, ναρκωτικά, σεξ, διακρίσεις, τζόγος και online παιχνίδι με άλλα άτομα.

Να αναφέρετε τι **δηλώνουν** οι τέσσερις πιο κάτω περιγραφικές ενδείξεις. Να δώσετε επιπλέον μία πολύ **σύντομη περιγραφή** για κάθε επισήμανση.

Επισήμανση 8 * 0.25			
A) 	B) 	Γ) 	Δ) 
Online	Ναρκωτικά	Τζόγος	Χυδαία γλώσσα
Το παιχνίδι μπορεί να παίζεται μέσω Διαδικτύου	Το παιχνίδι απεικονίζει τη χρήση ναρκωτικών ή εμπεριέχει αναφορές σε αυτήν.	Το παιχνίδι παροτρύνει τζόγο ή τον διδάσκει.	Το παιχνίδι εμπεριέχει χυδαία γλώσσα ή εκφράσεις ακατάλληλες για το συγκεκριμένο ηλικιακό επίπεδο

Ερώτημα (γ) (Βαθμοί 2)

Η εικόνα που ακολουθεί παρουσιάζει τον κατάλογο με τα συμβάντα (events) που μπορούν να εφαρμοστούν σε ένα παιχνίδι και το οποίο έχει φτιαχτεί στο περιβάλλον προγραμματισμού GameMaker.



Στον πιο κάτω πίνακα παρουσιάζονται τέσσερις περιπτώσεις όπου θα πρέπει ο προγραμματιστής του παιχνιδιού να επιλέξει κάποιο συμβάν (event). Να αναφέρετε **ποιο** από τα συμβάντα θα επιλέξετε σε κάθε περίπτωση (στις απαντήσεις σας, το ίδιο συμβάν δεν μπορεί να επιλέγει πάνω από μία φορά).

		4 * 0.5
1.	Όταν τελειώσουν όλες οι ζωές που έχει στη διάθεση του, το παιχνίδι να τελειώνει και να εμφανίζεται το συνολικό σκορ.	Other
2.	Ο κεντρικός ήρωας του παιχνιδιού όταν συγκρούεται με τον τοίχο της πίστας να σταματά να κινείται.	Collision
3.	Όταν περάσει ο προκαθορισμένος χρόνος διάρκειας του παιχνιδιού, να εμφανίζεται η τελική βαθμολογία και το παιχνίδι να τελειώνει.	Alarm
4.	Όταν ο ήρωας του παιχνιδιού μαζέψει όλα τα αντικείμενα που απαιτούνται, να μεταφέρεται σε καινούρια πίστα.	Step

ΜΕΡΟΣ Β'

Άσκηση 1

Μια εταιρεία που εκτελεί συναλλαγές μέσω του διαδικτύου, έχει την πιο κάτω πολιτική για τα ταχυδρομικά τέλη που χρεώνει για την αποστολή αντικειμένων:

Αριθμός Αντικειμένων	Κλιμακωτή Χρέωση ανά αντικείμενο
1 - 2	€12.25
3 - 6	€10.25
> 6	€8.75

(Επεξήγηση: για τα 2 πρώτα αντικείμενα υπάρχει χρέωση €12.25 το κάθε αντικείμενο. Τα επόμενα 4 αντικείμενα χρεώνονται €10.25 το ένα. Πέραν των 6 αντικειμένων η χρέωση είναι €8.75 το ένα).

Εσείς θα πρέπει να γράψετε πρόγραμμα στη C++ στο οποίο η εταιρεία για κάθε συναλλαγή θα δέχεται α) το **όνομα** της χώρας στην οποία θα γίνει η συναλλαγή και β) τον αριθμό των αντικειμένων που θα αποσταλούν. Το πρόγραμμα θα τερματίζει όταν δοθεί η λέξη «TELOS» στην **ονομασία της χώρας**. Στο πρόγραμμα θα πρέπει να απαντηθούν τα πιο κάτω ερωτήματα:

Ερώτημα (α) (Βαθμός 1)

Να κάνετε τις απαραίτητες δηλώσεις των μεταβλητών που χρειάζονται για την επίλυση του πιο πάνω προβλήματος.

Ερώτημα (β) (Βαθμοί 3)

Να ελέγχετε ότι ο αριθμός αντικειμένων είναι θετικός αριθμός. Σε αντίθετη περίπτωση να ενημερώνετε τον χρήστη εμφανίζοντας ένα μήνυμα λάθους και να ζητάτε ξανά τον αριθμό αντικειμένων, μέχρι ο χρήστης να δώσει ορθά δεδομένα.

Ερώτημα (γ) (Βαθμοί 3)

Να υπολογίσετε και να παρουσιάσετε το ποσό χρέωσης για την κάθε χώρα. Το ποσό πρέπει να τυπωθεί με ακρίβεια 2 δεκαδικών ψηφίων.

Ερώτημα (δ) (Βαθμοί 3)

Να υπολογίσετε και να τυπώσετε το συνολικό αριθμό των αντικειμένων που θα αποσταλούν στην Αμερική. Αν δεν θα αποσταλούν αντικείμενα στην Αμερική, τότε να τυπώσετε το μήνυμα «ΟΧΙ Αμερική».

Λύση

```
#include<iostream>
#include<iomanip>
using namespace std;
int main(){
    int p=0,num;
    double pay;
    string xora;

    cout<<"Dose onoma xoras:";
    cin>>xora;
    while (xora != "TELOS"){
        cout<<"Dose antikimena:";
        cin>>num;
        while (num<=0){
            cout<<"Lathos - Dose arithmo:";
            cin>>num;
        }
        if ( num <= 2)
            pay = num * 12.25;
        else if (num <= 6)
            pay = 2 * 12.25 + (num -2) * 10.25;
        else
            pay = 2 * 12.25 + 4 * 10.25 + (num -6) * 8.75;
        if( xora=="AMERIKI")
            p=p+num;

        cout<<"Xreosi="<<fixed<<setprecision(2)<<pay<<endl;
        cout<<"Dose onoma xoras";
        cin>>xora;

    }
    if (p == 0)
        cout<<"OXI Ameriki="<<p<<endl;
    else
        cout<<" Ameriki = "<<p<<endl;
    return 0;
}
```

0.25

Δήλωση -0.5
Αρχική τιμή - 0.25

Ερώτημα (α) 1β

0.5

0.5

0.5

1

Ερώτημα (β) - 3β

0.5

1

1

Ερώτημα (γ) - 3β

0.5
1 Ερώτημα (δ)

0.5

0.5 Ερώτημα (β)

1

0.5

Ερώτημα (δ) -3β

Αν μήνυμα 'OXI Αμερική' μέσα στην επανάληψη, τότε -1.5

Άσκηση 2

Στις τελικές εξετάσεις Πληροφορικής ο Ζαχαρίας εξετάστηκε σε 10 διαφορετικές ερωτήσεις. Όλες οι ερωτήσεις ήταν πολλαπλής επιλογής και ο μαθητής μπορούσε να επιλέξει μία από τις 5 πιθανές απαντήσεις από το Α - Ε.

Να φτιάξετε πρόγραμμα στη γλώσσα προγραμματισμού C++ το οποίο να απαντά στα πιο κάτω ερωτήματα:

Ερώτημα (α) (Βαθμοί 2)

Να διαβάζονται: α) οι λύσεις των 10 ερωτήσεων έτσι όπως αυτές δημοσιεύτηκαν από την υπηρεσία εξετάσεων και να καταχωρούνται στον πίνακα **Liseis** και β) οι απαντήσεις στις 10 ερωτήσεις που έδωσε ο Ζαχαρίας και να αποθηκεύονται στον πίνακα **Apantiseis**. Στη περίπτωση που δεν απάντησε μία ερώτηση να εισάγεται ως απάντηση το γράμμα 'Ο'.

Παράδειγμα:

Πίνακας Liseis

A	A	B	A	D	E	A	B	B	C
---	---	---	---	---	---	---	---	---	---

Πίνακας Apantiseis

C	A	B	D	D	E	O	B	A	O
---	---	---	---	---	---	---	---	---	---

Ερώτημα (β) (Βαθμοί 2)

Σε ένα τρίτο πίνακα με το όνομα **scores** να καταχωρούνται τα εξής γράμματα: 'S', σε περίπτωση που ο Ζαχαρίας απάντησε ορθά στην ερώτηση, 'L' σε περίπτωση που απάντησε λανθασμένα και 'N' σε περίπτωση που δεν έχει απαντήσει καθόλου την ερώτηση.

Συνεχίζοντας με το πιο πάνω παράδειγμα ο πίνακας scores θα είχε την πιο κάτω μορφή:

Πίνακας Scores

L	S	S	L	S	S	N	S	L	N
---	---	---	---	---	---	---	---	---	---

Ερώτημα (γ) (Βαθμοί 3)

Για κάθε απάντηση που είναι ορθή ο Ζαχαρίας παίρνει 3 βαθμούς, για κάθε απάντηση που είναι λανθασμένη χάνει 1 και κάθε ερώτηση που δεν απαντήθηκε 0. Να βρείτε και να εμφανίσετε α) το σύνολο της βαθμολογίας που πήρε ο Ζαχαρίας και β) το μέσο όρο όπως προκύπτει από τις ορθές και λανθασμένες απαντήσεις που έδωσε.

Σημείωση: Στον μέσο όρο δεν θα λάβατε υπόψη τις ερωτήσεις που δεν έχουν απαντηθούν.

Παράδειγμα: Σύμφωνα με τον πίνακα scores όπως αυτός φαίνεται πιο πάνω, υπάρχουν 5 ορθές απαντήσεις ($5 * 3 = 15$) και τρεις λανθασμένες ($3 * 1 = 3$). Άρα το σύνολο της βαθμολογίας σε αυτή την περίπτωση είναι 12 και ο μέσος όρος $1,5$ ($12 / 8 = 1,5$).

Ερώτημα (δ) (Βαθμοί 3)

Η εξεταστική επιτροπή με νέα οδηγία καθόρισε πως η εξέταση για να είναι επιτυχημένη θα πρέπει:

- α) Η συνολική βαθμολογία να είναι τουλάχιστον 10 βαθμοί
- β) Να καταμετρηθεί πόσες φορές μία λανθασμένη απάντηση (L) ακολουθείται από ερώτηση που δεν έχει απαντηθεί (N). Αν αυτό συμβαίνει πάνω από μία φορά τότε η εξέταση δεν θεωρείται πετυχημένη.

Παράδειγμα: Στον πίνακα Scores που ακολουθεί ο μαθητής πήρε συνολική βαθμολογία 12 (5 ορθές και 3 λανθασμένες απαντήσεις). Η εξέταση δεν θεωρείται επιτυχημένη γιατί υπάρχουν δύο φορές όπου μία λανθασμένη ερώτηση ακολουθείται από ερώτηση που δεν έχει απαντηθεί.

Πίνακας Scores

S	S	L	S	S	L	N	S	L	N
---	---	---	---	---	---	---	---	---	---

Εσείς θα πρέπει να παρουσιάσετε το μήνυμα «Επιτυχής Εξέταση» σε περίπτωση που ο Ζαχαρίας πέρασε την εξέταση και το μήνυμα «Αποτυχία» σε αντίθετη περίπτωση.

Λύση

```
#include<iostream>
using namespace std;
```

```
int main (){
```

```
    char apantiseis[10], liseis[10], scores[10];
    int i, sum=0, pl=0, count=0;
```

Δήλωση -0.5

Αρχικές τιμές – 0.5 (Ερώτημα γ)

```
    for (i=0; i<10; i++) {
        cin>>liseis[i];
        cin>>apantiseis[i];
```

0.5

0.5

0.5

Ερώτημα (α) – 2β

```
        if (liseis[i]==apantiseis[i])
            scores[i]='S';
        else if (apantiseis[i]=='O')
            scores[i]='N';
        else
            scores[i]='L';
    }
```

1

0.5

0.5

Ερώτημα (β) – 2β

```
    for (i=0; i<10; i++){
        if (scores[i]=='S')
            sum+=3;
        if (scores[i]=='L')
            sum-=1;
        if (scores[i]=='N')
            pl++;
    }
```

0.5

0.5

0.5

Ερώτημα (γ) – 3β

```
    double mo=(double)sum/(10-pl);
```

1

(Αν sum/10, τότε – 1)

```
    cout<<"To sinolo tis bathmologias einai: "<<sum<<endl;
```

0.25

```
    cout<<"O mesos oros einai: "<<mo<<endl;
```

0.25

```
    for (i=0; i<9; i++)
        if (scores[i]=='L' && scores[i+1]=='N')
            count++;
    if (count<=1 && sum>=10)
        cout<<"Epitixis eksetasi";
    else
        cout<<"Apotyxia";
```

0.5

1

0.25

1

Μηνύματα 0.25

Ερώτημα (δ) – 3β

```
    return 0;
}
```

Άσκηση 3

Μια υπεραγορά έχει μηχανογραφήσει το σύστημα των πελατών και των βαθμών, που τους παρέχει ανάλογα με τις αγορές τους.

Όταν ένας πελάτης θέλει να γίνει μέλος, δίνει τα προσωπικά του στοιχεία (όνομα, επίθετο, τηλέφωνο, πόλη, ημερομηνία εγγραφής) στον υπεύθυνο εξυπηρέτησης πελατών για να καταχωρηθούν στο αρχείο πελατών. Κάθε φορά που γίνεται μια νέα καταχώρηση, το σύστημα αυτόματα δημιουργεί ένα μοναδικό κωδικό για το νέο πελάτη, ο οποίος καταχωρείται και αυτός μαζί με τα στοιχεία του πελάτη στο αρχείο πελατών. Ταυτόχρονα, ενημερώνεται το αρχείο βαθμών το οποίο περιέχει τον κωδικό, τρέχουσα ημερομηνία και 100 αρχικούς βαθμούς του πελάτη. Αμέσως μετά τυπώνεται μια κάρτα με τον κωδικό, το όνομα και το επίθετο του πελάτη και δίνεται σ' αυτόν για να την προσκομίζει όταν έρχεται στην υπεραγορά για νέες αγορές.

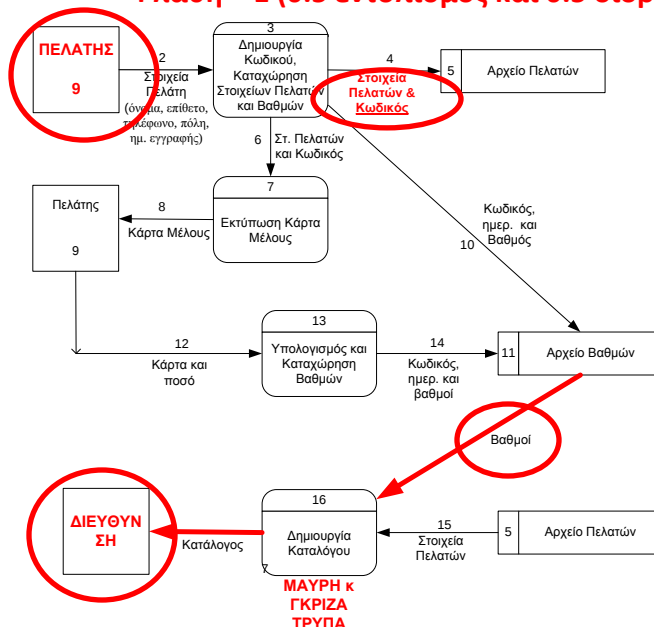
Ερχόμενος για αγορές, ο πελάτης παρουσιάζει την κάρτα του μαζί με το ποσό αγορών του. Γίνεται ο υπολογισμός των βαθμών (για κάθε 10 ευρώ αναλογεί ένας βαθμός) και γίνεται εισαγωγή στο αρχείο βαθμών με τον κωδικό του πελάτη, τρέχουσα ημερομηνία και βαθμούς.

Στο τέλος του κάθε μήνα, τυπώνεται ένας κατάλογος με τα προσωπικά στοιχεία, τον κωδικό και τους βαθμούς όλων των πελατών και παραδίδεται στον υπεύθυνο του λογιστηρίου.

Ερώτημα (α) (Βαθμοί 4)

Το παρακάτω Διάγραμμα Ροής Δεδομένων (ΔΡΔ) περιγράφει τη λειτουργία του πιο πάνω συστήματος. Να αναφέρετε και να περιγράψετε τη διόρθωση τεσσάρων (4) λαθών που υπάρχουν στο πιο κάτω ΔΡΔ.

4 λάθη * 1 (0.5 εντοπισμός και 0.5 διόρθωση)

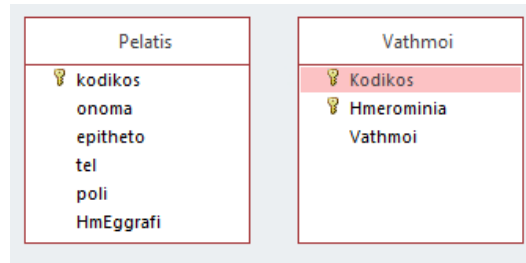


Ερώτημα (β) (Βαθμοί 2)

Να σχεδιάσετε τους δύο (2) πίνακες για το πιο πάνω σύστημα. Για τον κάθε πίνακα να καθορίσετε:

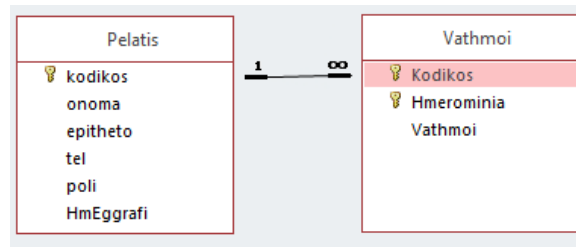
- Το όνομα του
- Τα πεδία του
- Το πεδίο ή πεδία που αποτελούν το πρωτεύων κλειδί (primary key)

1β ο κάθε πίνακας – 0.25 το ΠΚ



Ερώτημα (γ) (Βαθμοί 2)

Να σχεδιάσετε το Διάγραμμα Οντοτήτων Συσχετίσεων (ΔΟΣ) για το πιο πάνω σύστημα και να σημειώσετε το είδος πληθυκότητα (cardinality). **Αν many to many, τότε -2β**



Ερώτημα (δ) (Βαθμοί 2)

Το σχεδιάσετε ένα ερώτημα (query), με το οποίο θα παρουσιάζονται

α) το ονοματεπώνυμο και οι βαθμοί όλων των πελατών της ΛΕΥΚΩΣΙΑΣ και ΛΑΡΝΑΚΑΣ **και**

β) οι οποίοι έχουν περισσότερους από 100 βαθμούς και έκαναν εγγραφή το 2017.

Το ερώτημα να σχεδιαστεί συμπληρώνοντας το πλέγμα ερωτήματος στο επιπλέον φύλλο απαντήσεων που σας δόθηκε.

5 στήλες * 0.1 = 0.5

Field:	onoma	epitheto	Vathmoi	poli	Hmerominia
Table:	Pelatis	Pelatis	Vathmoi	Pelatis	Vathmoi
Sort:					
Show:	☒	☒	☒	☐	☐
Criteria:			> 100	"ΛΕΥΚΩΣΙΑ"	Between "1/1/2017" And "31/12/2017"
or:			> 100	"ΛΑΡΝΑΚΑ"	Between "1/1/2017" And "31/12/2017"

0.5

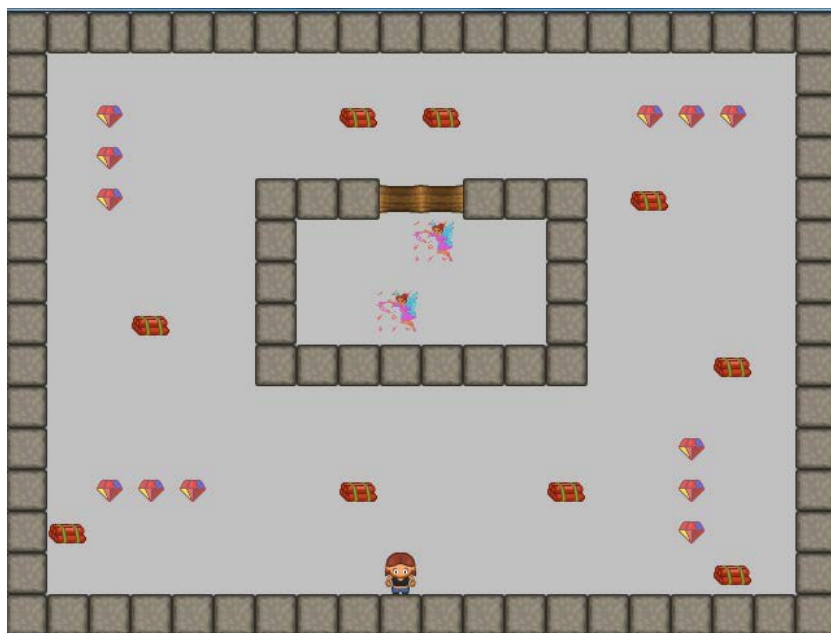
0.5

0.5

(Αν δεν γραφτούν στην 2^η γραμμή, τότε -0.5)

Άσκηση 4

Το παιχνίδι που παρουσιάζεται στην πιο κάτω εικόνα έχει φτιαχτεί από την Εμμέλεια για την φίλη της την Αριάδνη. Ο κεντρικός ήρωας είναι το ανθρωπάκι-κοριτσάκι το οποίο έχει ως σκοπό να μαζέψει τα διαμάντια που βρίσκονται μέσα στην πίστα. Για κάθε διαμάντι που μαζεύει κερδίζει τρεις βαθμούς. Εμπόδιο στο μάζεμα είναι οι δυναμίτες οι οποίοι βρίσκονται **σταθεροί** σε συγκεκριμένες θέσεις μέσα στην πίστα. Μόλις το κοριτσάκι συγκρουστεί με κάποιον από αυτούς χάνει τρεις βαθμούς. Μόλις μαζευτούν όλα τα διαμάντια ο ξύλινος τοίχος φεύγει και οι δύο νεράιδες που βρίσκονται μέσα στον εσωτερικό τοίχο και οι οποίες από την εκκίνηση του παιχνιδιού κινούνται σε τυχαίες κατευθύνσεις, ελευθερώνονται και κινούνται πλέον σε ολόκληρη την πίστα. Για να ολοκληρωθεί το παιχνίδι θα πρέπει το κοριτσάκι να μαζέψει και τις δύο αυτές νεράιδες. Για κάθε νεράιδα παίρνει επιπλέον 5 βαθμούς. Διευκρινίζεται πως το κοριτσάκι κινείται χρησιμοποιώντας το πληκτρολόγιο.



Τα αντικείμενα που έχουν φτιαχτεί για τις ανάγκες του πιο πάνω παιχνιδιού παρουσιάζονται στην εικόνα που ακολουθεί:



Για να απαντήσετε στα ερωτήματα που ακολουθούν να χρησιμοποιήσετε τους πιο κάτω πίνακες:

Events			
Keyboard <no key> <Left> <Up> <Right> <Down>	Collision obj_wall obj_fairy obj_diamond obj_dinamiti	Step	Create

Actions	
 Set the score to 0	Set Score  new score: 0 ☐ Relative
 Destroy the instance	
 Start moving in a direction	 Applies to ☒ Self ☐ Other ☐ Object: Directions: ↖ ↑ ↗ ← □ → ↙ ↓ ↘ Speed: 0 ☐ Relative
 If the number of instances is a value	object: No object number: 0 operation: Equal to
 Bounce against solid objects	Applies to ☒ Self ☐ Other ☐ Object:

Παρακάτω παρουσιάζεται παράδειγμα που δείχνει τον τρόπο με τον οποίο θα πρέπει να απαντήσετε στα ερωτήματα.

ΠΑΡΑΔΕΙΓΜΑ

Όταν συγκρούεται το κοριτσάκι (obj_girl) με τους δυναμίτες αφαιρούνται τρεις βαθμοί.

Ποιο ή ποια από τα events θα πρέπει να χρησιμοποιήσει και ποια actions, ούτως ώστε να αφαιρούνται οι τρεις βαθμοί;

ΑΠΑΝΤΗΣΗ

Επιλέγουμε το αντικείμενο **obj_girl**. Στη συνέχεια, από τα **events** επιλέγουμε το **collision** και πιο συγκεκριμένα το collision με το αντικείμενο δυναμίτη, **obj_dinamiti**. Από τα actions επιλέγουμε το **set the score to 0** και στο πεδίο **newscore** πληκτρολογούμε την τιμή **-3** και επιλέγουμε το **relative**.

Ερώτημα (α) (Βαθμοί 2)

Η Εμμέλεια παρατήρησε πως το κοριτσάκι (obj_girl) κινείται σε όλες τις κατευθύνσεις εκτός από αριστερά.

Ποιο ή ποια από τα events θα πρέπει να χρησιμοποιήσει και ποια actions, ούτως ώστε το κοριτσάκι να κινείται και αριστερά με ταχύτητα 5;

Ερώτημα (β) (Βαθμοί 2)

Οι νεράιδες (obj_fairy) δεν αναπηδούν όταν συγκρουστούν με τους τοίχους της πίστας.

Ποιο ή ποια από τα events θα πρέπει να χρησιμοποιήσει και ποια actions, ούτως ώστε οι νεράιδες να αναπηδούν όταν συγκρουστούν με τους τοίχους;

Ερώτημα (γ) (Βαθμοί 3)

Να περιγράψετε ποια actions και ποια events θα πρέπει να χρησιμοποιήσει η Εμμέλεια, ούτως ώστε το κοριτσάκι (obj_girl) μαζεύοντας τις νεράιδες, αυτές να **εξαφανίζονται** και για κάθε νεράίδα να **αυξάνεται η βαθμολογία** κατά **πέντε** βαθμούς.

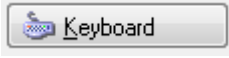
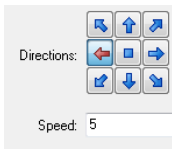
Ερώτημα (δ) (Βαθμοί 3)

Να περιγράψετε ποια actions και ποια events θα πρέπει να χρησιμοποιήσει η Εμμέλεια, ούτως ώστε μόλις το κοριτσάκι (obj_girl) μαζέψει όλες τα διαμάντια, να φεύγει- εξαφανίζεται ο ξύλινος τοίχος για να ελευθερωθούν οι νεράιδες.

Λύση

Ερώτημα (α) 2β $4 * 0.5$

- 1) Αντικείμενο: obj_girl

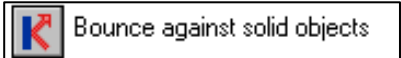
2)  3)  4) 

Επιλέγουμε το αριστερό βελάκι και βάζουμε ταχύτητα 5

Ερώτημα (β) 2β $4*0.5$

- 1) Αντικείμενο: obj_fairy

2)  3) επιλέγουμε το collision με το obj_wall

4) 

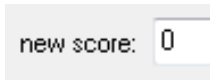
5) Επιλέγουμε: Applies to self. – **OK αν δεν το αναφέρει**

Ερώτημα (γ) 3β

- 1) Αντικείμενο: obj_girl

0.5

2)  3) επιλέγουμε το collision με το obj_fairy **0.5 + 0.25**

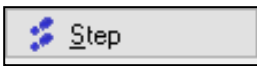
4)  5)  Βάζουμε 5 για new score **0.25 + 0.25**

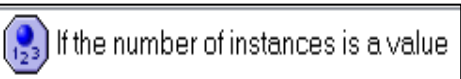
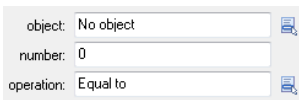
6)  Επιλέγουμε relative **0.5**

7)  8) Επιλέγουμε **other** **0.5 + 0.25**

Ερώτημα (δ) 3β $6*0.5$

- 1) Αντικείμενο: obj_girl

2) 

3)  

4) Επιλέγουμε από τα αντικείμενα το obj_diamond

5) 

- 6) επιλέγουμε από το object το αντικείμενο obj_wallwood.

ΜΕΡΟΣ Γ΄

Άσκηση 1

Στην τελική φάση του Παγκύπριου διαγωνισμού πληροφορικής «ΚΑΣΤΟΡΑΣ» διαγωνίστηκαν 20 μαθητές. Κάθε μαθητής συμμετείχε σε δύο εξετάσεις, μία γραπτή και μία προφορική. Η βαθμολογία και στις δύο εξετάσεις είναι μεταξύ 1 και 100. Για να θεωρηθεί επιτυχής η εξέταση θα πρέπει ο μαθητής στη γραπτή εξέταση να πάρει βαθμό τουλάχιστον 60 (μεγαλύτερο ή ίσον με 60) και στην προφορική τουλάχιστον 75 (μεγαλύτερο ή ίσον με 75)

Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο και να υλοποιεί τα πιο κάτω ερωτήματα.

Ερώτημα (α) (Βαθμοί 3)

Να καταχωρήσετε παράλληλα τα ονόματα των 20 μαθητών στον πίνακα **onoma**, τις βαθμολογίες της γραπτής εξέτασης στον πίνακα **grade1** και τις βαθμολογίες της προφορικής εξέτασης στον πίνακα **grade2**. Οι βαθμολογίες και στους δύο πίνακες είναι **ακέρατοι** αριθμοί.

Ερώτημα (β) (Βαθμοί 3)

Να καταχωρήσετε στον πίνακα **results** τον μέσο όρο των βαθμολογιών για κάθε μαθητή. Στους μαθητές που δεν έχουν περάσει την εξέταση να καταχωρήσετε τον **αριθμό 0**.

Ερώτημα (γ) (Βαθμοί 4)

Να βρείτε και να τυπώσετε **από τους μαθητές που έχουν περάσει την εξέταση**, τα ονόματα αυτών που πήραν τον μικρότερο μέσο όρο.

Σημείωση: Υπάρχει η πιθανότητα να υπάρχουν περισσότεροι από ένας μαθητές με τον ίδιο μέσο όρο.

Ερώτημα (δ) (Βαθμοί 5)

Να βρείτε και να τυπώσετε το μαθητή που συγκέντρωσε την ψηλότερη βαθμολογία σε κάθε εξέταση. Σε περίπτωση που ο νικητής και των δύο εξετάσεων είναι ό ίδιος ο μαθητής, να εμφανίζεται το μήνυμα: «Νικητής και στις δύο εξετάσεις είναι ο: » και δίπλα το όνομα του μαθητή.

Σημείωση: Θεωρήστε πως ένας μαθητής συγκέντρωσε την ψηλότερη βαθμολογία σε κάθε εξέταση, δεν υπάρχει δηλαδή ισοβαθμία με κάποιον άλλον μαθητή.

ΛΥΣΗ

```
#include<iostream>
#include<string>
using namespace std;
int main (){
    string onoma[20];
    int grade1[20],grade2[20],i,c=0,max1=0, max2=0,thesi1, thesi2, min=100;
    double results[20];
    for(i=0; i<20; i++){
        cout<<"Dose to onoma: ";
        cin>>onoma[i];
        cout<<"Dose proto vathmo: ";
        cin>>grade1[i];
        cout<<"Dose deytero vathmo: ";
        cin>>grade2[i];
    }
    for(i=0; i<20; i++){
        if (grade1[i]>=60 && grade2[i]>=75)
            results[i] = (double) (grade1[i]+grade2[i]/2);
        else
            results[i] = 0;
    }
    for(i=0; i<20; i++){
        if (results[i]<min && results[i]!=0)
            min=results[i];
    }
    for(i=0; i<20; i++){
        if (results[i]==min)
            cout<<onoma[i]<<endl;
    }
    for(i=0; i<20; i++){
        if (grade1[i]>max1){
            max1=grade1[i];
            thesi1=i; }
        if (grade2[i]>max2){
            max2=grade2[i];
            thesi2=i; }
    }
    if (thesi1==thesi2)
        cout<<"Nikitis kai stis dio eksetaseis einai o: ", onoma[thesi1];
    else{
        cout<<"Nikitis protis eksetasis einai: ",onoma[thesi1];
        cout<<"Nikitis deyteris eksetasis einai: ", onoma[thesi2];
    }
    return 0;}
```

Δήλωση 0.5
Αρχικές τιμές 0.5

Ερώτημα (α) – 3β

0.5
0.5
0.5
0.5

Ερώτημα (β) – 3β

1
1
1

Ερώτημα (γ) - 4β

1.5
0.5
2

Αν όχι Results[i]!=0, τότε -0.5β
Αν τυπωθεί ένα όνομα, τότε -1.5

Ερώτημα (δ) – 5β

1.5
1.5
1.5
0.5

Άσκηση 2

Στην αρχή του σχολικού έτους, η γραμματεία του σχολείου καταχωρεί τα προσωπικά στοιχεία των καθηγητών στο αρχείο καθηγητών (αριθμός ταυτότητας, όνομα, επίθετο, ειδικότητα, ημερομηνία διορισμού).

Στο τέλος του Σεπτεμβρίου η διεύθυνση του σχολείου αποφασίζει ποιες επιτροπές θα λειτουργήσουν στο σχολείο και τα στοιχεία των επιτροπών καταχωρούνται στο αρχείο επιτροπών (αριθμός επιτροπής, περιγραφή, σκοπός).

Αργότερα οι καθηγητές συμπληρώνουν μια αίτηση με τις επιτροπές που θέλουν να συμμετέχουν (αριθμός ταυτότητας καθηγητή και αριθμός επιτροπής), οι οποίες καταχωρούνται από τη γραμματεία στο αρχείο Αιτήσεων.

Όταν καταχωρηθούν οι αιτήσεις όλων των καθηγητών, δημιουργείται ένας κατάλογος με την περιγραφή της επιτροπής και τα στοιχεία των καθηγητών που επέλεξαν την κάθε επιτροπή. Ο κατάλογος δίνεται στη διεύθυνση του σχολείου.

Ερώτημα (α) (Βαθμοί 1)

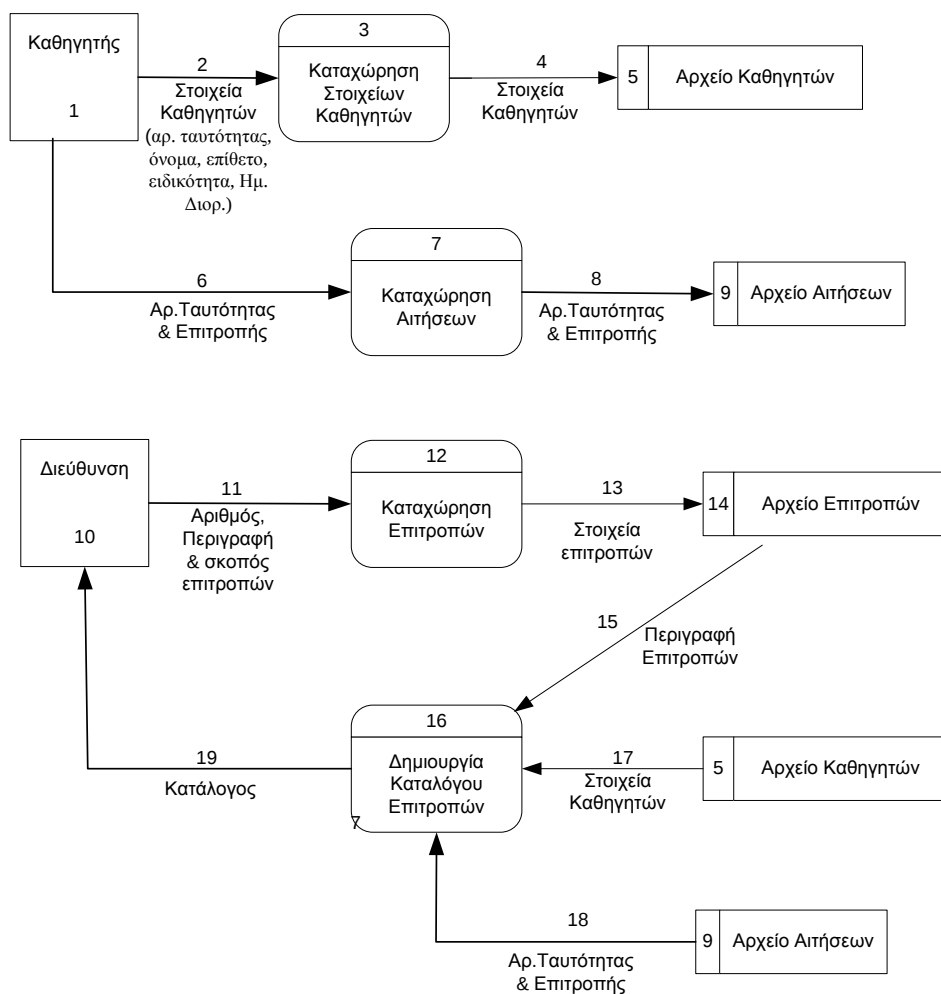
Δύο βασικά λάθη στο σχεδιασμό των Διαγραμμάτων Ροής Δεδομένων (ΔΡΔ) είναι η Μαύρη Τρύπα και το Θαύμα. Να εξηγήσετε πότε δημιουργούνται τα πιο πάνω λάθη. **2*0.5**

Μαύρη Τρύπα: Διαδικασία στην οποία υπάρχουν μόνο ροές εισόδου

Θαύμα: Διαδικασία στην οποία υπάρχουν μόνο ροές εξόδου.

Ερώτημα (β) (Βαθμοί 9)

Να σχεδιάσετε το Διάγραμμα Ροής Δεδομένων (ΔΡΔ) για τη λειτουργία που περιγράφεται πιο πάνω.



Οντότητες: $2 \times 0.5 = 1$
 Διαδικασίες: $4 \times 1 = 4$
 Αρχεία: $3 \times 0.5 = 1.5$
 Ροές: $10 \times 0.25 = 2.5$

Λανθασμένες ροές: -0.25 η κάθε μια

Μαύρες Τρύπες και Θαύματα: -0.5 το κάθε ένα

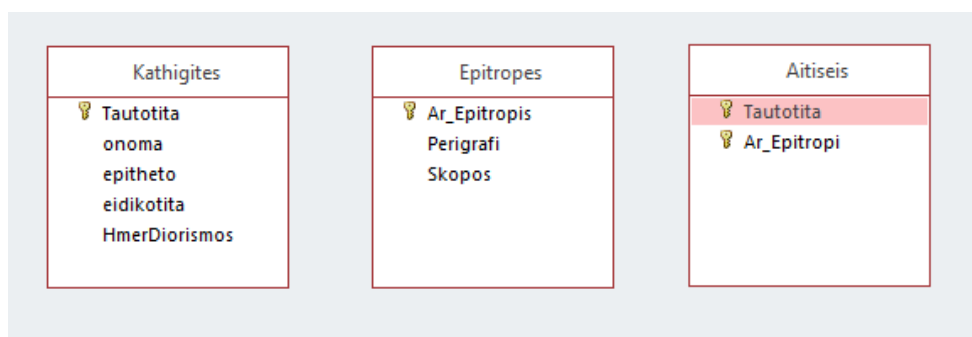
Αρίθμηση: -0.5

Ερώτημα (γ) (Βαθμοί 3)

Να καθορίσετε τους πίνακες για το πιο πάνω σύστημα. Για κάθε πίνακα να καθορίσετε:

- Το όνομα του
- Τα πεδία του
- Το πεδίο ή πεδία που αποτελούν το πρωτεύων κλειδί (primary key)

1β ο κάθε πίνακας -0.25ΠΚ

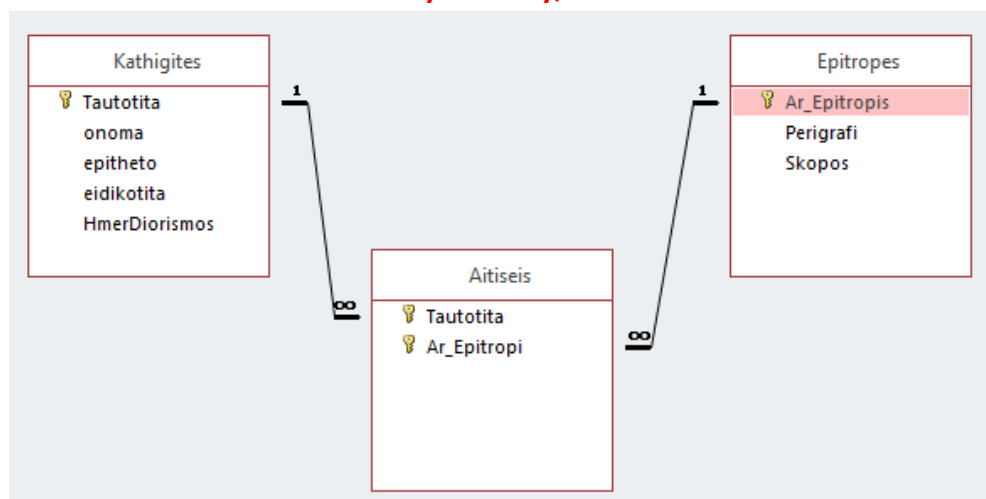


Ερώτημα (δ) (Βαθμοί 2)

Να σχεδιάσετε το Διάγραμμα Οντοτήτων Συσχετίσεων (ΔΟΣ) για το πιο πάνω σύστημα και να σημειώσετε την πληθυκότητα (cardinality). Να λάβετε υπόψη ότι μια επιτροπή μπορεί να έχει πολλούς καθηγητές και ένας καθηγητής μπορεί να συμμετέχει σε 1 ή περισσότερες επιτροπές.

1β η κάθε σχέση

Αν many to many, τότε -1.5



ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΜΑΪΟΥ - ΙΟΥΝΙΟΥ 2018

ΜΑΘΗΜΑ: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ ΤΑΞΗ: Β΄

ΗΜΕΡΟΜΗΝΙΑ: 30/05/2018

ΧΡΟΝΟΣ: 2.5 ΩΡΕΣ

ΟΝΟΜΑ: ΤΜΗΜΑ: ΑΡ.:

ΟΔΗΓΟΣ ΔΙΟΡΘΩΣΗΣ

ΟΔΗΓΙΕΣ:

- Το εξεταστικό δοκίμιο αποτελείται από δεκατέσσερις (14) σελίδες.
- Το εξεταστικό δοκίμιο περιλαμβάνει τρία μέρη Α΄, Β΄ και Γ΄.
- Όλες οι απαντήσεις σας να δοθούν στη γλώσσα προγραμματισμού C++.
- Όλες οι απαντήσεις σας να γραφούν στα φύλλα απαντήσεων που θα σας δοθούν.
- Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.
- Τα Λογικά Διαγράμματα, τα Διαγράμματα Ροής Δεδομένων, τα Διαγράμματα Σχέσεων Οντοτήτων και τα περιεχόμενά τους μπορούν να γίνουν με μολύβι.
- Κατοχή κινητού τηλεφώνου = Δολίευση.
- Απαγορεύεται η χρήση διορθωτικού υλικού.

ΜΕΡΟΣ Α΄

Το μέρος Α΄ αποτελείται από έξι (6) ασκήσεις και κάθε άσκηση βαθμολογείται με πέντε (5) μονάδες.

Άσκηση 1

Σας δίνεται το πιο κάτω πρόγραμμα, το οποίο διαβάσει τις τιμές τριών προϊόντων που αγόρασε ένας πελάτης και υπολογίζει την έκπτωση, η οποία είναι το 25% της συνολικής τους τιμής.

```
#include<iostream>
using namespace std;

int main () {

cout<<"Dose tis times twn triwn proiontwn:";
cin>>price1>>price2>>price3;

sale =(price1+price2+price3)*0.25;
cout<<"Ekptosi="<<sale<<endl;

return 0;
}
```

Ερώτημα (α) (Βαθμοί 1)

Να γράψετε την εντολή ή τις εντολές δήλωσης των απαραίτητων μεταβλητών.

float price1,price2,price3,sale;

Ερώτημα (β) (Βαθμοί 1)

Να γράψετε την εντολή που θα δήλωνε ως σταθερά το ποσοστό έκπτωσης 25%.

#define psale 0.25 ή const float psale=0.25;

Ερώτημα (γ) (Βαθμοί 3)

Να δηλώσετε τη βιβλιοθήκη, τις μεταβλητές και τις εντολές που χρειάζονται, ώστε το πρόγραμμα να υπολογίζει και να τυπώνει με ακρίβεια δύο (2) δεκαδικών ψηφίων το τελικό ποσό (συνολική τιμή – έκπτωση) που θα πληρώσει ο πελάτης.

#include<iomanip> 0,5

float final; 0,5

final=(price1+price2+price3)-sale; 1

cout<<"Teliko Poso="<<fixed<<setprecision(2)<<final; 1

Άσκηση 2

Ερώτημα (α) (Βαθμοί 1)

Να γράψετε το αποτέλεσμα της πιο κάτω έκφρασης:

$$\begin{aligned} & \text{round}(\text{pow}(10-8,2) + \text{abs}(\text{trunc}(9.5) - 19.5)) + (19\%4) \\ &= \text{round}(\text{pow}(2,2) + \text{abs}(9 - 19.5)) + 3 \\ &= \text{round}(4 + \text{abs}(-10.5)) + 3 \\ &= \text{round}(4 + 10.5) + 3 = \text{round}(14.5) + 3 = 15 + 3 = \underline{\underline{18}} \end{aligned}$$

Ερώτημα (β) (Βαθμοί 2)

Να γράψετε τα αποτελέσματα των πιο κάτω λογικών εκφράσεων, αν $a=1$, $b=3$ και $c=-7$:

$$(i) \quad (a+b==b-c) \ || \ (2*c \leq 10) \ \&\& \ (c \neq 7)$$

$$= (1+3==3+7) \ || \ (2*-7 \leq 10) \ \&\& \ (-7 \neq 7)$$

$$= (4==10) \ || \ (-14 \leq 10) \ \&\& \ (-7 \neq 7)$$

$$= F \ || \ T \ \&\& \ T = F \ || \ T = T = \underline{\underline{1}}$$

$$(ii) \quad ((4-a) \neq (10/3)) \ || \ (a*b == c+10) \ \&\& \ !(a-b \geq c)$$

$$= ((4-1) \neq (10/3)) \ || \ (1*3 == -7+10) \ \&\& \ !(1-3 \geq -7)$$

$$= (3 \neq 3) \ || \ (3 == 3) \ \&\& \ !(-2 \geq -7)$$

$$= F \ || \ T \ \&\& \ !(T) = F \ || \ T \ \&\& \ F = F \ || \ F = F = \underline{\underline{0}}$$

Ερώτημα (γ) (Βαθμοί 2)

(i) Να μετατρέψετε τον αριθμό $(135)_{10}$ από το δεκαδικό σύστημα στο δυαδικό σύστημα αρίθμησης.

(ii) Να μετατρέψετε τον αριθμό $(10101011)_2$ από το δυαδικό σύστημα στο δεκαδικό σύστημα αρίθμησης.

135/2 = 67	υπόλοιπο 1
67/2 = 33	υπόλοιπο 1
33/2=16	υπόλοιπο 1
16/2=8	υπόλοιπο 0
8/2=4	υπόλοιπο 0
4/2=2	υπόλοιπο 0
2/2=1	υπόλοιπο 0
1/2=0	υπόλοιπο 1

Λύση: **10000111**

$$10101011_2 = \underline{171}_{10}$$

$$\begin{aligned} 10101011 &= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = \\ &= 128 + 0 + 32 + 0 + 8 + 0 + 2 + 1 = 171 \end{aligned}$$

Άσκηση 3

(Βαθμοί 5)

Να γράψετε πρόγραμμα, το οποίο να διαβάζει **10** χαρακτήρες και να τους τοποθετεί σε ένα πίνακα με όνομα **xaraktires**. Το πρόγραμμα να τυπώνει το πλήθος των εμφανίσεων του χαρακτήρα **D** που ακολουθείται με δολάριο «\$». Στο πρόγραμμα, να υπάρχουν τα κατάλληλα μηνύματα εισόδου και εξόδου.

Παράδειγμα Εισόδου	Παράδειγμα Εξόδου
Dose xaraktires: A B D \$ H D D \$ H D	Plithos= 2

```
#include<iostream>
```

```
using namespace std;
```

```
0,25 – Βιβλιοθήκη + namespace + main()
```

```
int main () {
```

```
    char xaraktires[10];
```

```
0,5 plithos=0 και 0,5 υπόλοιπα μαζί
```

```
    int i,plithos=0;
```

```
cout<<"Dose xaraktires: ";
```

```
for (i=0; i<10; i++)
```

```
    cin>>xaraktires[i];
```

0,25 cout
0,5 - συνταξη for
0,5 cin

```
for (i=0; i<9; i++)
```

```
    if (xaraktires[i]=='D')
```

```
        if (xaraktires[i+1]=='$')
```

```
            plithos++;
```

0,5 - συνταξη for
0,5
0,5
0,5

```
cout<<"Plithos= "<<plithos<<endl;
```

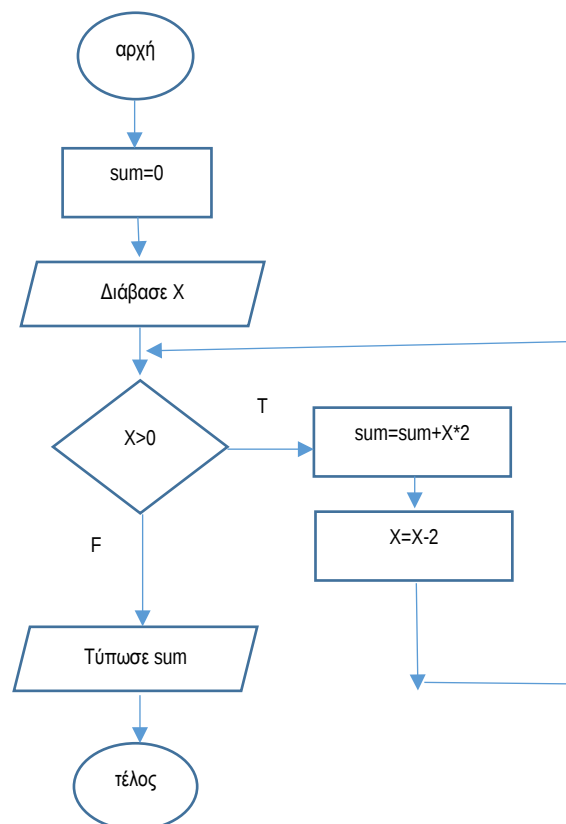
0,5

```
return 0;
```

```
}
```

Άσκηση 4

Σας δίνεται το πιο κάτω Λογικό διάγραμμα:



Ερώτημα (α) (Βαθμοί 2)

Να μετατρέψετε το πιο πάνω λογικό διάγραμμα σε πρόγραμμα.

```
#include<iostream>
using namespace std;
```

0,125

```
int main () {
```

```
int x,sum=0;
```

0,5

```
cout<<"Dose to X:";
cin>>x;
```

0,25

```
while (x>0){
sum=sum+x*2;
x=x-2;
}
```

0,5
0,25
0,25

```
cout<<"sum=: "<<sum;
```

0,125

```
return 0;
}
```

Ερώτημα (β) (Βαθμοί 2)

Χρησιμοποιώντας τη μέθοδο της προκαταρκτικής εκτέλεσης, να υπολογίσετε τα αποτελέσματα, αν δοθεί αρχικά η τιμή για το $x = 7$.

Μεταβλητές		Συνθήκη		Παρουσίαση
X	sum	X>0	A/Ψ	
7	0	7>0	A	
5	14	5>0	A	
3	24	3>0	A	
1	30	1>0	A	
-1	32	-1>0	Ψ	
				sum= 32

0,5

0,5

0,5

0,5

Ερώτημα (γ) (Βαθμοί 1)

Αν στο πρόγραμμα δεν υπήρχε η εντολή `sum = 0`, τι θα άλλαζε στο αποτέλεσμα;

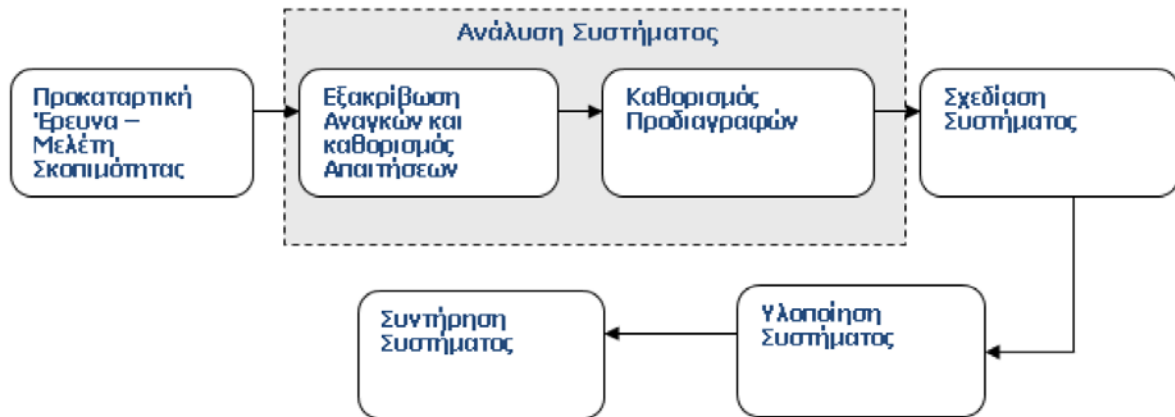
Το sum θα είχε μια απρόσμενη τιμή αφού η θέση μνήμης που δεσμεύει η μεταβλητή θα είχε μια άλλη τιμή σαν αρχική.

Άσκηση 5

Ερώτημα (α) (Βαθμοί 3)

Να αναφέρετε σε σειρά τις 6 φάσεις του Κύκλου Ζωής και Ανάπτυξης ενός Πληροφοριακού Συστήματος.

0,5 για το κάθε ένα



Ερώτημα (β) (Βαθμοί 1)

Να αναφέρετε τι περιλαμβάνει η τελευταία φάση.

Η τελευταία φάση περιλαμβάνει οποιεσδήποτε αλλαγές θα πρέπει να γίνουν στο σύστημα συνεπεία νέων αναγκών και απαιτήσεων του πελάτη, καθώς επίσης και διορθώσεις οποιονδήποτε λαθών εμφανιστούν στο σύστημα, κατά την διάρκεια της λειτουργίας του.

Ερώτημα (γ) (Βαθμοί 1)

Η Κεντρική Τράπεζα επέβαλε νέους κανονισμούς για τον υπολογισμό των επιτοκίων για όλα τα δάνεια. Το σύστημα πρέπει να τροποποιηθεί ούτως, ώστε να συμμορφώνεται με τα νέα δεδομένα. Να γράψετε αν η πρόταση αυτή είναι **Πρόβλημα**, **Ευκαιρία** ή **Εντολή**;

Άσκηση 6

Ερώτημα (α) (Βαθμοί 2)

Να απαντήσετε εάν είναι «**Σωστό**» ή «**Λάθος**» το κάθε ένα από τα πιο κάτω:

0,5 για το κάθε ένα

- A) Ένα παιγνίδι οδήγησης, το οποίο περιλαμβάνει την ολοκλήρωση διαδρομών σε δεδομένο χρόνο, είναι παιγνίδι αγώνων ταχύτητας (racer). **Σωστό**
- B) Το περιεχόμενο των παιγνιδιών στα οποία δίνεται η διαβάθμιση PEGI 16 κρίνεται κατάλληλο για όλες τις ηλικιακές ομάδες. **Λάθος**
- Γ) Το σύστημα PEGI λαμβάνει υπόψιν τις ικανότητες που απαιτούνται για ένα παιγνίδι. **Λάθος**
- Δ) Οι διαβαθμίσεις του συστήματος PEGI είναι 3,7,12,16,18. **Σωστό**

Ερώτημα (β) (Βαθμοί 3)

Οι Περιγραφικές Ενδείξεις στο πίσω μέρος της συσκευασίας ενός παιχνιδιού αναφέρουν τους κύριους λόγους για τους οποίους ένα παιχνίδι έλαβε μια ορισμένη ηλικιακή διαβάθμιση.

Να γράψετε την Περιγραφική Ένδειξη για κάθε επισήμανση.

0,5 για το κάθε ένα

α



Χυδαία
γλώσσα

β



Διακρίσεις

γ



Τζόγος

δ



OnLine

ε



Φόβος

στ



Βία

ΤΕΛΟΣ Α΄ ΜΕΡΟΥΣ

ΜΕΡΟΣ Β΄

Το μέρος Β΄ αποτελείται από τέσσερις (4) ασκήσεις και κάθε άσκηση βαθμολογείται με δέκα (10) μονάδες.

Άσκηση 1

Σε ένα μαθητικό διαγωνισμό τραγουδιού, λαμβάνουν μέρος μαθητές και μαθήτριες ηλικίας από 15 έως και 18 ετών.

Να σχεδιάσετε λογικό διάγραμμα το οποίο:

Ερώτημα (α) (Βαθμοί 1)

Για κάθε διαγωνιζόμενο, να διαβάξει τον αριθμό ταυτότητας, την ηλικία και το φύλο (Κ για κορίτσια, Α για αγόρια).

Ερώτημα (β) (Βαθμοί 2)

Να ελέγχει ότι η ηλικία του κάθε διαγωνιζομένου είναι μεταξύ 15 και 18 συμπεριλαμβανομένων, διαφορετικά να ενημερώνει τον χρήστη εμφανίζοντας μήνυμα λάθους και να ζητά ξανά ηλικία διαγωνιζομένου, μέχρι ο χρήστης να δώσει ορθά δεδομένα για την ηλικία.

Ερώτημα (γ) (Βαθμοί 1)

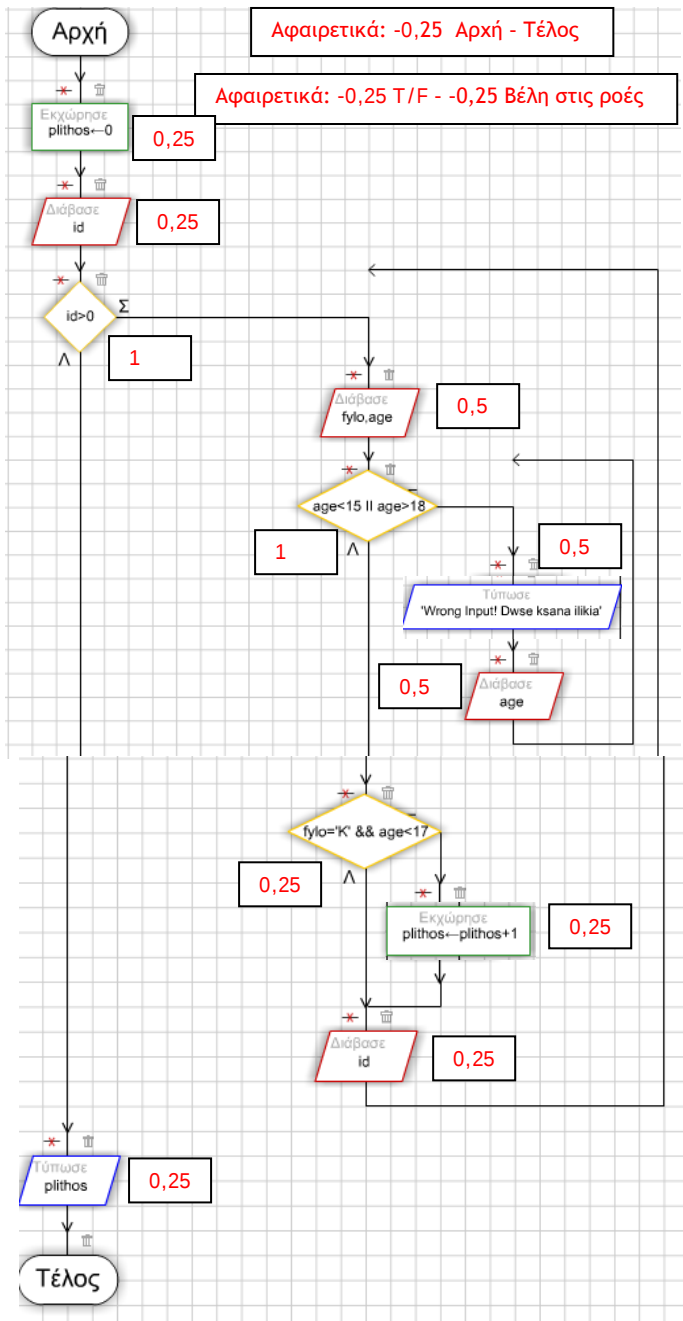
Να υπολογίζει και να τυπώνει τον αριθμό των κοριτσιών που είναι κάτω από 17 χρονών που θα λάβουν μέρος στον διαγωνισμό.

Ερώτημα (δ) (Βαθμοί 1)

Το πρόγραμμα να τερματίζει, όταν για αριθμό ταυτότητας δοθεί μηδέν(0) ή αρνητικός αριθμός.

Ερώτημα (ε) (Βαθμοί 5)

Να μετατρέψετε το πιο πάνω λογικό διάγραμμα στο αντίστοιχο πρόγραμμα. Στο πρόγραμμα, να παρουσιάζονται τα κατάλληλα μηνύματα κατά την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.



0,25 – Βιβλιοθήκη + namespace + main() + return 0

```
#include<iostream>
using namespace std;
```

```
int main () {
```

```
int id,age,plithos=0;
```

0,75 όπου plithos=0 0,25

```
char fylo;
```

```
cout<<"Dose arithmo tautotitas: ";
```

```
cin>>id;
```

```
while (id>0){
```

```
    cout<<"Dose fylo: ";
```

```
    cin>>fylo;
```

```
    cout<<"Dose ilikia: ";
```

```
    cin>>age;
```

```
    while (age<15 || age>18){
```

```
        cout<<"Wrong Input! Dwse ksana ilikia: ";
```

```
        cin>>age;
```

0,25

```
    }
```

```
    if (fylo=='K' && age<17)
```

```
        plithos++;
```

```
    cout<<"Dose arithmo tautotitas: ";
```

```
    cin>>id;
```

```
}
```

```
cout<<"Plithos koritsiwn >17= "<<plithos<<endl;
```

0,5

```
return 0;
```

```
}
```

Άσκηση 2

Σε μια εταιρεία, κατά την παραλαβή προϊόντων γίνεται έλεγχος του κωδικού και με βάση τον κωδικό του προϊόντος εντοπίζεται η αποθήκη στην οποία θα μεταφερθούν, όπως φαίνεται στον πιο κάτω πίνακα.

Κωδικός Προϊόντος	Αριθμός Αποθήκης
Μονοψήφιος Αριθμός	Ο ίδιος ο αριθμός
Διψήφιος Αριθμός	Τελευταίο ψηφίο του αριθμού
Τριψήφιος Αριθμός	Μεσαίο ψηφίο του αριθμού
Οποιοσδήποτε άλλος αριθμός	Να τυπώνεται «Wrong Code»

Να γράψετε πρόγραμμα, το οποίο:

Ερώτημα (α) (Βαθμοί 2)

Να διαβάσει 100 θετικούς ακέραιους αριθμούς που αντιστοιχούν στους 100 κωδικούς προϊόντων που παρέλαβε η εταιρεία την προηγούμενη Δευτέρα.

Ερώτημα (β) (Βαθμοί 8)

Να εντοπίζει και να εμφανίζει τον αριθμό της αποθήκης στην οποία θα μεταφερθεί το κάθε προϊόν ή το μήνυμα «Wrong Code», εάν δοθεί λανθασμένος κωδικός.

Στο πρόγραμμα, να παρουσιάζονται τα κατάλληλα μηνύματα κατά την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.

Παράδειγμα Εισόδου Δώσε κωδικό: 124	Παράδειγμα Εξόδου Αποθήκη = 2
---	---

Παράδειγμα Εισόδου Δώσε κωδικό: 7	Παράδειγμα Εξόδου Αποθήκη = 7
---	---

Παράδειγμα Εισόδου Δώσε κωδικό: 1234	Παράδειγμα Εξόδου Wrong Code
--	--

```

#include<iostream>
using namespace std;

int main () {
int i,code,apothiki;

for (i=1; i<=100; i++){
    cout<<"Dose kwdiko: ";
    cin>>code;
    if (code>=1 && code<=9){
        apothiki=code;
        cout<<"Apothiki = "<<apothiki<<endl;
    }
    else if (code>=10 && code<=99){
        apothiki=code%10;
        cout<<"Apothiki = "<<apothiki<<endl;
    }
    else if (code>=100 && code<=999){
        apothiki=(code%100)/10;
        cout<<"Apothiki = "<<apothiki<<endl;
    }
    else
        cout<<"Wrong Code"<<endl;
}

return 0;
}

```

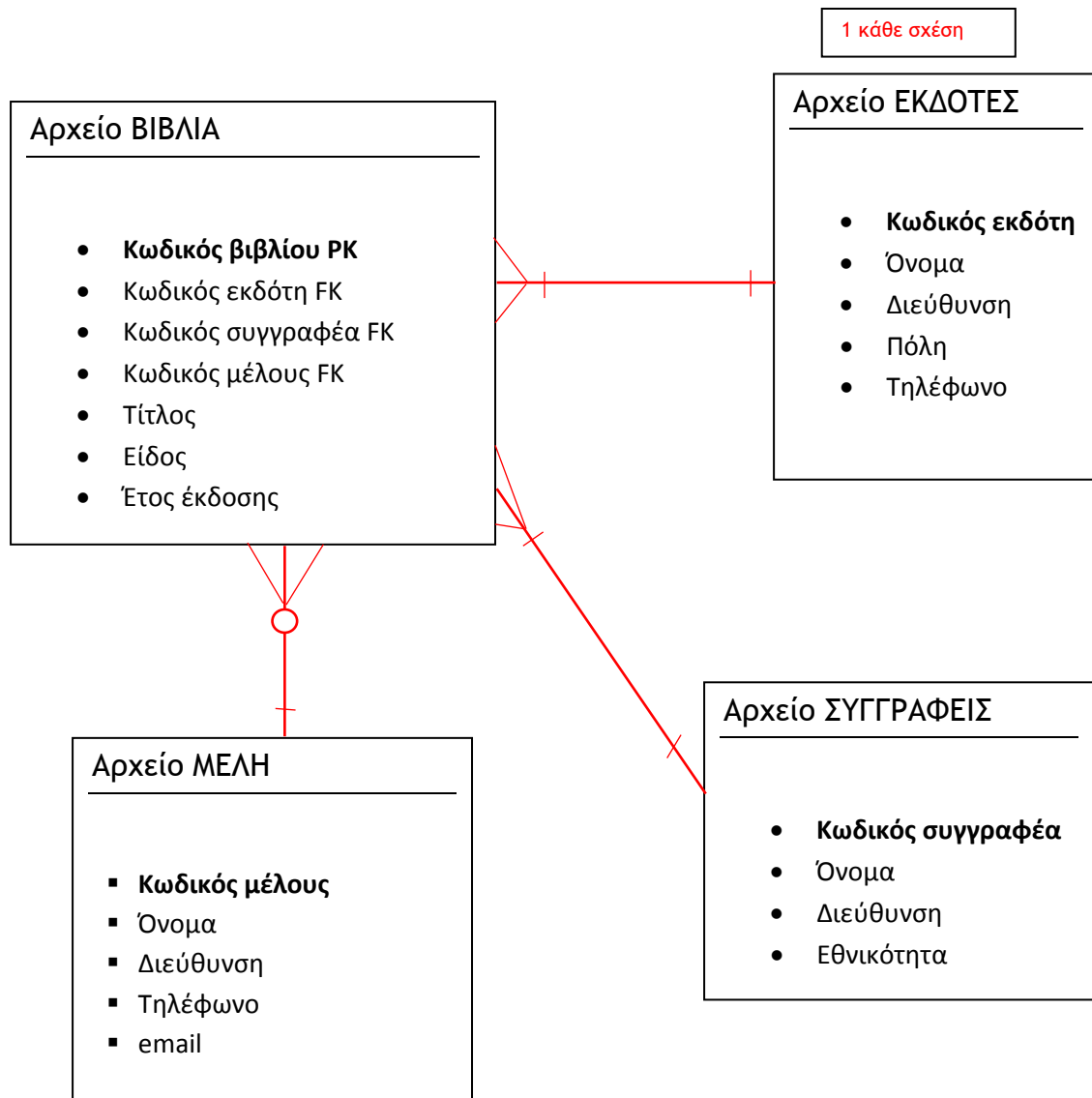
Άσκηση 3

Σε μια δανειστική βιβλιοθήκη, έχει δημιουργηθεί ένα αυτοματοποιημένο σύστημα διαχείρισης των πληροφοριών. Στη βάση δεδομένων, υπάρχουν τέσσερα αρχεία: ΒΙΒΛΙΑ, ΕΚΔΟΤΕΣ, ΣΥΓΓΡΑΦΕΙΣ και ΜΕΛΗ.

Ερώτημα (α) (Βαθμοί 3)

Να σχεδιάσετε τις σχέσεις μεταξύ των αρχείων του συστήματος όπου να φαίνεται η πληθυκότητα (cardinality). Να λάβετε υπόψη τα ακόλουθα:

- Ένας εκδότης εκδίδει ένα ή περισσότερα βιβλία
- Κάθε βιβλίο έχει έναν εκδότη
- Κάθε βιβλίο έχει ένα συγγραφέα.
- Ένας συγγραφέας έγραψε τουλάχιστον ένα βιβλίο.
- Ένα μέλος μπορεί να δανειστεί ένα, περισσότερα ή καθόλου βιβλία.



Ερώτημα (β) (Βαθμοί 3)

Να δημιουργήσετε ένα ερώτημα (query) το οποίο να εμφανίζει τους τίτλους των βιβλίων του συγγραφέα Καζαντζάκη, που έχουν εκδοθεί πριν το 1950 .

Να σχεδιάσετε στο φύλλο απαντήσεων τον πιο κάτω πίνακα συμπληρωμένο σύμφωνα με τα πιο πάνω κριτήρια.

Field:	Τίτλος	Όνομα	Έτος έκδοσης				
Table:	ΒΙΒΛΙΑ	ΣΥΓΓΡΑΦΕΙΣ	ΒΙΒΛΙΑ				
Sort:							
Show:	☒	☐	☐	☐	☐	☐	☐
Criteria:							
or:	«Καζαντζάκης»			<1950			

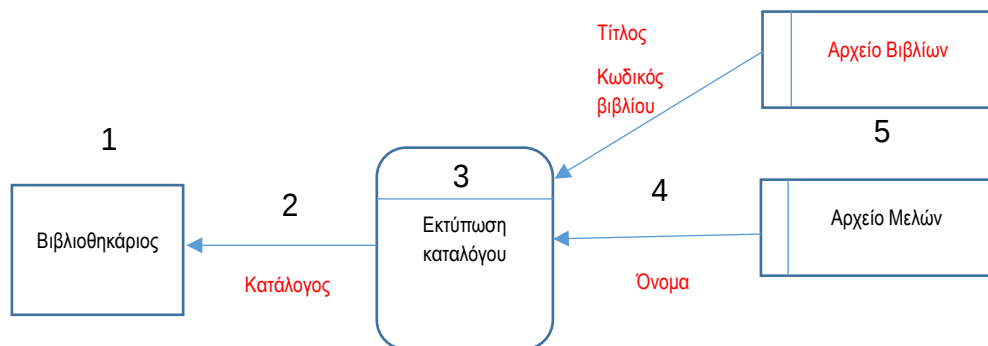
1 κάθε γραμμή

Ερώτημα (γ) (Βαθμοί 4)

Στο τέλος κάθε εβδομάδας, ο βιβλιοθηκάριος παραλαμβάνει κατάλογο που περιέχει το όνομα κάθε μέλους, τον τίτλο και τον κωδικό των βιβλίων που έχει δανειστεί.

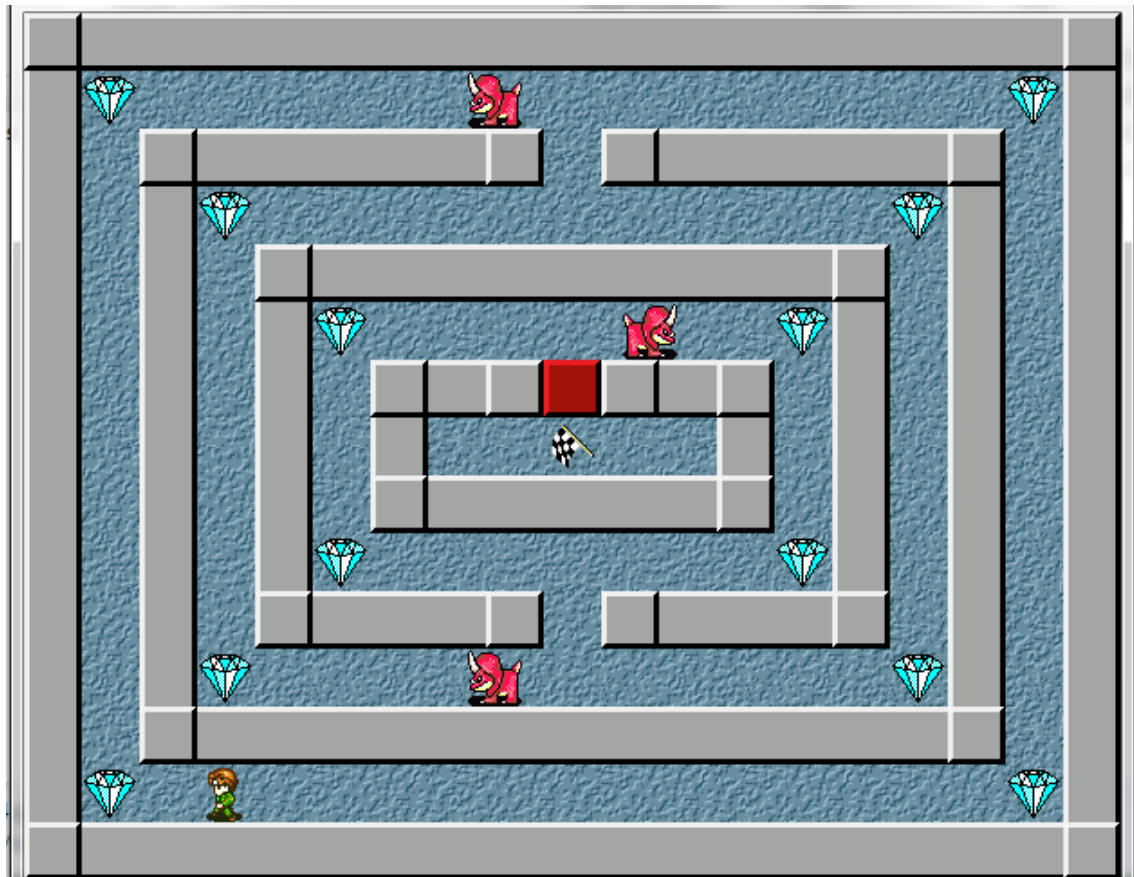
Το παρακάτω Διάγραμμα Ροής Δεδομένων (ΔΡΔ) περιγράφει τη λειτουργία του πιο πάνω συστήματος.

Στο ΔΡΔ, υπάρχουν λάθη. Να ξανασχεδιάσετε στο φύλλο απαντήσεων το ΔΡΔ διορθωμένο.



Άσκηση 4

Στο παρακάτω παιχνίδι, ο κεντρικός ήρωας, ο οποίος ελέγχεται από τον χρήστη, είναι η ανθρώπινη φιγούρα που βρίσκεται στην κάτω αριστερή γωνία στην πιο κάτω εικόνα. Στόχος του παιχνιδιού είναι ο ήρωας να φτάσει στην καρό σημαία που βρίσκεται στο κέντρο. Για κάθε διαμάντι που μαζεύει, κερδίζει 10 βαθμούς. Τρία τέρατα κινούνται στον χώρο και όταν φτάσουν σε κάποιο τοίχο, επιστρέφουν στην αντίθετη κατεύθυνση. Επαφή του ήρωα με τέρας έχει ως αποτέλεσμα την αφαίρεση 5 βαθμών.



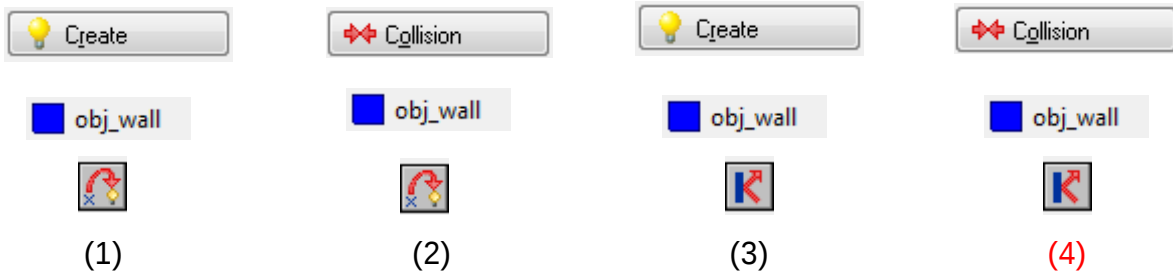
Ερώτημα (α) (Βαθμοί 1)

Να σημειώσετε ποια από τις πιο κάτω επιλογές των events και actions θα πρέπει να διαλέξετε έτσι, ώστε, όταν πατώντας το δεξί βέλος του πληκτρολογίου, ο ήρωας να κινείται δεξιά.

 Keyboard	 Keyboard	 Mouse	 Keyboard
<Up>	<Right>	Right button	<Right>
			
(1)	(2)	(3)	(4)

Ερώτημα (β) (Βαθμοί 1)

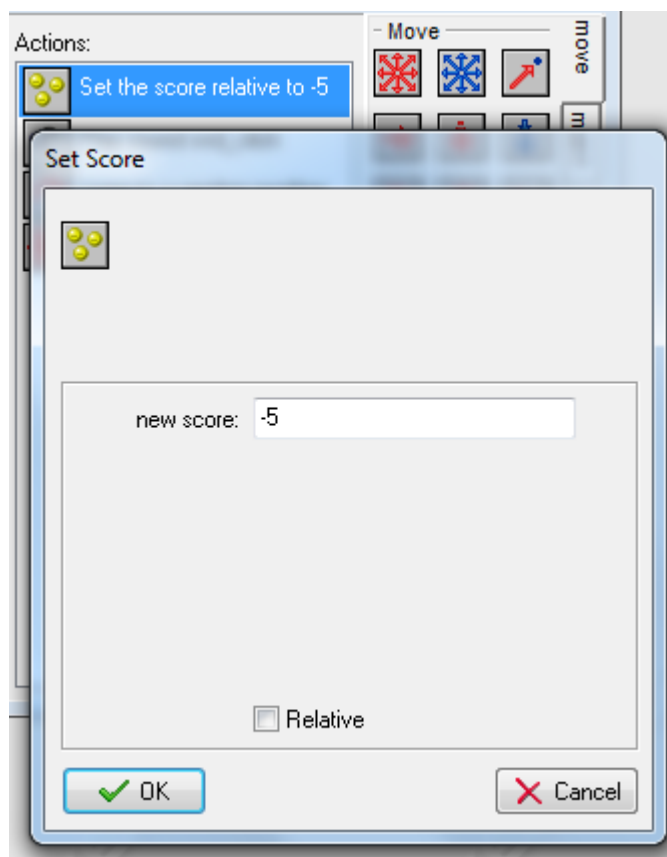
Να σημειώσετε ποια από τις πιο κάτω επιλογές των events και actions θα πρέπει να διαλέξετε έτσι, ώστε, όταν ένα τέρας φτάσει σε κάποιο τοίχο, να επιστρέφει στην αντίθετη κατεύθυνση.



Ερώτημα (γ) (Βαθμοί 2)

Στο event της σύγκρουσης του ήρωα με το τέρας, έχει γίνει ο παρακάτω προγραμματισμός, για να αφαιρούνται 5 βαθμοί από την τρέχουσα βαθμολογία. Παρόλα αυτά φαίνεται ότι το αποτέλεσμα στη βαθμολογία δεν είναι ορθό. Τι θα διορθώνατε, για να λαμβάνει υπόψη την προηγούμενη βαθμολογία;

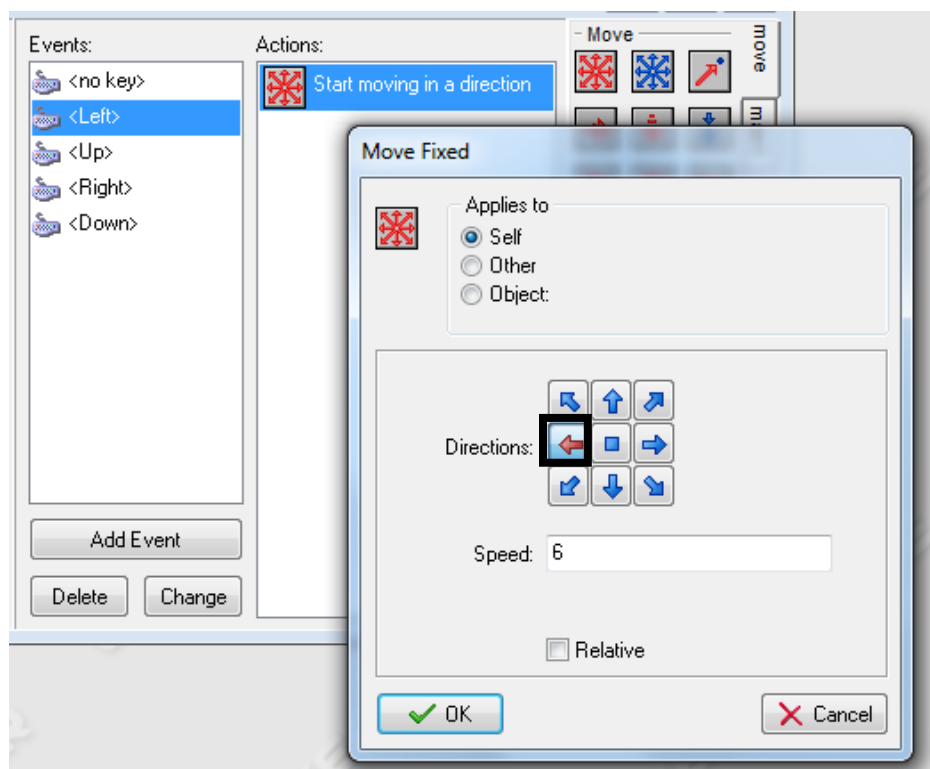
Πρέπει να επιλεγεί το Relative ☒ Relative



Ερώτημα (δ) (Βαθμοί 3)

Τι αποτέλεσμα έχει ο παρακάτω προγραμματισμός στο object του κεντρικού ήρωα; Στα Directions έχει επιλεγεί ←

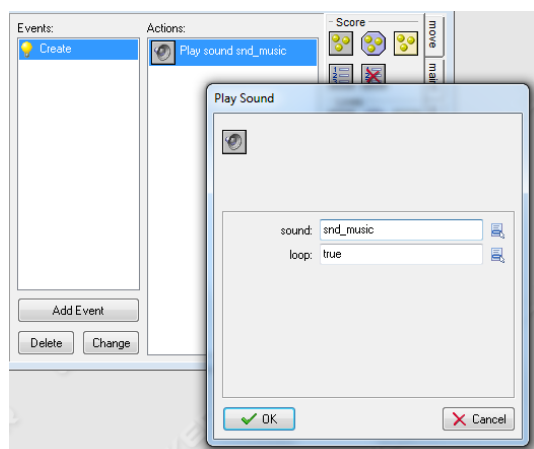
Πατώντας το αριστερό βέλος του πληκτρολογίου ο ήρωας κινείται αριστερά με ταχύτητα 6.



Ερώτημα (ε) (Βαθμοί 3)

Τι αποτέλεσμα έχει ο παρακάτω προγραμματισμός;

Με το που αρχίζει το παιχνίδι ξεκινά μουσική, η οποία όταν τελειώνει ξαναρχίζει από την αρχή, έτσι ώστε να παίζει σε όλη τη διάρκεια του παιχνιδιού.



ΤΕΛΟΣ Β' ΜΕΡΟΥΣ

ΜΕΡΟΣ Γ΄

Το μέρος Γ΄ αποτελείται από δύο (2) ασκήσεις και κάθε άσκηση βαθμολογείται με δεκαπέντε (15) μονάδες.

Άσκηση 1

Στο παγκόσμιο πρωτάθλημα ποδοσφαίρου, παίρνουν μέρος 32 χώρες από όλο τον κόσμο. Για την οργάνωση των αγώνων, κάθε ομάδα τοποθετείται σε όμιλο. Για τον σκοπό αυτόν, τα ονόματα των χωρών τοποθετήθηκαν σε πίνακα με όνομα **team** και ο όμιλος κάθε χώρας σε παράλληλο πίνακα με όνομα **group**. Σε έναν τρίτο πίνακα με όνομα **pnts**, καταχωρείται η μέχρι στιγμής βαθμολογία της καθεμιάς.

Μια μερική αναπαράσταση των πινάκων με φανταστικά δεδομένα φαίνεται πιο κάτω:

team	group	pnts
Russia	A	3
Uruguay	A	4
France	C	6
Germany	F	6
Mexico	F	0
...	...	

Να γράψετε πρόγραμμα, το οποίο να υλοποιεί τα πιο κάτω. Στο πρόγραμμα, να παρουσιάζονται τα κατάλληλα μηνύματα κατά την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.

Ερώτημα (α) (Βαθμοί 5)

Να δηλώσετε όλες τις απαραίτητες βιβλιοθήκες και μεταβλητές και να γράψετε τις κατάλληλες εντολές για την εισαγωγή των δεδομένων στους πίνακες.

Ερώτημα (β) (Βαθμοί 2)

Να τυπώσετε τις χώρες που ανήκουν στον όμιλο F.

Ερώτημα (γ) (Βαθμοί 2)

Να τυπώσετε τον όμιλο στον οποίο ανήκει η Αργεντινή (Argentina).

Ερώτημα (δ) (Βαθμοί 2)

Να υπολογίσετε και να τυπώσετε τον αριθμό των χωρών που δεν έχουν ακόμα κανένα βαθμό.

Ερώτημα (ε) (Βαθμοί 2)

Να τυπώσετε τη χώρα που προηγείται στον όμιλο D. Σε περίπτωση που ισοβαθούν και άλλες χώρες στην πρώτη θέση, να τυπωθούν και αυτές.

Ερώτημα (στ) (Βαθμοί 2)

Να ελέγξετε εάν η Κροατία (Croatia) και η Νιγηρία (Nigeria) ανήκουν στον ίδιο όμιλο και να τυπώσετε το μήνυμα «ανήκουν στον ίδιο όμιλο» ή «δεν ανήκουν στον ίδιο όμιλο» ανάλογα.

```

#include <iostream>
#include <cmath>
#define N 32
using namespace std;

int main() {
    string team[N];
    char group[N];
    int i, pnts[N];

    for (i=0; i<N; i++) {
        cout<<"dose omada : ";
        cin>>team[i];
        cout<<"dose omilo : ";
        cin>>group[i];
        cout<<"dose yathmologia : ";
        cin>>pnts[i];
    }
    cout<<endl<<"omades omilou F:"<<endl;
    for (i=0; i<N; i++)
        if (group[i]=='F')
            cout<<team[i]<<endl;

    for (i=0; i<N; i++)
        if (team[i]=="Argentina")
            cout<<"H Argentinini anhkei ston omilo "<<group[i]<<endl;

    int c=0;
    for (i=0; i<N; i++)
        if (pnts[i]==0)
            c++;
    cout<<c<<" omades den exoun kanena bathmo"<<endl;

    int max=0;
    for (i=0; i<N; i++)
        if (group[i]=='D' && pnts[i]>max)
            max=pnts[i];
    cout<<"stin korifi tou D omilou:"<<endl;
    for (i=0; i<N; i++)
        if (group[i]=='D' && pnts[i]==max)
            cout<<team[i]<<endl;

    char Cgroup, Ngroup;
    for (i=0; i<N; i++) {
        if (team[i]=="Croatia") Cgroup=group[i];
        if (team[i]=="Nigeria") Ngroup=group[i];
    }
    if (Cgroup==Ngroup)
        cout<<"anikoun ston idio omilo";
    else
        cout<<"den anikoun ston idio omilo";
}

```

0,5

2

2,5

2

2

2

2

2

Άσκηση 2

Με το νέο σύστημα διορισμών, οι νέοι καθηγητές πρέπει να παρακαθήσουν σε εξετάσεις, στις οποίες θα πιστοποιηθούν οι γνώσεις τους. Για τον σκοπό αυτόν, χρησιμοποιείται αυτοματοποιημένο σύστημα και ακολουθείται η πιο κάτω διαδικασία:

Οι υποψήφιοι για πρόσληψη προσέρχονται για εγγραφή και δίνουν τα προσωπικά τους στοιχεία (ονοματεπώνυμο, διεύθυνση, τηλέφωνο), τα οποία καταχωρούνται στο αρχείο υποψηφίων. Μαζί με τα στοιχεία αυτά, καταχωρείται και ένας μοναδικός αριθμός (κωδικός εγγραφής), ο οποίος δημιουργείται αυτόματα από το σύστημα. Αμέσως μετά, παραδίδεται στον υποψήφιο η βεβαίωση εγγραφής του, που περιλαμβάνει όλα του τα στοιχεία μαζί με τον κωδικό.

Σε μεταγενέστερο στάδιο, οι υποψήφιοι λαμβάνουν μέρος σε γραπτές εξετάσεις. Αφού τελειώσουν οι εξετάσεις και τα γραπτά διορθωθούν, οι διορθωτές παραδίδουν τον κωδικό εγγραφής και τον βαθμό κάθε υποψηφίου, τα οποία καταχωρούνται στο αρχείο αποτελεσμάτων.

Μετά από μια εβδομάδα και με βάση τα αποτελέσματα των εξετάσεων, δημιουργείται μια λίστα, που παραδίδεται στα Μέσα Μαζικής Επικοινωνίας (ΜΜΕ) για δημοσίευση. Για σκοπούς προστασίας των προσωπικών δεδομένων των υποψηφίων, η λίστα αυτή περιλαμβάνει μόνο τους κωδικούς τους και τους βαθμούς που πήραν στην εξέταση.

Μια επιστολή που περιλαμβάνει το ονοματεπώνυμο, τη διεύθυνση, τον κωδικό και το αποτέλεσμα της εξέτασης (βαθμός) τυπώνεται και αποστέλλεται στον υποψήφιο για ενημέρωση.

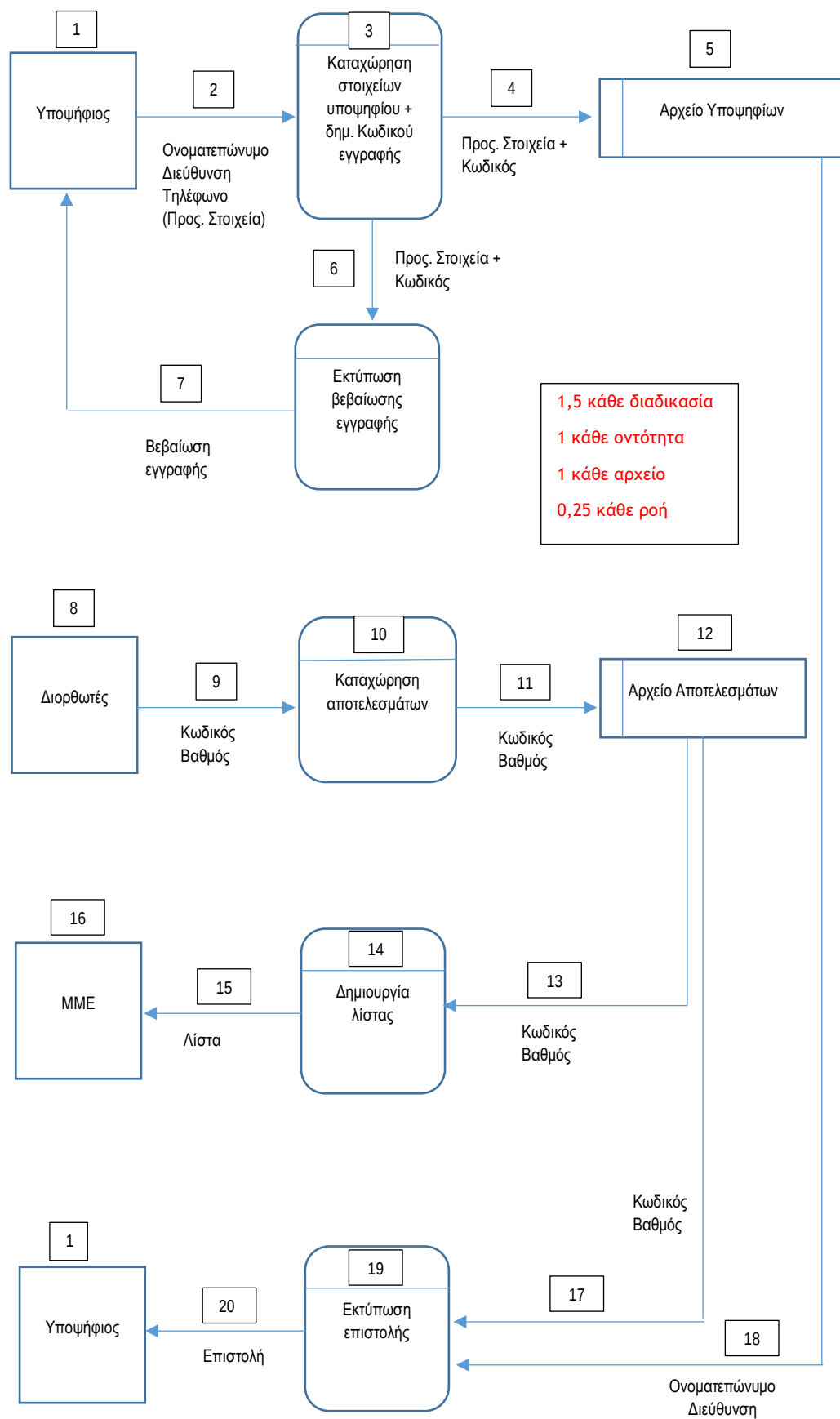
Να σχεδιάσετε το Διάγραμμα Ροής Δεδομένων (ΔΡΔ) για τη λειτουργία που περιγράφεται πιο πάνω.

(Βαθμοί 15)

ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ

Η Διευθύντρια

Παναγιώτα Χρυσοχού – Αναστασιάδου



ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΜΑΪΟΥ-ΙΟΥΝΙΟΥ 2018

Μάθημα: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τάξη: Β΄

ΗΜΕΡΟΜΗΝΙΑ: 04/06/2018

ΔΙΑΡΚΕΙΑ: 2 ΩΡΕΣ και 30 ΛΕΠΤΑ

ΟΔΗΓΙΕΣ

- Το εξεταστικό δοκίμιο αποτελείται από δεκαοκτώ (18) σελίδες.
- Το εξεταστικό δοκίμιο αποτελείται από τρία (3) μέρη Α΄, Β΄ και Γ΄.
- Το μέρος Α΄ αποτελείται από έξι (6) ερωτήσεις και κάθε ερώτηση βαθμολογείται με πέντε (5) μονάδες.
- Το μέρος Β΄ αποτελείται από τέσσερις (4) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δέκα (10) μονάδες.
- Το μέρος Γ΄ αποτελείται από δύο (2) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δεκαπέντε (15) μονάδες.
- Να γράψετε όλες τις απαντήσεις στα φύλλα εξέτασης που θα σας δοθούν.
- Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.
- Τα Λογικά Διαγράμματα, τα Διαγράμματα Ροής Δεδομένων, τα Διαγράμματα Σχέσεων Οντοτήτων και τα περιεχόμενά τους μπορούν να γίνουν με μολύβι.
- Κατοχή κινητού τηλεφώνου = Δολίευση
- Απαγορεύεται η χρήση διορθωτικού υλικού.
- Οι μοναδικές βιβλιοθήκες που επιτρέπονται στη δημιουργία προγραμμάτων, είναι η `<iostream>` και η `<cmath>`, εκτός αν αναφέρεται διαφορετικά στην εκφώνηση.

ΜΕΡΟΣ Α΄

Άσκηση 1

Ερώτημα (α) (Βαθμοί 1)

Να γράψετε το αποτέλεσμα της πιο κάτω έκφρασης, η οποία είναι γραμμένη στην γλώσσα προγραμματισμού C++ :

`round(3.48) + 3 * pow(2,2) - trunc(1.47-(5%4)) + sqrt(16) + abs(-1)`

3 + 12 - 1 + 4 + 1 = 19

Ερώτημα (β) (Βαθμοί 2)

Να γράψετε τα αποτελέσματα των πιο κάτω λογικών εκφράσεων οι οποίες είναι γραμμένες στη γλώσσα προγραμματισμού C++, αν **x=true**, **y=false** και **z=true** :

(ι) `(x==(x && z)) || ((z || y) != x) && (x && y || z || !y)`

`(T==(T && T)) || ((T || F) != T) && (T && F || T || !F))=`

`(T==T)||((T!=T) && (F||T||!F))=T||(F&&T)==T||F=TRUE`

(ιι) `((x && y) || !z) || !((x || y) == z)`

`((T&&F)||!T) ||!((T||F) == T)=(F ||F)||(!T==T)=F||F=FALSE`

Ερώτημα (γ) (Βαθμοί 2)

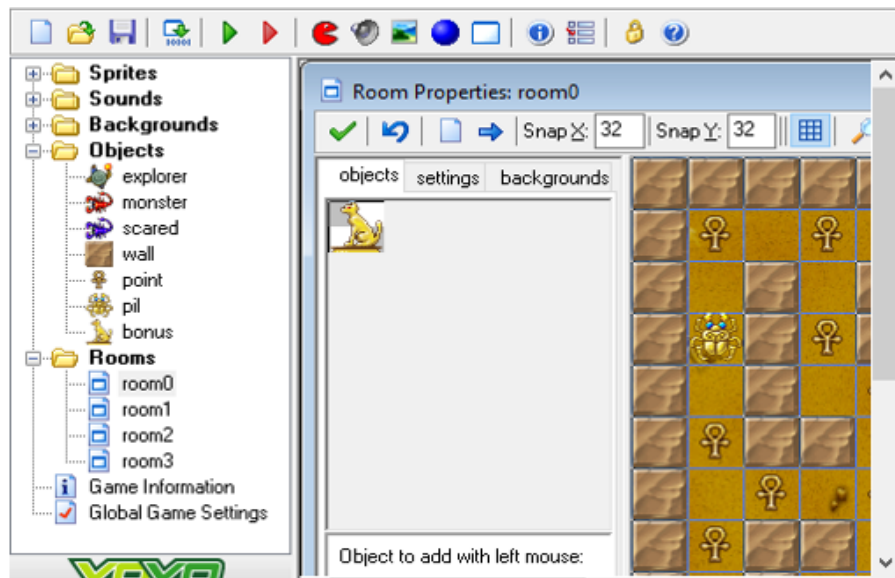
Να εξηγήσετε τη διαφορά ανάμεσα στον χρήστη και στο πελάτη ενός πληροφοριακού συστήματος. Μπορεί το ίδιο άτομο να είναι ταυτόχρονα χρήστης και πελάτης σε ένα πληροφοριακό σύστημα;

Ο χρήστης είναι αυτός ο οποίος θα χρησιμοποιεί το πρόγραμμα και ο πελάτης είναι αυτός ο οποίος θα πληρώσει για το πρόγραμμα. Πολλές φορές ο πελάτης και ο χρήστης είναι το ίδιο άτομο.

Άσκηση 2

Ερώτημα (α) (Βαθμοί 2)

Στην πιο κάτω εικόνα φαίνεται ένα στιγμιότυπο από την δημιουργία του παιχνιδιού **Treasure Island** από την Υογο Games στο Game Maker. Να γράψετε πόσα αντικείμενα (objects) και πόσα δωμάτια (πίστες) χρησιμοποιούνται στο παιχνίδι αυτό.

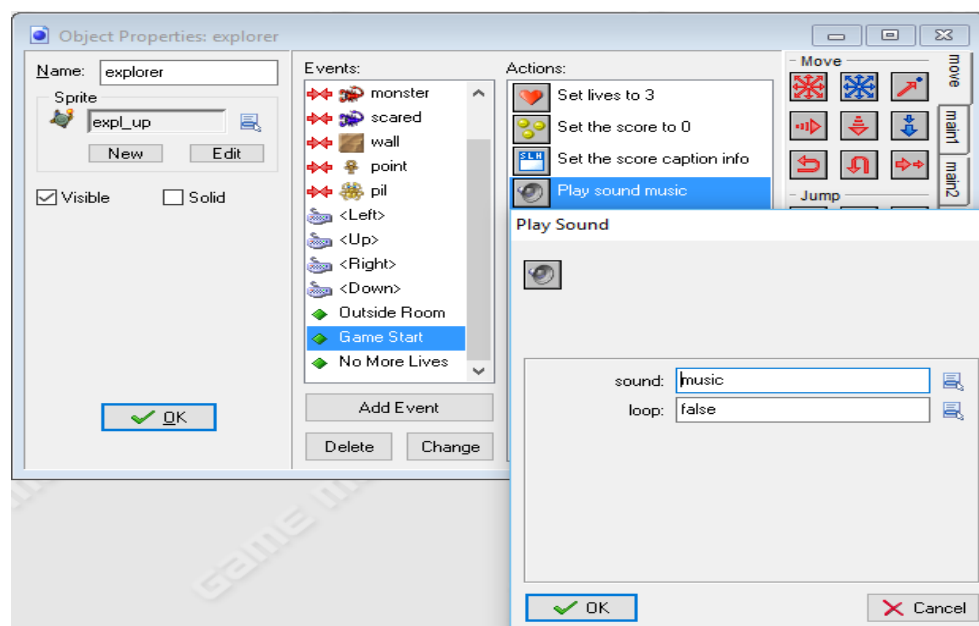


Δωμάτια (rooms): 4

Αντικείμενα (Objects): 7

Ερώτημα (β) (Βαθμοί 2)

Έχετε προσθέσει στο αντικείμενο (object) **explorer** το event **Game Start**. Στα actions έχετε προσθέσει το **Play sound Music** και έχετε εισαγάγει το μουσικό κομμάτι με το όνομα **music** όπως φαίνεται στην πιο κάτω εικόνα. Παρόλα αυτά, έχετε τρέξει το παιχνίδι και το μουσικό αυτό κομμάτι **δεν** ακούγεται καθ' όλη την διάρκεια του παιχνιδιού. Πως θα λύσετε το πρόβλημα;



Ιδιότητα Loop: Πρέπει να αλλάξει η τιμή από False σε True.

Ερώτημα (γ) (Βαθμοί 1)

Ποιο από τα events που φαίνονται στην πιο κάτω εικόνα πρέπει να προσθέσετε στο αντικείμενο του εξερευνητή (**explorer**), ώστε όταν **περνά πάνω από ένα αντικείμενο**, το αντικείμενο αυτό να **διαγράφεται**;

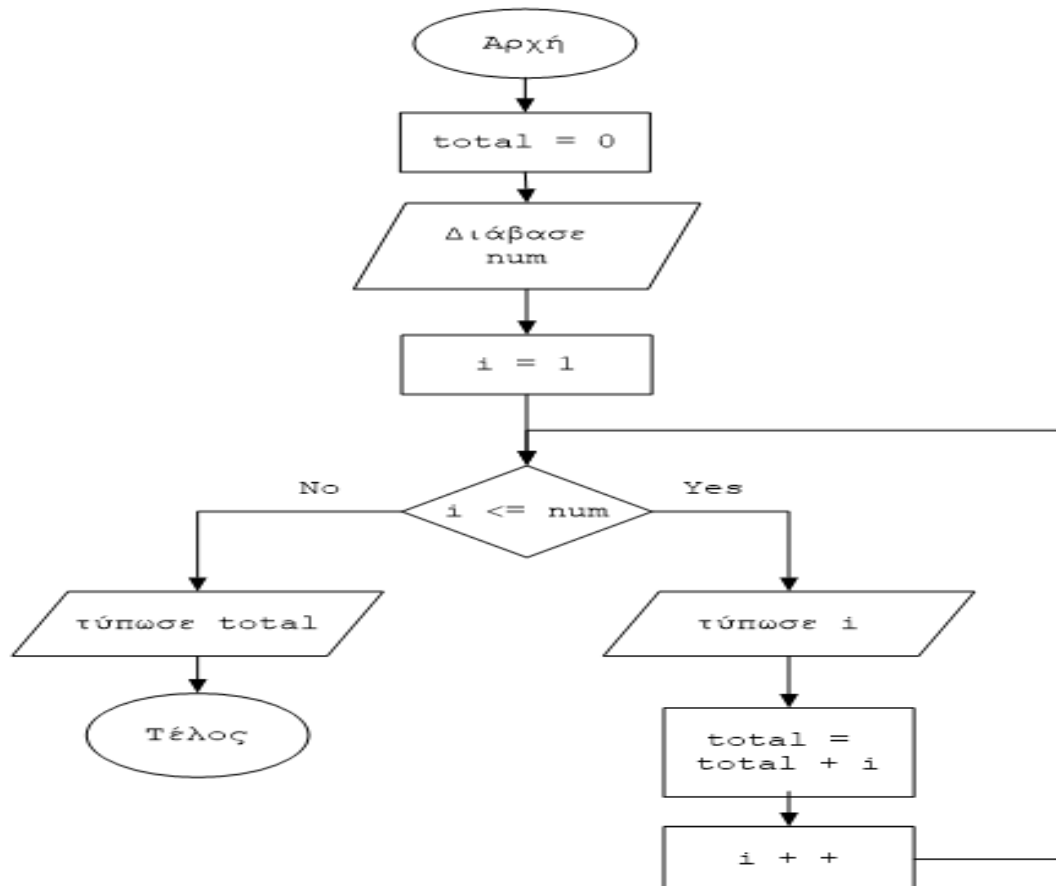


Απάντηση:



Άσκηση 3

Σας δίνεται το πιο κάτω λογικό διάγραμμα.



Ερώτημα (α) (Βαθμοί 1)

Ποιο θα είναι το αποτέλεσμα αν δοθεί αρχικά η τιμή για το **num=4**; Να σχεδιάσετε τον πίνακα προκαταρκτικής εκτέλεσης.

total	num	i	i<=num	T/F	Παρουσίαση
0	4	1	1<=4	T	1
1		2	2<=4	T	2
3		3	3<=4	T	3
6		4	4<=4	T	4
10		5	5<=4	F	10

Ερώτημα (β) (Βαθμοί 2)

Αν δοθεί αρχικά και πάλι η τιμή για το **num=4** και επιπρόσθετα διαγράψουμε την εντολή `i++`, θα δημιουργηθεί ένα ανεπιθύμητο πρόβλημα. Ποιο θα είναι το πρόβλημα αυτό; Να εξηγήσετε.

Ατέλειωτη δομή επανάληψης. Το loop δεν θα τελειώσει ποτέ.

Ερώτημα (γ) (Βαθμοί 2)

Το πιο πάνω λογικό διάγραμμα έχει μεταφραστεί στο πιο κάτω πρόγραμμα. Στο πρόγραμμα αυτό υπάρχουν λάθη. Να εντοπίσετε δύο (2) από τα λάθη και να τα διορθώσετε, αναφέροντας τον αριθμό της γραμμής στην οποία εμφανίζεται το κάθε λάθος μαζί με την διορθωμένη εντολή.

```
[1] #include <iostream>
[2] using namespace std;
[3] int main(){
[4] int i,total;
[5] total=0;
[6] cout<<"Please insert a number: ";
[7] cin>>num;
[8] for (i=1;i<=num;i++)
[9] cout<<i<<" ";
[10] total=i;}
[11] cout<<endl<<"The total is: "<<total<<endl;
[12] return 0;}
```

Γραμμή [4] - να δηλωθεί η μεταβλητή num

Γραμμή [8] - δεν υπάρχει αγκύλη { για να ανοίξει το loop

Γραμμή [11] - endl αντί endl

Άσκηση 4

Το Λύκειο Λιβαδιών σε συνεργασία με τον Δήμο Λιβαδιών, αποφάσισε να λειτουργήσει την βιβλιοθήκη του και ως προς το κοινό, πέραν από τους μαθητές του σχολείου. Για τον σκοπό αυτό θα πρέπει να δημιουργηθεί ένα ηλεκτρονικό σύστημα. Μια ομάδα μαθητών του λυκείου θα αναλάβει να διεκπεραιώσει το σύστημα αυτό. Το σύστημα αυτό θα περιέχει τις πληροφορίες των επισκεπτών της βιβλιοθήκης, των βιβλίων αλλά και της διαδικασίας δανεισμού ενός βιβλίου. Συγκεκριμένα ο επισκέπτης θα κάνει την εγγραφή του στην βιβλιοθήκη παρέχοντας τα προσωπικά του στοιχεία και θα του δίνεται ένας κωδικός. Αν θέλει να δανειστεί βιβλία θα δίνει τον κωδικό του και τα βιβλία που αποφάσισε να δανειστεί και με τη διαδικασία δανεισμού τα βιβλία θα χρεώνονται στον λογαριασμό του και θα πρέπει να τα επιστρέψει σε μια συγκεκριμένη μέρα.

Ερώτημα (α) (Βαθμοί 1)

Έχετε αναλάβει τον ρόλο του αναλυτή στην ομάδα δημιουργίας του συστήματος αυτού. Να αναφέρετε δύο (2) από τα προσόντα που πρέπει να έχετε ως αναλυτής συστημάτων για να μπορέσετε να φέρετε εις πέρας με επιτυχία τα καθήκοντα σας.

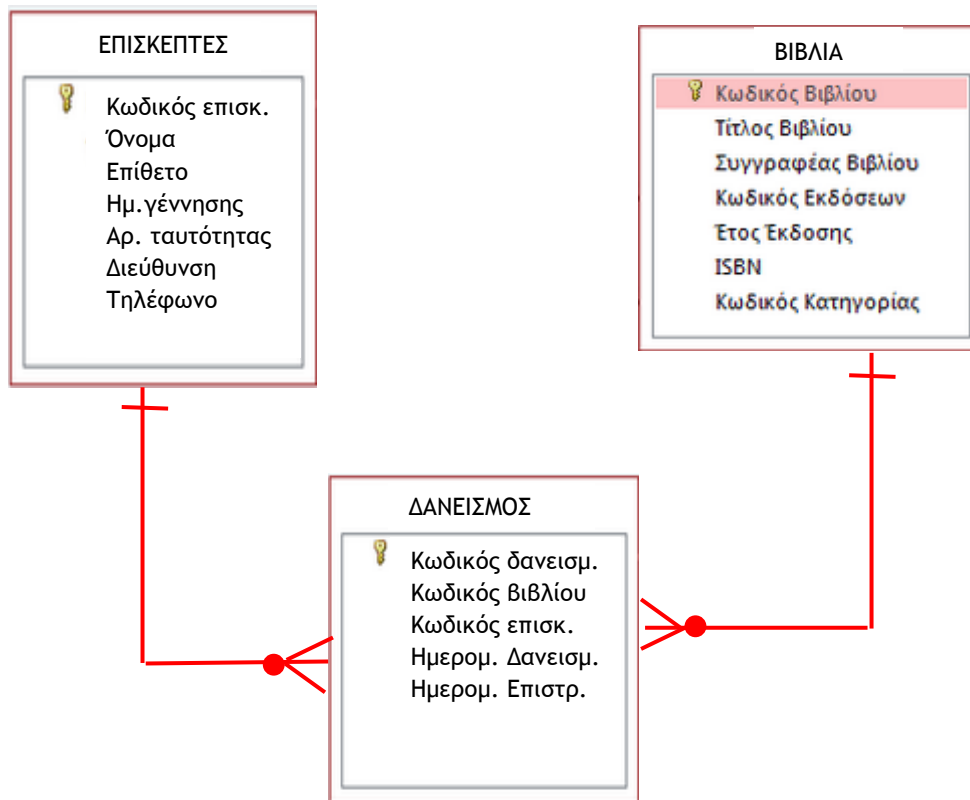
1. Γνώση πληροφοριακών συστημάτων και τεχνολογίας
2. Προγραμματιστική εμπειρία
3. Γενική γνώση επιχειρήσεων
4. Ικανότητα επίλυσης προβλημάτων
5. Διαπροσωπικές ικανότητες επικοινωνίας
6. Διαπροσωπικές ικανότητες σχέσεων
7. Διαλλακτικότητα και προσαρμογή
8. Χαρακτήρας και ηθική
9. Ικανότητες ανάλυσης και σχεδίασης συστημάτων

(2 οποιαδήποτε από τα πιο πάνω προσόντα)

Ερώτημα (β) (Βαθμοί 2)

Να σχεδιάσετε τις σχέσεις μεταξύ των τριών (3) πινάκων του συστήματος που εμφανίζονται πιο κάτω και να σημειώσετε την πληθυκότητα (cardinality). Να λάβετε υπόψη τα ακόλουθα:

- Ένας επισκέπτης δεν είναι υποχρεωμένος να δανειστεί κάποιο βιβλίο αλλά μπορεί να δανειστεί πολλά ταυτόχρονα.
- Υπάρχει περίπτωση ένα βιβλίο να μην δανειστεί από κανέναν μαθητή.



Ερώτημα (γ) (Βαθμοί 2)

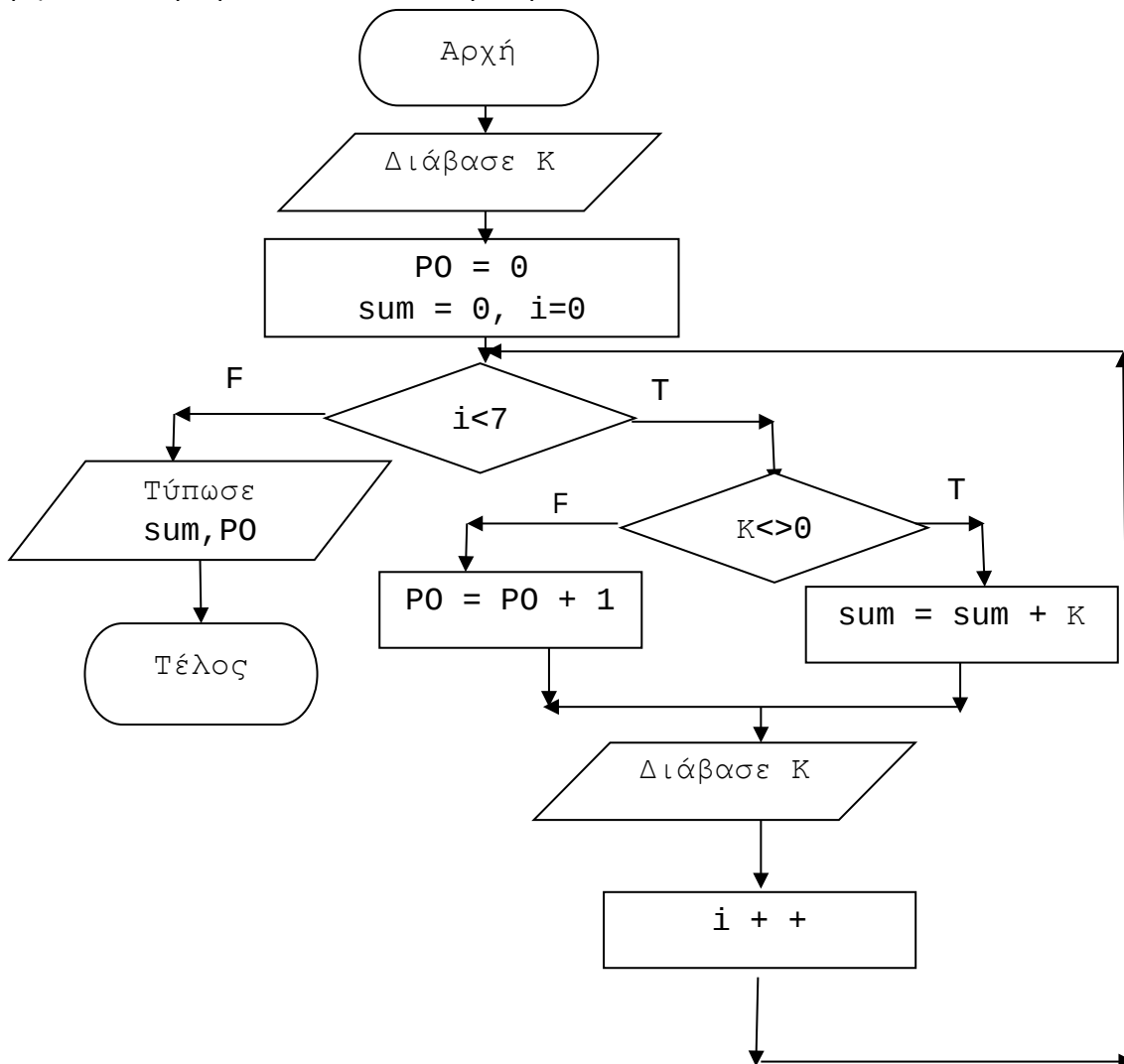
Να δημιουργήσετε ερώτημα, το οποίο να εμφανίζει το κωδικό βιβλίου, τον τίτλο βιβλίου, το έτος έκδοσης και τον κωδικό κατηγορίας, των βιβλίων που ανήκουν στην συγγραφέα «**Λένα Μαντά**». Τα αποτελέσματα να εμφανίζονται σε αύξουσα σειρά με βάση τον κωδικό βιβλίου.

Για το ερώτημα να σχεδιάσετε ένα πλέγμα όπως το πιο κάτω, όπου στο πάνω μέρος θα εμφανίζονται οι πίνακες από τους οποίους αντλείτε τα δεδομένα και οι σχέσεις μεταξύ τους. Στο κάτω μέρος να εμφανίζονται το όνομα του πίνακα, το όνομα του κάθε πεδίου, τα κατάλληλα κριτήρια και το είδος της ταξινόμησης.

Field:	Κωδικός βιβλίου	Τίτλος βιβλίου	Συγγραφέας βιβλίου	Έτος έκδοσης	Κωδικός κατηγορίας
Table:	βιβλία	βιβλία	βιβλία	βιβλία	βιβλία
Sort:	ascending				
Show:	☒	☒	☐	☒	☒
Criteria:			"Λένα Μαντά"		
or:					

Άσκηση 5

Σας δίνεται το πιο κάτω λογικό διάγραμμα. Το πρόγραμμα θα διαβάσει 7 ακέραιους αριθμούς και αν αυτοί είναι άνισοι του μηδενός τότε θα υπολογίζει το άθροισμα τους, διαφορετικά θα υπολογίζει το πλήθος των αριθμών που είναι μηδέν. Στο τέλος, θα τυπώνει το πλήθος των μηδενικών αριθμών αλλά και το άθροισμα.



Ερώτημα (α) (Βαθμοί 1)

Να γράψετε την εντολή ή τις εντολές δήλωσης των απαραίτητων μεταβλητών.

```
#include <iostream>
using namespace std;
int K, P0=0, sum=0, i=0;
```

Ερώτημα (β) (Βαθμοί 2)

Να γράψετε τις μεταβλητές και τις εντολές που χρειάζεστε, ώστε το πρόγραμμα, εκτός από αυτά που φένεται να κάνει στο λογικό διάγραμμα, να υπολογίζει και να τυπώνει **και** το πλήθος των αριθμών που είναι άνισοι του μηδενός, ξεκινώντας από την εντολή:

```
cin>> K;
```

```

int plithos=0;
cin>>K;
while (i<7){
    if (K<>0){
        sum=sum+K;
        plithos=plithos+1;
    }
    else
        P0=P0+1;
cin>>K;
}
cout<<sum<<endl;
cout<<P0<<endl;
cout<<plithos;
return 0;
}

```

Ερώτημα (γ) (Βαθμοί 2)

Να δηλώσετε την απαραίτητη βιβλιοθήκη και να τροποποιήσετε το πρόγραμμά σας έτσι ώστε να υπολογίζει τον μέσο όρο των αριθμών που είναι άνισοι του μηδενός και να τον τυπώνει στο τέλος, με ακρίβεια δύο (2) δεκαδικών ψηφίων.

```

#include <iomanip>
int plithos=0;
float mo;
cin>>K;
while (i<7){
    if (K<>0){
        sum=sum+K;
        plithos=plithos+1;
    }
    else
        P0=P0+1;
cin>>K;
}
mo=float(sum/7);
cout<<sum<<endl;
cout<<P0<<endl;
cout<<plithos<<endl;
cout<<fixed<<setprecision(2)<<mo;
return 0;
}

```

Άσκηση 6

Ένα σχολείο θα πραγματοποιήσει εκδρομή για τους μαθητές του και για την μεταφορά τους χρειάζεται να ενοικιάσει λεωφορεία. Η εταιρεία λεωφορείων διαθέτει λεωφορεία των 50 θέσεων. Το πρόγραμμα θα διαβάζει το **πλήθος των ατόμων** που θα πάνε εκδρομή και θα υπολογίζει και θα τυπώνει το **πλήθος των λεωφορείων** που θα χρειαστεί το σχολείο για να πραγματοποιήσει την εκδρομή.

Σας δίνεται το πιο κάτω πρόγραμμα:

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){
    int students, NoOfBus;
    cout<<"Δώσε αριθμό μαθητών: "<<endl;
    cin >> students;
    NoOfBus = students/50.00;
    cout<<"Συνολικός αριθμός λεωφορείων: "<< NoOfBus;
    return 0;
}
```

Ερώτημα (α) (Βαθμοί 1)

Το πρόγραμμα έχει δοκιμαστεί για αριθμό μαθητών 135 και παρόλο που η μαθηματική πράξη είναι σωστή, εντούτοις το πρόγραμμα δεν εμφανίζει τον σωστό αριθμό λεωφορείων (αντί να εμφανίζει 3, εμφανίζει 2. Να αλλάξετε την απαραίτητη εντολή, έτσι ώστε να διορθωθεί το πρόβλημα αυτό.

NoOfBus = round(students/50.00);

Ερώτημα (β) (Βαθμοί 2)

Να προσθέσετε τις κατάλληλες εντολές, ώστε να υπολογίζει και να τυπώνει το **πλήθος των μαθητών** που θα έχει το **τελευταίο λεωφορείο**.

```
#include <iostream>
#include <cmath>
using namespace std;
int main(){
    int students, studentsLast;
    float NoOfBus;
    cout<<"NO of students: "<<endl;
    cin >> students;
    NoOfBus = round(students/50.00);
    cout<<"Total No of Buses: "<<NoOfBus<<endl;
    studentsLast=students%50;
    cout<<"No of students in last bus: "<<studentsLast;
    return 0;}

```

Ερώτημα (γ) (Βαθμοί 2)

Να υπολογίσετε και να τυπώσετε το **συνολικό κόστος ενοικίασης** (σε 2 δεκαδικά ψηφία) αν η μίσθωση του κάθε λεωφορείου στοιχίζει **55 ευρώ**. Στην περίπτωση που το σχολείο μισθώσει πέραν των **10** λεωφορείων η εταιρεία προσφέρει **έκπτωση 30%**.

Παράδειγμα Εισόδου 1 357	Παράδειγμα Εξόδου 1 Δώσε αριθμό μαθητών: Συνολικός αριθμός λεωφορείων:7 Συνολικό Κόστος ενοικίασης:440.00
------------------------------------	---

```
#include <iostream>
#include <cmath>
#include<iomanip>
using namespace std;
int main(){
    int students, studentsLast;
    float NoOfBus, cost;
    cout<<"NO of students: "<<endl;
    cin >> students;
    NoOfBus = round(students/50.00);
    cout<<"Total No of Buses: "<<NoOfBus<<endl;
    studentsLast=students%50;
    cost=NoOfBus*55.00;
    if (NoOfBus>0)
        cost=cost-(cost*0.30);
    cout<<"No      of      students      in      last      bus:
"<<studentsLast<<endl;
    cout<<"Total Cost: "<<fixed<<setprecision(2)<<cost;
    return 0;}
```

ΤΕΛΟΣ Α΄ ΜΕΡΟΥΣ

ΜΕΡΟΣ Β΄

Άσκηση 1

Στο νηπιαγωγείο της γειτονιάς σας αποφάσισαν να παίζουν ένα νέο παιχνίδι. Το παιχνίδι έχει ως εξής: δύο (2) παιδιά αγωνίζονται σε ένα αγώνα δρόμου. Το πρώτο παιδάκι ξεκινά από το σημείο a και προχωρά με ένα άλμα x εκατοστόμετρα μπροστά κάθε φορά, ενώ το δεύτερο παιδάκι, ξεκινά από το σημείο b και προχωρά με κάθε άλμα y εκατοστόμετρα μπροστά κάθε φορά. Και τα δύο παιδάκια κάνουν άλματα στην ίδια ευθεία.

Να δημιουργήσετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο θα υλοποιεί τα πιο κάτω ερωτήματα.

Ερώτημα (α) (Βαθμοί 2)

Να δηλώσετε όλες τις μεταβλητές για το πιο πάνω πρόβλημα και να δώσετε τις κατάλληλες εντολές, οι οποίες να διαβάζουν τα δεδομένα εισόδου. Τα δεδομένα εισόδου είναι τέσσερις ακέραιοι αριθμοί: a , x , b και y .

```
#include<iostream>
#include<cmath>
using namespace std;
int main(){
    int a,b,x,y,pA=1,d,dA,dB,jumps=0;
    cin >> a >> x >> b >> y;
```

Ερώτημα (β) (Βαθμοί 2)

Μετά από πόσα άλματα το πρώτο παιδάκι θα διανύσει απόσταση μεγαλύτερη των δέκα (10) μέτρων από το σημείο εκκίνησης.

```
pA=1000 / x+1;
cout<<"Kid A makes "<<pA<<" jumps"<<endl;
```

Ερώτημα (γ) (Βαθμοί 3)

Πόση απόσταση θα έχουν τα δύο (2) παιδάκια μεταξύ τους μετά από πενήντα (50) άλματα. Ο αριθμός αυτός να μην εμφανίζεται με αρνητικό πρόσημο.

```
d=(50*x+a) - (50*y+b);
cout << "Distance after 50 jumps: "<<abs(d)<<endl;
```

Ερώτημα (δ) (Βαθμοί 3)

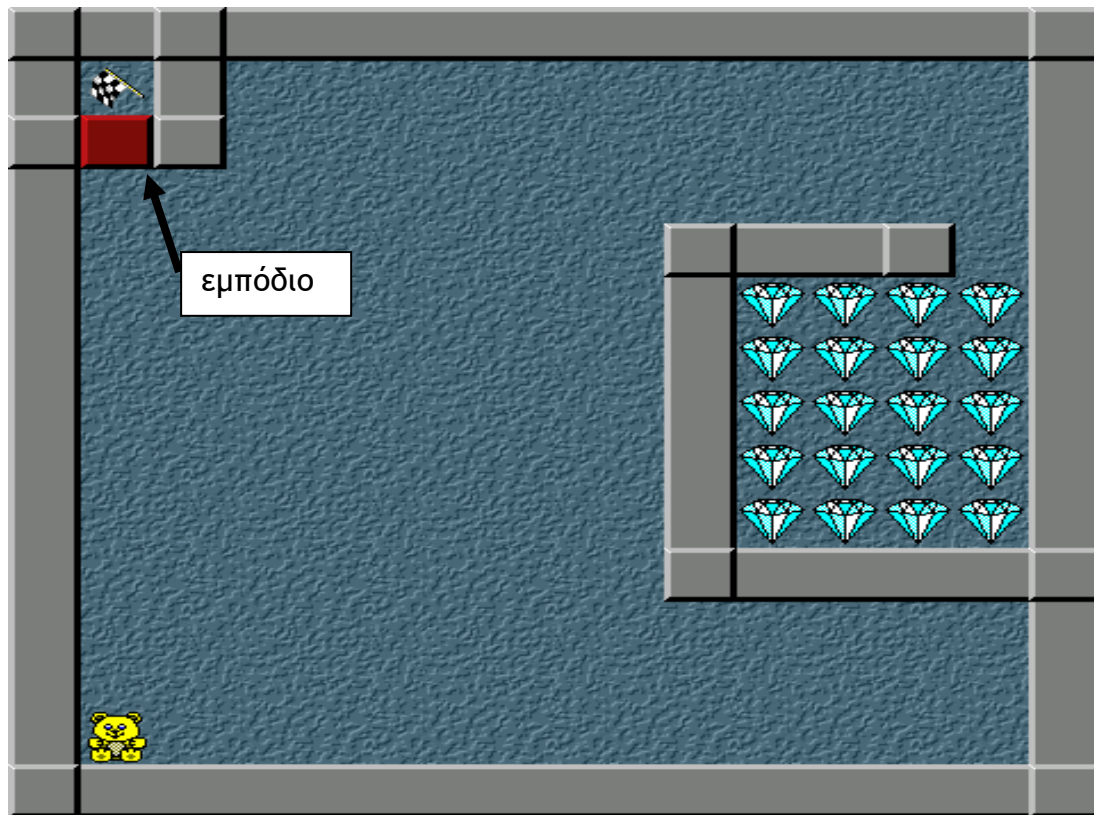
Ας υποθέσουμε ότι τα παιδιά μετά τον ίδιο αριθμό αλμάτων, μπορούν να προσγειωθούν στο ίδιο σημείο. Να υλοποιήσετε δομή επανάληψης, η οποία να βρίσκει τον αριθμό αυτό, ή αν δεν μπορέσουν να συναντηθούν μετά από 100 (εκατόν) άλματα, η δομή να τερματίζει και να εμφανίζεται το μήνυμα "No meeting point".

```
dA=a;
dB=b;
while( (dA!=dB)&&(jumps<=100) ){
    dA+=x;
    dB+=y;
    jumps++;
}
if(jumps>100)
    cout << "No meeting point" << endl;
else
    cout << "They meet after " << jumps << " jumps" <<
endl;
return 0;
}
```

Παράδειγμα Εισόδου 1 3 6 7 5	Παράδειγμα Εξόδου 1 Kid A makes 167 jumps Distance after 50 jumps: 46 They meet after 4 jumps
Παράδειγμα Εισόδου 2 2 5 10 12	Παράδειγμα Εξόδου 2 kid A makes 201 jumps Distance after 50 jumps: 358 No meeting point

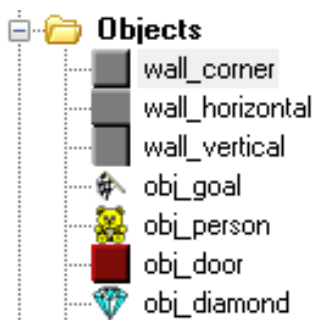
Άσκηση 2

Έχετε σχεδιάσει το παρακάτω παιχνίδι. Ο κεντρικός σας ήρωας είναι αρκουδάκι. Στόχος του παιχνιδιού είναι φτάσει το αρκουδάκι στην καρό σημαία. Υπάρχει ένα κόκκινο αντικείμενο-εμπόδιο, το οποίο θα εξαφανίζεται μόλις το αρκουδάκι μαζέψει όλα τα διαμάντια.



Ερώτημα (α) (Βαθμοί 2)

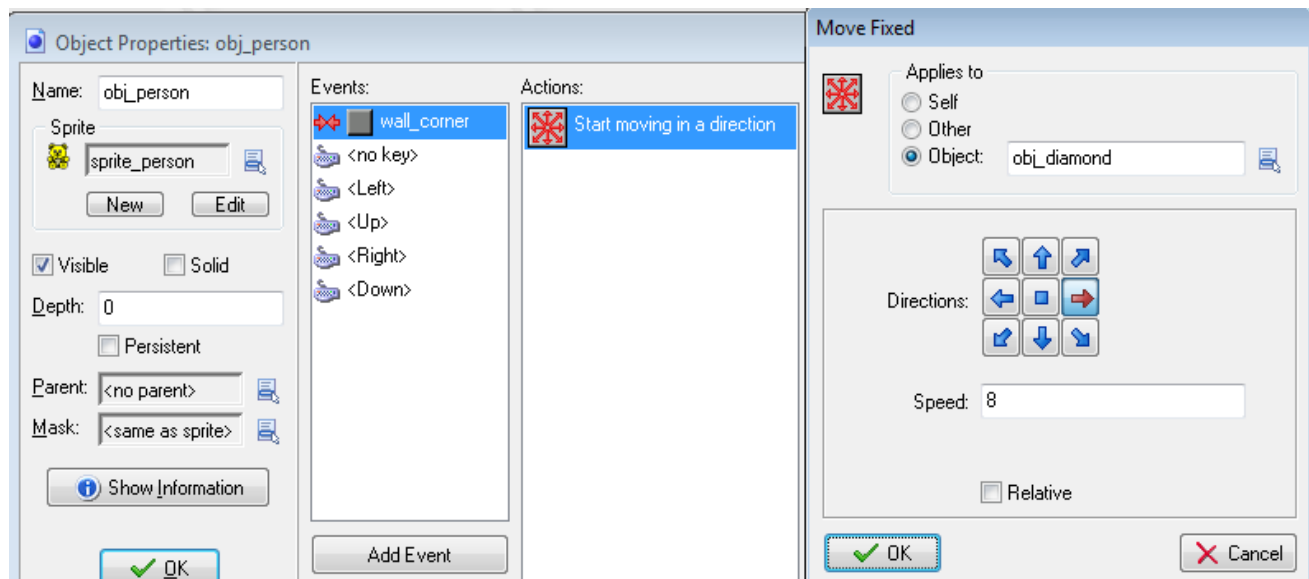
Ποια είναι τα αντικείμενα (objects), τα οποία χρειάζεται να δημιουργήσετε για το πιο πάνω παιχνίδι.



**Μπορούμε να δεχτούμε και το wall ως ένα object :
Wall, Goal, Person, door, diamond**

Ερώτημα (β) (Βαθμοί 2)

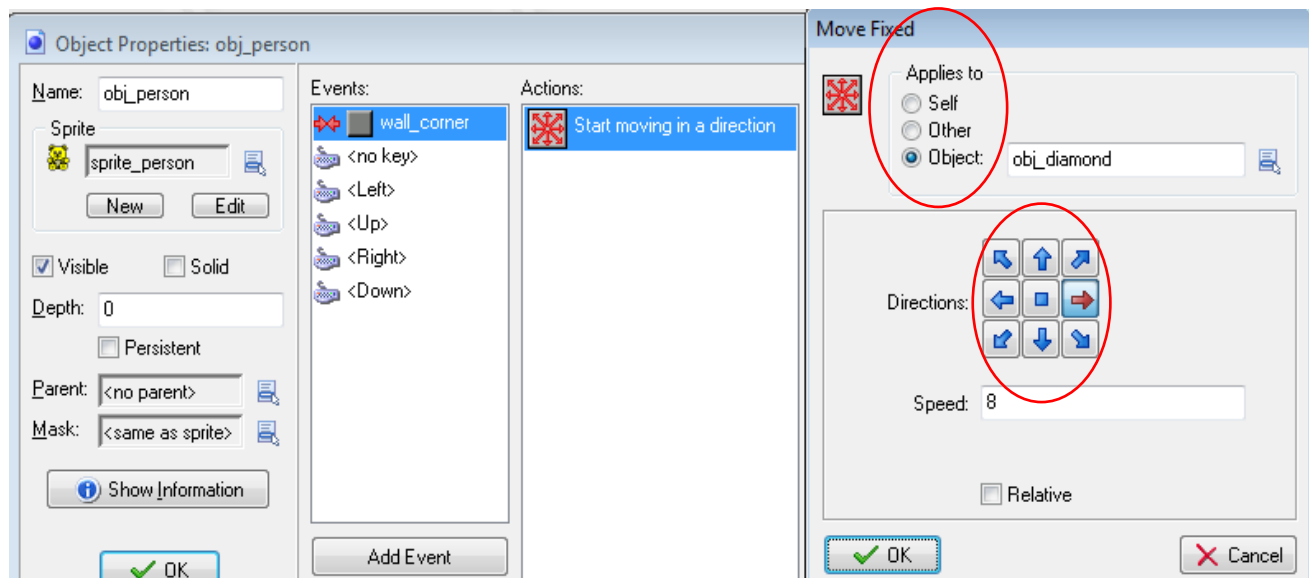
Τι αποτέλεσμα έχει ο παρακάτω προγραμματισμός;



Όταν το αρκουδάκι έχει collision με τον τοίχο τότε μετακινούνται τα διαμάντια δεξιά με ταχύτητα 8.

Ερώτημα (γ) (Βαθμοί 3)

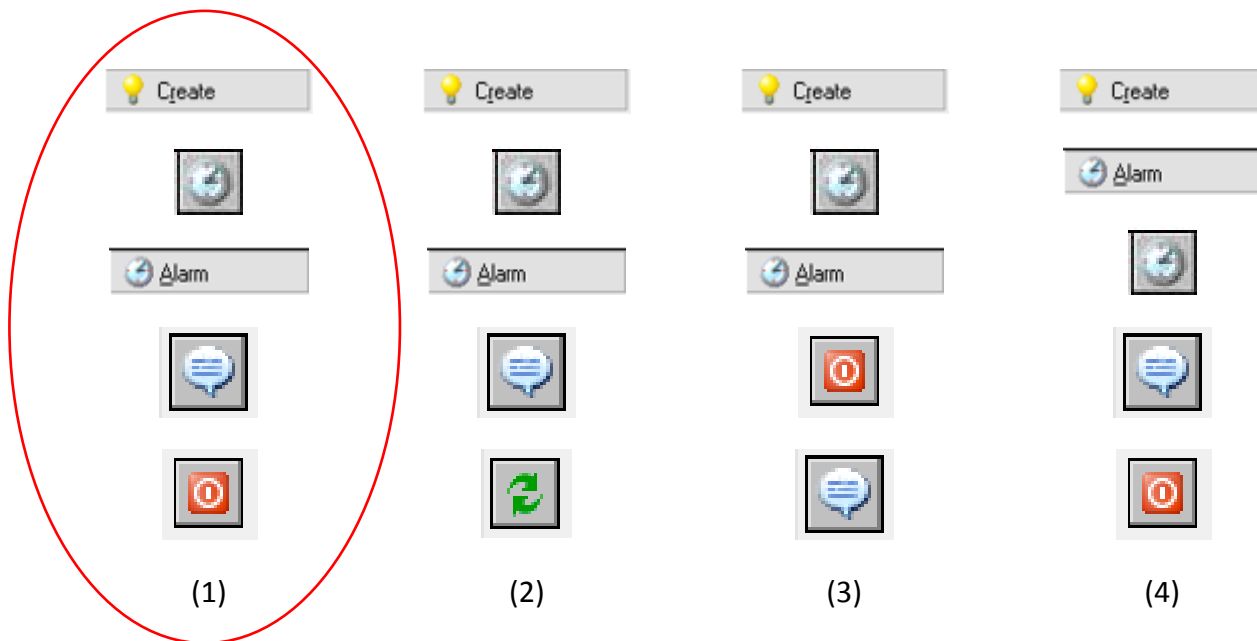
Όταν ο ήρωάς μας (το αρκουδάκι) συγκρούεται (έχει collision) με τον τοίχο θέλουμε να σταματά. Τι χρειάζεται να αλλάξετε στην παραπάνω οθόνη ώστε να έχετε αυτό το αποτέλεσμα;



Applies to: SELF και Directions: STOP

Ερώτημα (δ) (Βαθμοί 3)

Να σημειώσετε ποιες από τις πιο κάτω επιλογές των events και των actions θα πρέπει να διαλέξετε με την σωστή σειρά έτσι ώστε το παιχνίδι να τερματίζεται σε **3 λεπτά** από τη στιγμή που ξεκινά εμφανίζοντας το μήνυμα «**ΤΕΛΟΣ ΧΡΟΝΟΥ**».



Η επιλογή (1) .

Άσκηση 3

Το Gaming CLUB «Το Έξυπνο Παιχνίδι» διατηρεί μηχανογραφημένο σύστημα εξυπηρέτησης πελατών. Το σύστημα περιλαμβάνει δύο αρχεία, στο ένα βρίσκονται καταχωρημένα τα στοιχεία των πελατών (κωδικός πελάτη, όνομα, επίθετο, διεύθυνση, τηλέφωνο) και στο άλλο είναι καταχωρημένα τα παιχνίδια προς ενοικίαση. Στο δεύτερο αρχείο βρίσκονται καταχωρημένα τα πιο κάτω στοιχεία: κωδικός παιχνιδιού, είδος παιχνιδιού, τίτλος παιχνιδιού, κόστος, ηλικιακό επίπεδο και κωδικός πελάτη, ημερομηνία ενοικίασης.

Όταν παρουσιαστεί ένας νέος πελάτης στο κατάστημα, δίνει τα στοιχεία του για να εγγραφεί στο σύστημα. Τα στοιχεία αυτά μαζί με τον κωδικό του πελάτη που δημιουργείται αυτόματα από το σύστημα καταχωρούνται στο αρχείο πελατών. Αφού γίνει η καταχώρηση του πελάτη δημιουργείται η κάρτα μέλους η οποία περιέχει όλα τα πιο πάνω στοιχεία και παραδίδεται στο νέο πελάτη.

Όταν θελήσει να ενοικιάσει κάποιο παιχνίδι, δίνει την κάρτα μέλους του και τον τίτλο του παιχνιδιού. Στη συνέχεια γίνεται έλεγχος για τη διαθεσιμότητα του παιχνιδιού από το αρχείο παιχνιδιών. Αν είναι διαθέσιμο, καταχωρείται στο αρχείο παιχνιδιών ο κωδικός πελάτη μαζί με τον κωδικό του συγκεκριμένου παιχνιδιού και την ημερομηνία ενοικίασης, η οποία δημιουργείται αυτόματα από το σύστημα. Στο τέλος κάθε μήνα εκτυπώνεται κατάλογος με τους πελάτες του καταστήματος οι οποίοι έχουν δανειστεί παιχνίδια με κόστος τουλάχιστον € 10 και παραδίδεται στον ιδιοκτήτη.

Ερώτημα (α) (Βαθμοί 2)

Να καθορίσετε τους πίνακες για το πιο πάνω σύστημα. Για κάθε πίνακα να καθορίσετε:

- το όνομα του
- το πρωτεύον κλειδί του
- τα πεδία και τον τύπο των πεδίων του κάθε πίνακα

Πίνακας Πελατών	
Κωδικός Πελάτη	Number
Όνομα	Text
Επίθετο	Text
Διεύθυνση	Text
Τηλέφωνο	Text

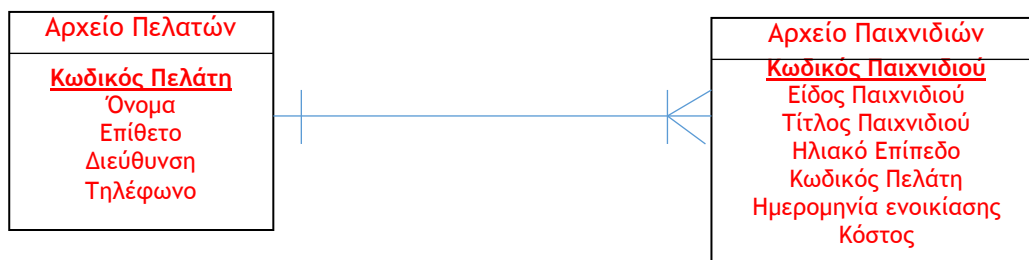
Πρωτεύον Κλειδί: Κωδικός Πελάτη

Πίνακας Παιχνιδιών	
Κωδικός Παιχνιδιού	Number
Είδος Παιχνιδιού	Text
Τίτλος Παιχνιδιού	Text
Ηλιακό Επίπεδο	Text
Κωδικός Πελάτη	Number
Ημερομηνία ενοικίασης	Date/Time
Κόστος	Currency

Πρωτεύον Κλειδί: Κωδικός Παιχνιδιού

Ερώτημα (β) (Βαθμοί 2)

Να σχεδιάσετε το Διάγραμμα Σχέσεων Οντοτήτων (ΔΣΟ) για το πιο πάνω σύστημα και να σημειώσετε την πληθυντικότητα (cardinality).



Ερώτημα (γ) (Βαθμοί 3)

Να σχεδιάσετε ερώτημα (query), το οποίο να βρίσκει τα δεδομένα για την εκτύπωση των πελατών του καταστήματος οι οποίοι έχουν δανειστεί παιχνίδια οι οποίοι έχουν δανειστεί παιχνίδια με κόστος τουλάχιστον € 10. Θα πρέπει να εμφανίζονται με τη σειρά όλα τα στοιχεία των πελατών, ο τίτλος του παιχνιδιού και η ημερομηνία ενοικίασης ταξινομημένα σε αύξουσα σειρά πρώτα με βάση το επίθετο και μετά με το όνομα.

Για το ερώτημα να σχεδιάσετε ένα πλέγμα όπως το πιο κάτω, όπου στο πάνω μέρος θα εμφανίζονται οι πίνακες από τους οποίους αντλείτε τα δεδομένα και οι σχέσεις μεταξύ τους. Στο κάτω μέρος να εμφανίζονται το όνομα του πίνακα, το όνομα του κάθε πεδίου, τα κατάλληλα κριτήρια και το είδος της ταξινόμησης.

Field:					
Table:					
Sort:					
Show:	☐	☐	☐	☐	☐
Criteria:					
or:					

Λύση :

Ερώτημα 3

Πελάτες

- * Κωδικός Πελάτη
- Όνομα
- Επίθετο
- Διεύθυνση
- Τηλέφωνο

Παιχνίδια

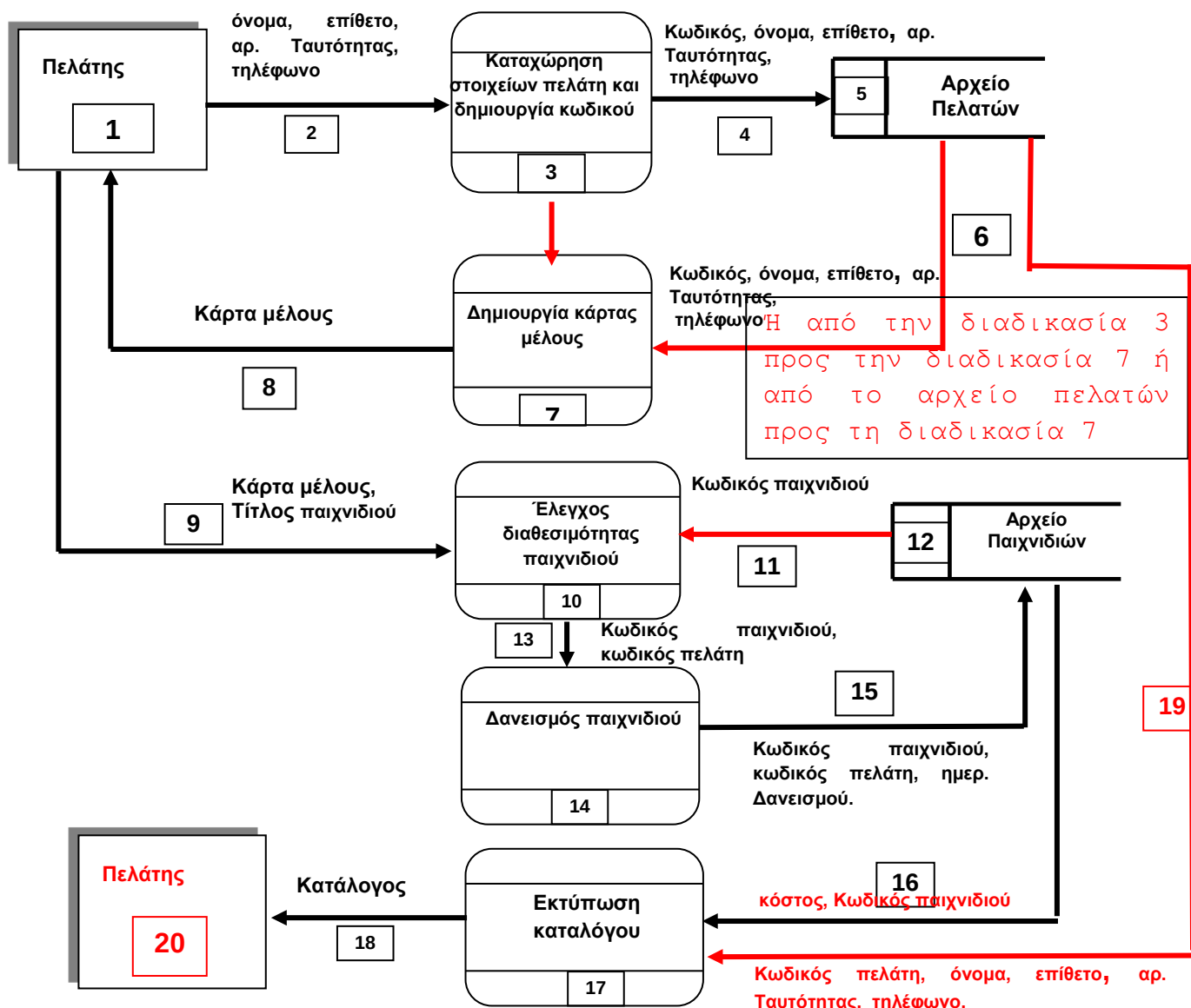
- * Κωδικός Παιχνιδιού
- Είδος Παιχνιδιού
- Τίτλος Παιχνιδιού
- Ηλικιακό Επίπεδο
- Κωδικός Πελάτη
- Ημερομηνία Ενοικίασης
- Κόστος

1 — ∞

Field:	Κωδικός Πελάτη	Όνομα	Επίθετο	Διεύθυνση	Τίτλος Παιχνιδιού	Ημερομηνία Ενοικίασης	Κόστος
Table:	Πελάτες	Πελάτες	Πελάτες	Πελάτες	Παιχνίδια	Παιχνίδια	Παιχνίδια
Sort:			Ascending				
Show:	☒	☒	☒	☒	☒	☒	☐
Criteria:							>=10
or:							

Ερώτημα (δ) (Βαθμοί 3)

Βάσει των απαιτήσεων του Gaming CLUB «Το Έξυπνο Παιχνίδι», ο βοηθός σας σχεδίασε το πιο κάτω ΔΡΔ, το οποίο όμως περιέχει τουλάχιστον τέσσερα (4) λάθη. Να τα εντοπίσετε και να τα διορθώσετε. Να ξανασχεδιάσετε το ΔΡΔ με διορθωμένα τα λάθη.



Άσκηση 4

Ο κύριος Παναγιώτης διατηρεί εδώ και τριάντα χρόνια μια επιχείρηση, η οποία ασχολείται με την παροχή οικιακών υπηρεσιών. Η εταιρεία αποτελείται από 20 υπαλλήλους. Δυστυχώς τα τελευταία χρόνια δεν πάει και τόσο καλά οικονομικά η επιχείρηση του κυρίου Παναγιώτη. Για να μην χρειαστεί να απολύσει κανέναν από τους υπαλλήλους του, αποφάσισε να κάνει περικοπές στους μισθούς τους, για κάποιο χρονικό διάστημα.

Να δημιουργήσετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο θα υλοποιεί τα πιο κάτω ερωτήματα.

Ερώτημα (α) (Βαθμοί 3)

Να δηλώσετε όλες τις απαραίτητες μεταβλητές για το πιο πάνω πρόβλημα και να δώσετε τις κατάλληλες εντολές, οι οποίες να διαβάζουν τα δεδομένα εισόδου. Το πρόγραμμα θα διαβάσει τον αριθμό κοινωνικών ασφαλίσεων (**ΚΚ**) και τον μισθό (**salary**) για κάθε υπάλληλο.

```
#include <iostream>
Using namespace std;
int main (){
    float kk, salary;
    cout<<"ΚΚ:";
    cin<<kk;
    cout<<"Μισθος:";
    cin>>salary;
    while (kk!=777){
        cout<<"ΚΚ:";
        cin<<kk;
        cout<<"Μισθος:";
        cin>>salary;
    }
    return 0;}
```

Θα θεωρηθεί σωστό και αν χρησιμοποιήσουν και δομή επανάληψης **do while**

Ερώτημα (β) (Βαθμοί 3)

Να υπολογίσετε και να τυπώσετε τον μισθό με τον οποίο θα αμείβεται πλέον ο κάθε υπάλληλος με βάση τον πιο κάτω πίνακα. Οι αποκοπές θα είναι κλιμακωτές.

Μισθός (€)	Αποκοπές (%)
μέχρι 1500	0%
1501-2000	2%
2001-2500	3%
2501 και άνω	4%

```
float apokopes;
while (kk!=777){
    if (salary>=1501 && salary<=2000){
        apokopes=salary*0.02;
    }
    else if (salary>=2001 && salary<=2500){
        apokopes=salary*0.03;
    }
    else if (salary>=2501){
        apokopes=salary*0.04;
    }
    salary=salary-apokopes;
}
```

```

cout<<"New salary: "<<salary<<endl;
cout<<"KK:";
cin>>kk;
cout<<"Misthos:";
cin>>salary;
}

```

Ερώτημα (γ) (Βαθμοί 2)

Να υπολογίσετε και να τυπώσετε το συνολικό ποσό αποκοπών ($\text{Tapokopes} = \text{misthos} * \text{apokopes}$), καθώς και τον μέσο όρο (MOapokopes).

```

float tapokopes, moapokopes;
int plithosM=0;
cout<<"KK:";
cin>>kk;
cout<<"Misthos:";
cin>>salary;
while (kk!=777){
    if (salary>=1501 && salary<=2000){
        apokopes=salary*0.02;
        plithosM++;
    }
    else if (salary>=2001 && salary<=2500){
        apokopes=salary*0.03;
        plithosM++;
    }
    else if (salary>=2501){
        apokopes=salary*0.04;
        plithosM++;
    }
    salary=salary-apokopes;
    tapokopes=tapokopes+apokopes;
    cout<<"New salary: "<<salary<<endl;
    cout<<"KK:";
    cin>>kk;
    cout<<"Misthos:";
    cin>>salary;

}
moapokopes=tapokopes/plithosM;
cout<<"MO: "<<moapokopes<<endl;

```

```

cout<<"total apokopes: "<<tapokopes<<endl;
cout<<"plithos ipallilon me meiosis >100: "<<plithos100;
return 0;
}

```

Ερώτημα (δ) (Βαθμοί 2)

Να υπολογίσετε και να τυπώσετε το πλήθος των υπαλλήλων που θα τύχουν μείωσης μισθού μεγαλύτερης από εκατό (100) ευρώ (>100).

```

int plithos100=0;
cout<<"KK:";
cin>>kk;
cout<<"Misthos:";
cin>>salary;
while (kk!=777){
    if (salary>=1501 && salary<=2000){
        apokopes=salary*0.02;
        plithosM++;
        if(apokopes>100)
            plithos100++;
    }
    else if (salary>=2001 && salary<=2500){
        apokopes=salary*0.03;
        plithosM++;
        if(apokopes>100)
            plithos100++;
    }
    else if (salary>=2501){
        apokopes=salary*0.04;
        plithosM++;
        if(apokopes>100)
            plithos100++;
    }
    salary=salary-apokopes;
    tapokopes=tapokopes+apokopes;
    cout<<"New salary: "<<salary<<endl;
    plithos++;
    cout<<"KK:";
    cin>>kk;
    cout<<"Misthos:";
    cin>>salary;
}

```

```

moapokopes=tapokopes/plithos;
cout<<"Μ0: "<<moapokopes<<endl;
cout<<"total apokopes: "<<tapokopes<<endl;
cout<<"plithos ipallilon me meiosis >100: "<<plithos100;
return 0;
}

```

Παράδειγμα Εισόδου	Παράδειγμα Εξόδου
3 854256 1326,85 123654 2005,63 524513 3045,45 777	Δώσε αριθμό υπαλλήλων: ΚΚ: Μισθός: ΚΚ: Μισθός: ΚΚ: Μισθός: ΚΚ: Υπάλληλος 1-νέος μισθός: 1326,85 Υπάλληλος 2-νέος μισθός: 1965,52 Υπάλληλος 3-νέος μισθός: 2954,09 Συνολικές αποκοπές: 131,47 Μέσος Όρος: 65,74 Πλήθος υπαλλήλων μείωσης >100: 0

ΤΕΛΟΣ Β' ΜΕΡΟΥΣ

ΜΕΡΟΣ Γ΄

Άσκηση 1

Να γράψετε πρόγραμμα στη γλώσσα προγραμματισμού C++ το οποίο να κάνει τα πιο κάτω:

Ερώτημα (α) (Βαθμοί 4)

Θα ζητά από το χρήστη τον αριθμό των μαθητών ενός σχολείου (NoSt) και ακολούθως θα ζητά για κάθε μαθητή: όνομα (name), κωδικό φύλου (Gender) (Α για αγόρι και Κ για κορίτσι), αριθμό δικαιολογημένων απουσιών (DA), αριθμό αδικαιολόγητων απουσιών (AA) και αριθμό απουσιών στο ΔΔΚ (DDK) (Δημιουργικότητα – Δράση – Κοινωνική Προσφορά). Η επανάληψη θα τερματίζει είτε όταν δοθούν τα στοιχεία για όλους του μαθητές του σχολείου, είτε όταν δοθεί όνομα μαθητή "EXIT".

Να θεωρήσετε ότι όλα τα στοιχεία, εκτός από τον κωδικό φύλου, δίνονται σωστά και δεν χρειάζεται κανένας έλεγχος. Στην περίπτωση του κωδικού φύλου πρέπει να γίνεται έλεγχος. Αν δοθεί λάθος (εκτός από Α ή Κ), να ενημερώνεται σχετικά ο χρήστης και να ζητά να δώσει ξανά τον κωδικό φύλου.

Ερώτημα (β) (Βαθμοί 3)

Θα βρίσκει και θα τυπώνει το όνομα του μαθητή που έχει τις περισσότερες απουσίες στο ΔΔΚ. Σε περίπτωση ύπαρξης δύο ή περισσότερων τέτοιων μαθητών, θα τυπώνει τον τελευταίο από αυτούς.

Ερώτημα (γ) (Βαθμοί 4)

Θα υπολογίζει και θα τυπώνει τον αριθμό των αγοριών (dangerB) και τον αριθμό των κοριτσιών (dangerG) που έχουν από 40 μέχρι 50 (συμπεριλαμβανομένων) αδικαιολόγητες απουσίες (AA) και κινδυνεύουν να μείνουν στάσιμοι.

Ερώτημα (δ) (Βαθμοί 4)

Θα υπολογίζει και θα τυπώνει τον αριθμό των μαθητών που έμειναν στάσιμοι, καθώς επίσης και το ποσοστό (%) αυτών των μαθητών σε σχέση με το σύνολο των μαθητών του σχολείου (NoSt). Στάσιμος μένει ο μαθητής ο οποίος έχει περισσότερες από 50 αδικαιολόγητες (AA) απουσίες ή περισσότερες από 160 απουσίες στο σύνολο (δικαιολογημένες και αδικαιολόγητες).

Άσκηση 2

Μια μεγάλη εμπορική εταιρεία χοντρικής πώλησης που προμηθεύει μεταπωλητές με εμπορεύματα, αποφάσισε να αυτοματοποιήσει το σύστημα παραγγελιών της για να παρέχει καλύτερη εξυπηρέτηση στους πελάτες της. Κατά τη φάση «Εξακρίβωση Αναγκών και Καθορισμός Απαιτήσεων» έχει εξακριβωθεί ότι το Πληροφοριακό Σύστημα πρέπει να περιλαμβάνει τις ακόλουθες λειτουργίες:

Για κάθε νέο πελάτη θα καταχωρούνται στο αρχείο πελατών τα εξής στοιχεία: ονοματεπώνυμο, όνομα εταιρείας, διεύθυνση, ηλεκτρονικό ταχυδρομείο, τηλέφωνο και ο κωδικός πελάτη που θα δημιουργείται αυτόματα από το σύστημα. Ο κωδικός πελάτη θα δίνεται και στον πελάτη.

Όταν ένας πελάτης θέλει να παραγγείλει εμπορεύματα, δίνει τον κωδικό του και τα εμπορεύματα που θέλει να παραγγείλει. Αφού πρώτα γίνει έλεγχος των στοιχείων του πελάτη από το σύστημα (ονοματεπώνυμο, όνομα εταιρείας), θα καταχωρείται η παραγγελία του στο αρχείο παραγγελιών. Η κάθε παραγγελία θα περιλαμβάνει τον κωδικό του πελάτη, ένα αριθμό παραγγελίας ο οποίος δημιουργείται αυτόματα από το σύστημα, την ημερομηνία της παραγγελίας (τρέχουσα ημερομηνία που καταχωρείται αυτόματα από το σύστημα), καθώς επίσης τα εμπορεύματα της παραγγελίας. Αμέσως μετά θα δημιουργείται μια βεβαίωση παραγγελίας που θα δίνεται στον πελάτη και η οποία θα περιλαμβάνει τον κωδικό και το όνομα της εταιρείας του πελάτη, τον αριθμό και την ημερομηνία της παραγγελίας και τα εμπορεύματα που παράγγειλε.

Όταν μια παραγγελία διεκπεραιωθεί, τότε ο υπεύθυνος της αποθήκης θα δώσει τον κωδικό του πελάτη και τον αριθμό της παραγγελίας καθώς επίσης και την ημερομηνία υλοποίησης της παραγγελίας, για να ενημερωθεί σχετικά το αρχείο παραγγελιών.

Στο τέλος κάθε ημέρας, θα δημιουργείται κατάλογος παραγγελιών που δεν έχουν ακόμη διεκπεραιωθεί. Ο κατάλογος θα περιλαμβάνει τον κωδικό και το όνομα της εταιρείας του κάθε πελάτη, καθώς επίσης τον αριθμό και την ημερομηνία της παραγγελίας τους. Ο κατάλογος θα δίνεται στον υπεύθυνο αποθήκης της εταιρείας.

Ερώτημα (α) (Βαθμοί 3)

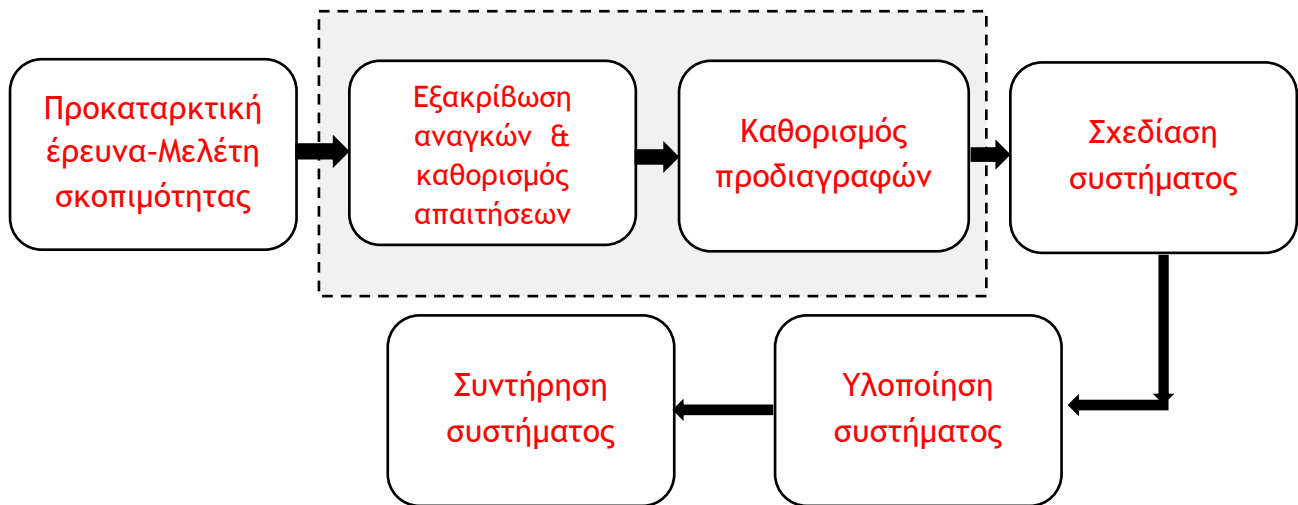
Να αναγνωρίσετε τις εξωτερικές οντότητες, τις διαδικασίες και τα αρχεία που περιγράφουν το πιο πάνω σενάριο.

Ερώτημα (β) (Βαθμοί 4)

Να σχεδιαστεί το Διάγραμμα Σχέσεων Οντοτήτων (ΔΣΟ, το οποίο να περιγράφει το πιο πάνω. Όπου κρίνεται αναγκαίο, να βάλετε επιπρόσθετα πεδία στους πίνακες, ώστε να μπορούν να συσχετιστούν μεταξύ τους.

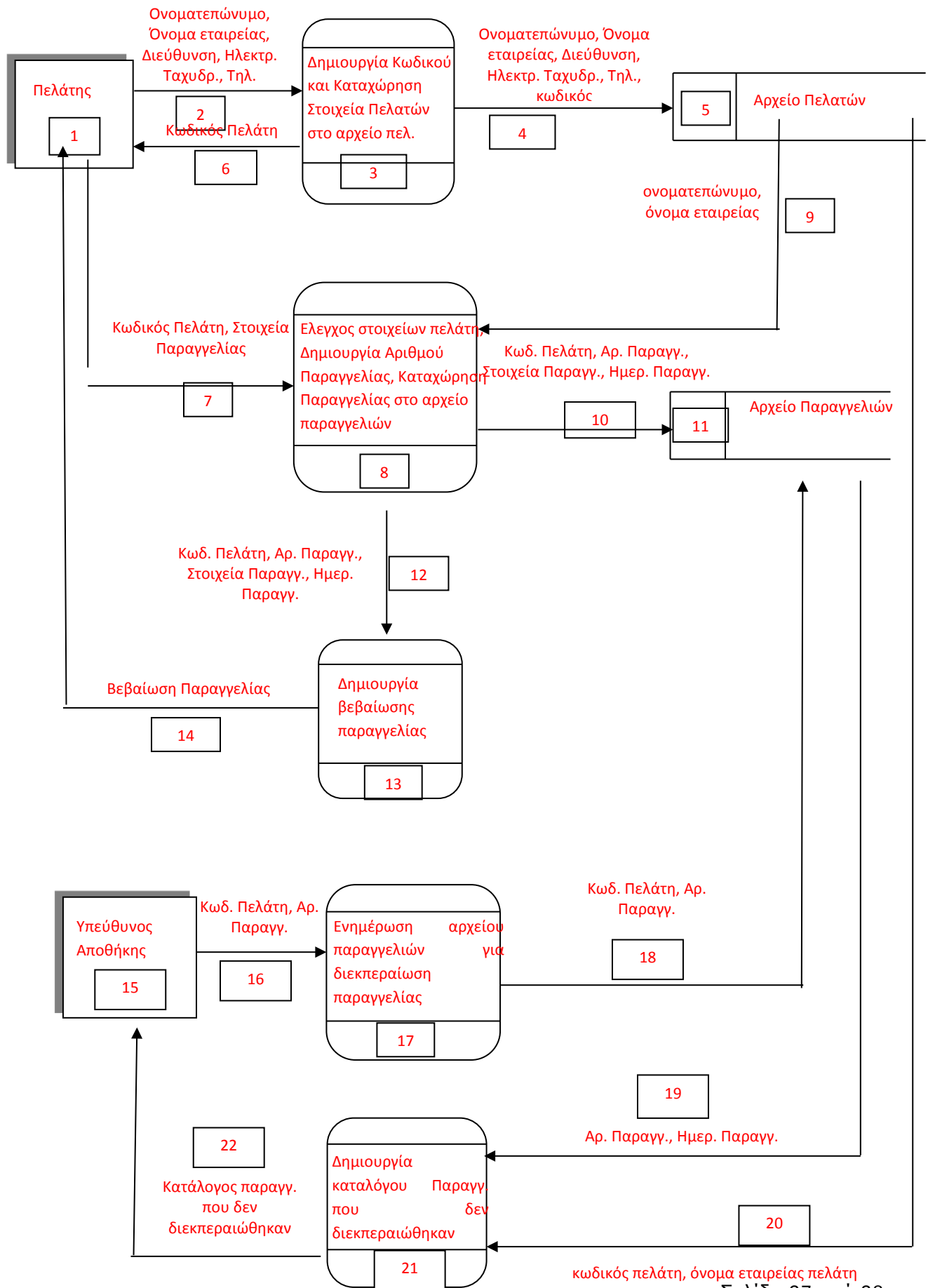
Ερώτημα (γ) (Βαθμοί 3)

Ποιες είναι κατά σειρά οι φάσεις που θα διέλθει το πληροφοριακό σύστημα από τη σύλληψη μέχρι την απόσυρσή του; Να συμπληρώσετε το πιο κάτω διάγραμμα.



Ερώτημα (δ) (Βαθμοί 5)

Να σχεδιάσετε το Διάγραμμα Ροής Δεδομένων (ΔΡΔ), το οποίο να περιγράφει το πιο πάνω.



Δεκτή λύση αν συγχωνευτούν οι διαδικασίες 8 και 13 σε μια διαδικασία.

ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ

Εισηγητές:

Βραχίμη Σουρουλλά Μαρία
Γιωργάλλας Παναγιώτης

Συντονιστής

Φιλιππίδης Αντώνης, Β.Δ.

Διευθυντής

Λοϊζίδης Πέτρος

ΛΥΣΕΙΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ

ΜΕΡΟΣ Α

Άσκηση 1

Ερώτημα (α)

- 1) #define pi 3.14;
ή
- 2) Const float pi=3.14;

Ερώτημα (β)

```
#include <cmath>
```

```
E=pi*pow(r,2);
```

Ερώτημα (γ)

```
#include <iomanip>
```

```
Float PER;
```

```
PER=2*pi*r;
```

```
Cout<<"PERIMETROS"<<fixed<<setprecision(2)<<PER;
```

Άσκηση 2

Ερώτημα (α)

1. $X=2*(\text{pow}(A,2)+\text{pow}(B,2));$
2. $Y=\text{sqrt}(3*(K-L));$

Ερώτημα (β)

1. 2
2. 9

Ερώτημα (γ)

1. 1001111
2. 39

Άσκηση 3

Ερώτημα (α)

Μεταβλητές		Συνθήκη		Παρουσίαση
Y	X	$X \leq 6$	T/F	
3	6	$6 \leq 6$	T	8 4
4	8	$8 \leq 6$	F	

Ερώτημα (β)

[5] cin>>y;

[7] while (x<=6)

Άσκηση 4

Ερώτημα (α)

```
#include <iostream>
using namespace std;
int main(){
    int j,s,p,i;
    j=2;
    s=0;
    p=1;
    i=1;
    while (i<=3){
        p=p*p+j;
        i=i+1;
        s=s+p;
    }
    cout<<s<<" "<<p;
    return 0;
}
```

Άσκηση 5

Ερώτημα (α)

<u>field</u>	cName	capital	cost
<u>criteria</u>			>300

Ερώτημα (β)

<u>field</u>	cName	capital	cost	river
<u>criteria</u>				Δούναβης
or				Σηκουάνας

Άσκηση 6

Ερώτημα (α)

Objects=8 Rooms=2

Ερώτημα (β)

Στο Applies to θα πρέπει να είναι επιλεγμένο το Self και όχι το Other

Ερώτημα (γ)

Collision

ΜΕΡΟΣ Β

Άσκηση 1

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;
int main(){
    int i, num, sumall,max,pl;
    string add,maxadd;
    sumall=0; max=0; pl=0;
    for (i=1;i<=6;i++){
        cout<<"dose diefthinsi kai arithmo H/Y"<<endl;
        cin>>add>>num;
        while (num<0 || num>100){
            cout<<"lathos arithmos"<<endl;
            cin>>num;}
        sumall=sumall+num;
        if (num>=max){
            max=num;
            maxadd=add;}
        if (num>70)
            pl=pl+1;}
    cout<<"sinolo H/Y"<<fixed<<setprecision(2)<<sumall<<endl;
    cout<<"diefthinsi me perissotera"<<maxadd<<endl;
    cout<<"apothikes me >70"<<pl;
    return 0;}
```

Άσκηση 2

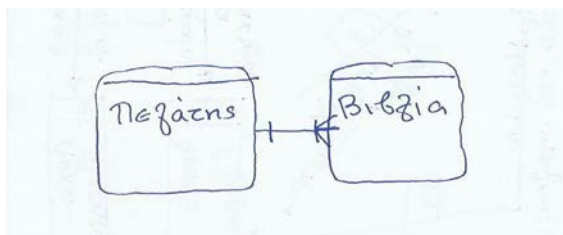
```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string onoma;
    float poso,sum,sum1,mo,mo1;
    int pl,pl1;
    sum=0;
    sum1=0;
    pl=0;
    pl1=0;
    cout<<"dose poso prosforas"<<endl;
    cin>>poso;
    while (poso>0){
        cout<<"dose onoma"<<endl;
        cin>>onoma;
        pl=pl+1;
        sum=sum+poso;
        if (poso>10000){
            sum1=sum1+poso;
            pl1=pl1+1;}
        cout<<"dose poso prosforas"<<endl;
        cin>>poso;}
    mo=sum/pl;
    mo1=sum1/pl1;
    cout<<"plithos prosforon"<<pl<<endl;
    cout<<"mesos oros olon"<<mo<<endl;
    cout<<"mesos oros prosforon>10000"<<mo1;
    return 0;}
```

Άσκηση 3

Ερώτημα (α)

1. Στη ροή 2 λείπει το τηλέφωνο.
2. Έπρεπε να υπάρχει μεταξύ του 1 και του 3 μια διαδικασία για την καταχώρηση των στοιχείων του πελάτη
3. Η ροή 4 έπρεπε να ξεκινά από την διαδικασία της καταχώρησης(Δες λάθος 2)
4. Στη ροή 7 έπρεπε να υπάρχει ο τίτλος του βιβλίου και το όνομα του συγγραφέα
5. Η ροή 9 έπρεπε να ξεκινά από το 10 προς το 8
6. Στη ροή 11 έπρεπε να υπάρχει ο κωδικός του βιβλίου
7. Η ροή 15 έπρεπε να ξεκινά από το 3 και όχι από το 10

Ερώτημα (β)



Άσκηση 4

Ερώτημα (α)

```
int main(){  
    int i,x;  
    i=1;  
    while (i<=10){  
        x=x+i;  
        i++;}  
    cout<<x;  
    return 0;}
```

Ερώτημα (β)

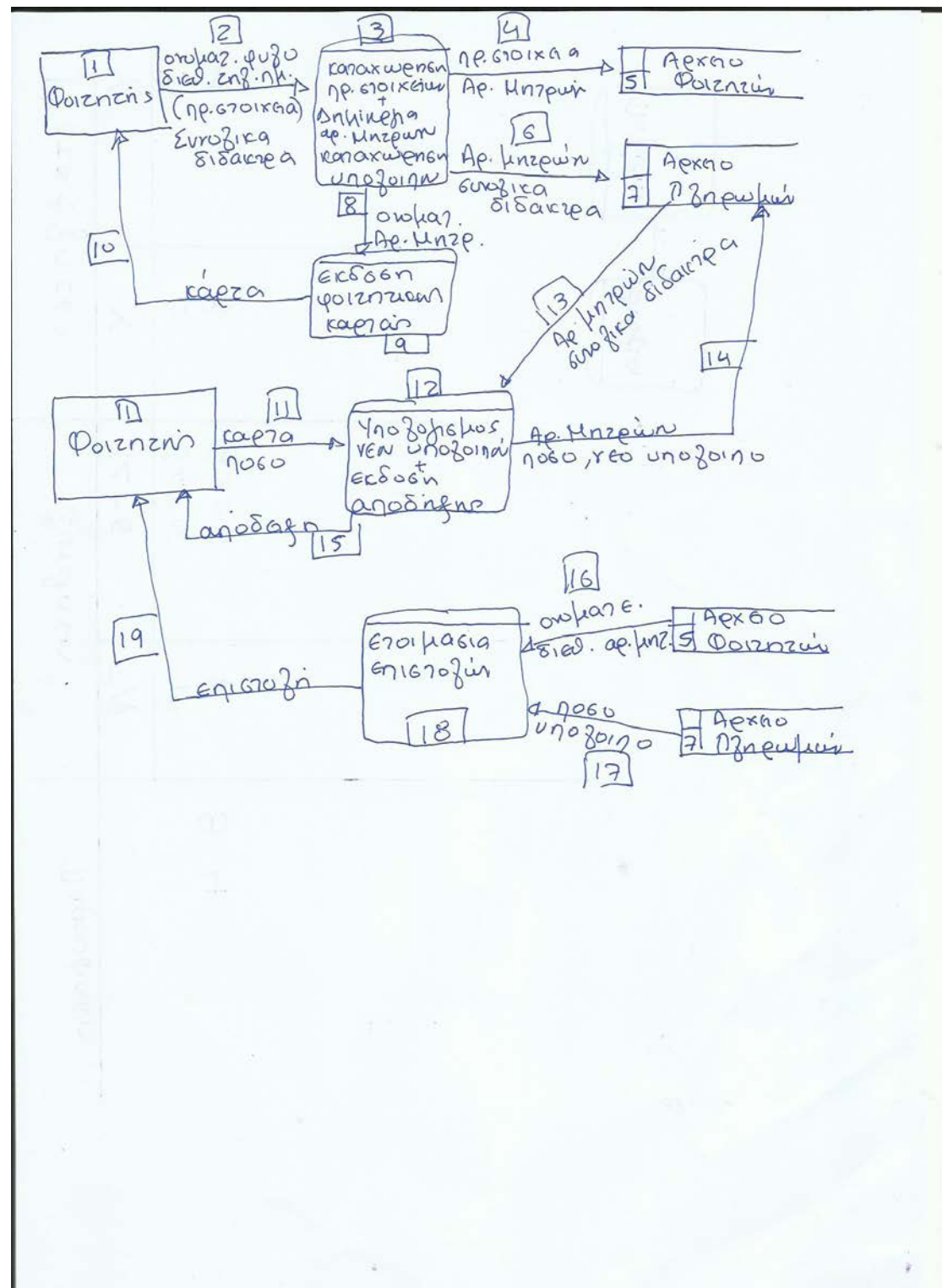
```
int main(){  
    int i,num,sum;  
    sum=0;  
    for (i=100; i>10; i--){  
        cin>>num;  
        sum=sum+num;}  
    cout<<sum;  
    return 0;}
```

ΜΕΡΟΣ Γ

Άσκηση 1

```
#include <iostream>
#include <string>
using namespace std;
int main(){
    string names[10],maxname;
    int B1[10],B2[10],i,pl;
    float total[10],max,sum,mo;
    sum=0;
    for (i=0; i<10; i++){
        cout<<"dose onoma kai tiw 2 bathmologies"<<endl;
        cin>>names[i]>>B1[i]>>B2[i];
        total[i]=(float)(B1[i]+B2[i])/2;
        sum=sum+total[i];
    }
    mo=sum/10;
    max=total[0];
    for (i=0; i<10; i++)
        if (total[i]>=max)
            {max=total[i];
            maxname=names[i];}
    cout<<"to onoma me ti megaliteri teliki"<<maxname<<endl;
    pl=0;
    for (i=0; i<10; i++)
        if (total[i]>mo)
            pl=pl+1;
    cout<<"plithos me bathmologia>mo"<<pl;
    return 0;}
```

Άσκηση 2



ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΜΑΙΟΥ-ΙΟΥΝΙΟΥ 2018

Μάθημα: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
Τάξη: Β΄

ΛΥΣΕΙΣΜΕΡΟΣ Α΄**Άσκηση 1**

Δείκτες Επιτυχίας
B7.1.5 Δηλώνουν μεταβλητές και σταθερές. B7.1.8 Εισαγάγουν βασικές εντολές εξόδου. B7.1.9 Εισαγάγουν εντολή εκχώρησης τιμής. B7.2.1 Αναγνωρίζουν την ανάγκη χρήσης μεταβλητών σε ένα πρόγραμμα. B7.2.2 Εντοπίζουν ποιες μεταβλητές (Variables) χρειάζονται με βάση τις ανάγκες του αλγορίθμου/προγράμματος. B7.2.7 Χρησιμοποιούν μεταβλητές (Variables) και σταθερές (Constants) με βάση τις ανάγκες του αλγορίθμου/προγράμματος. B7.2.9 Χρησιμοποιούν την εντολή εξόδου (π.χ. cout)

Λύση**Ερώτημα (α) 1μ**

```
telikos= (tetrA+tetrB)/2;// telikos= (float)(tetrA+tetrB)/2;
```

Ερώτημα (β) 2μ

```
#include<cmath>  
cout<<"Η teliki vathmologia einai: "<<round(telikos)<<endl
```

Ερώτημα (γ) 2μ

```
int athrisma;  
athrisma=tetrA + tetrB;  
cout<<"Το athrisma einai ton dyo tetraminon einai:"<<athrisma<<endl;
```

Άσκηση 2

Δείκτες Επιτυχίας

B7.3.5 Αναγνωρίζουν τους περιορισμούς των τύπων δεδομένων και τη διαδικασία μετατροπής τους από έναν τύπο σε άλλο, όταν χρησιμοποιούνται ταυτόχρονα σε μια έκφραση.

B7.3.6 Διατυπώνουν εκφράσεις/παραστάσεις οι οποίες περιλαμβάνουν δεδομένα με διαφορετικούς τύπους και να προβλέπουν το αποτέλεσμα.

B7.5.1 Αναγνωρίζουν τους συγκριτικούς τελεστές και να αναφέρουν λεκτικά τι αναπαριστά ο καθένας.

B7.5.5 Προβλέπουν το αποτέλεσμα (true ή false) μιας συνθήκης με μέχρι δυο συγκριτικούς και έναν λογικό τελεστή δεδομένων των τιμών των μεταβλητών που χρησιμοποιούνται.

B7.5.6 Διατυπώνουν συνθήκες με μέχρι δύο συγκριτικούς τελεστές και έναν λογικό τελεστή που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση.

B7.5.8 Διατυπώνουν συνθήκες με περισσότερους από έναν λογικούς και συγκριτικούς τελεστές που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση.

B7.5.9 Διατυπώνουν τη δομή της διακλάδωσης If..Else.

Λύση

```
#include<iostream>
```

```
#include<iomanip> (ερώτημα γ 0.25)
```

```
using namespace std;
```

```
int main (){
```

```
int monades;
```

```
float xreosi, ekptosi;
```

```
cout<<"Dose monades katanalosis: "; cin>>monades;
```

```
if .....(monades<=100)... {  
    xreosi=...monades * 0.10;...
```

(ερώτημα α)

```
}
```

```
else
```

```
{  
    xreosi=...(100 * 0.10) + (monades-100) * 0.15...  
}
```

```
if (xreosi>200)
```

```
{ ekptosi = xreosi * 0.10.....
```

```
    xreosi = .....xreosi - ekptosi ..... }
```

(Ερώτημα β)

```
cout<<"o xreosi einai:" <<fixed<<setprecision(2)<<xreosi...<<endl; (ερώτημα γ 0.75)
```

```
return 0;
```

```
}
```

Άσκηση 3

Δείκτες Επιτυχίας
B7.3.3 Ακολουθούν τους κανόνες προτεραιότητας των αριθμητικών τελεστών κατά την προκαταρκτική εκτέλεση ενός προγράμματος με εκφράσεις/παραστάσεις. B7.6.12 Ελέγχουν την ορθότητα της λύσης του προβλήματος χρησιμοποιώντας τη μέθοδο της προκαταρκτικής εκτέλεσης και της χρήσης του debugger για επαλήθευση.

Λύση

Ερώτημα (α) 3μ.

Μεταβλητές		Απόφαση				Παρουσίαση
A	B	B<=3	A/Ψ	A<=10	A/Ψ	AAA
4	0	0<=3	A	10<=10	A	10 □ 2
8	2	2<=3	A	20<=10	Ψ	24 □ 4
10	4	4<=3	Ψ			ΤΕΛΟΣ
20						
24						

Ερώτημα (β) 2μ.

```
[4] B αντί b
[8] while (x<=6) αντί for
[9] απουσιάζει ;
[10] *=2 αντί *=2;
[13] cout<<"AAA" αντί cout<<"AAA"
```

Άσκηση 4

Λύση

```
#include<iostream>
using namespace std;
int main (){
```

```
    int i, num, sum;
    sum=0;
```

```
    for (i=1;i<=10;i++)
```

```
    {
        cout<<"Dose arithmo:"<<endl;
        cin>>num;
```

```
        if (num%2==0)
            sum=sum+num;
```

```
    }
```

```
    if (sum>0)
```

```
        cout<<"Athrisma zygon:"<<sum;
```

Ερώτημα (α) 1μ.

Ερώτημα (β) 2μ.

Ερώτημα (γ) 2μ.

```

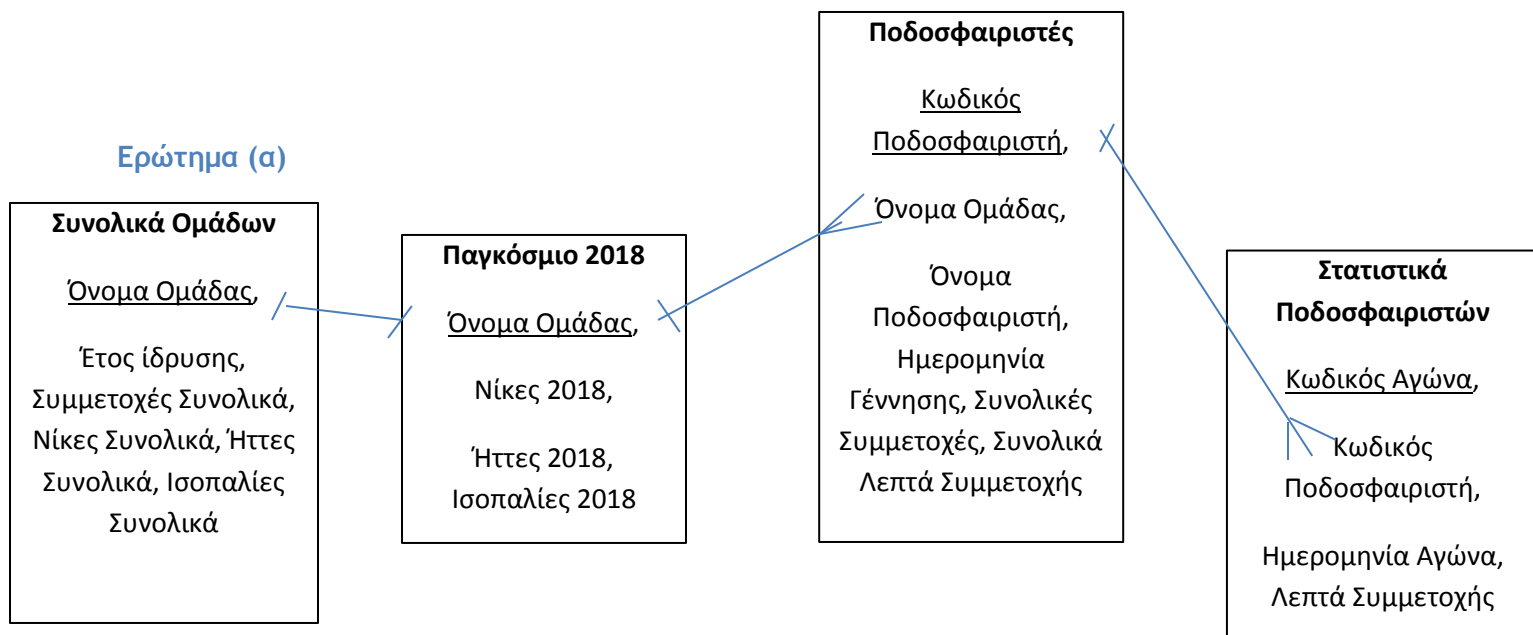
else
    cout<<"Den dothikan zygoi"
;
return 0;
}

```

Άσκηση 5

Δείκτες Επιτυχίας
B6.4.3 Αναφέρουν τεχνικές εξακρίβωσης αναγκών και να εισηγούνται κατάλληλη τεχνική σε εργασιακά παραδείγματα. B6.8.8 Δημιουργούν σχέσεις Ένα-Προς-Πολλά (One-To-Many, 1-∞) και Ένα-Προς-Ένα (One-To-One, 1-1) μεταξύ δύο πινάκων σε μια Βάση Δεδομένων. B6.8.14 Δημιουργούν ερωτήματα, επιλέγοντας πεδία από έναν ή περισσότερους σχετιζόμενους πίνακες και εφαρμόζουν κριτήρια και ταξινόμηση, σύμφωνα με τις απαιτήσεις της σχεδίασης του πληροφοριακού συστήματος.

Ερώτημα (α)



Ερώτημα (β)

1. Προκαταρτική Έρευνα – Μελέτη Σκοπιμότητας
2. Εξακρίβωση Αναγκών και καθορισμός Απαιτήσεων
3. Καθορισμός Προδιαγραφών

Ερώτημα (γ)

Field:	Όνομα Ποδοσφαι	Όνομα Ομάδας	Ημερομηνία Αγώνα	Λεπτά Συμμετοχής	
Table:	Ποδοσφαιριστές	Ποδοσφαιριστές	Στατιστικά Ποδοσφα	Στατιστικά Ποδοσφα	
Sort:		Ascending			
Show:	☒	☒	☒	☒	☐
Criteria:					
or:					

Άσκηση 6

Δείκτες Επιτυχίας
B7.8.3 Κατηγοριοποιούν παραδείγματα παιγνιδιών με βάση τα χαρακτηριστικά τους.
B7.8.13 Υλοποιούν το παιγνίδι με τη χρήση του περιβάλλοντος ανάπτυξης.
B7.8.14 Μεταγλωττίζουν το παιγνίδι τους σε πρόγραμμα και να εντοπίζουν και να διορθώνουν σφάλματα.

Ερώτημα (α)

Το Πανευρωπαϊκό Σύστημα Πληροφόρησης για τα Ηλεκτρονικά Παιγνίδια (PEGI) είναι ένα ευρωπαϊκό σύστημα ταξινόμησης των βιντεοπαιχνιδιών που δημιουργήθηκε για να βοηθήσει τους Ευρωπαίους καταναλωτές να κάνουν συνειδητές αποφάσεις για την αγορά ηλεκτρονικών παιγνιδιών.

Ερώτημα (β)

Ο σκοπός του Game Maker είναι να παράσχει ένα πρόγραμμα κατασκευής δισδιάστατων και τρισδιάστατων παιγνιδιών σε ένα συνηθισμένο προσωπικό υπολογιστή. αρέχει ένα περιβάλλον που μπορεί εύκολα να χρησιμοποιηθεί από αρχάριους χρήστες και την ίδια στιγμή, παρέχει τεράστιες δυνατότητες για ευπαρουσίαστα και εξελιγμένα παιγνίδια.

Ερώτημα (γ)

1. Mouse
2. Create
3. Step
4. collision

ΜΕΡΟΣ Β'

Άσκηση 1

Δείκτες Επιτυχίας
B7.2.11 Χρησιμοποιούν την εντολή εκχώρησης τιμής (Assignment).
B7.3.1 Χρησιμοποιούν βασικούς αριθμητικούς τελεστές (π.χ. +, -, *, /, %) και παρενθέσεις ().
B7.3.2 Να χρησιμοποιούν εξειδικευμένους αριθμητικούς τελεστές (π.χ. ++ και --).
B7.3.8 Χρησιμοποιούν έτοιμες συναρτήσεις από βιβλιοθήκες (εισόδου/εξόδου, μαθηματικών, μετατροπών, κειμένου, κ.λπ.) σε ένα πρόγραμμα.
B7.5.6 Διατυπώνουν συνθήκες με μέχρι δύο συγκριτικούς τελεστές και έναν λογικό τελεστή που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση.
B7.5.7 Χρησιμοποιούν τους κανόνες προτεραιότητας για να προβλέπουν το αποτέλεσμα (true ή false) μιας συνθήκης με πολλούς συγκριτικούς και λογικούς τελεστές δεδομένων των τιμών των μεταβλητών που χρησιμοποιούνται.
B7.5.8 Διατυπώνουν συνθήκες με περισσότερους από έναν λογικούς και συγκριτικούς τελεστές που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση.
B7.5.20 Επιλέγουν κατάλληλη δομή διακλάδωσης ανάλογα με τις δυνατότητες, τους περιορισμούς και τα χαρακτηριστικά της για επίλυση του προβλήματος.
B7.6.5 Επιλέγουν την κατάλληλη δομή επανάληψης με βάση την περιγραφή συγκεκριμένου προβλήματος.
B7.6.9 Επιλέγουν κατάλληλη δομή επανάληψης ανάλογα με τις δυνατότητες, τους περιορισμούς και τα χαρακτηριστικά της για επίλυση του προβλήματος.

Λύση

```
#include <iostream>
using namespace std;
int main()
{
    char fylo;
    int apant, i, plall, plkor, plagor, korex;
    plall=0;plkor=0;plagor=0;korex=0;

    cout<<"Dose fylo"<<endl;
    cin>>fylo;
    while (fylo=='A' || fylo=='K')
    {
        cout<<"Dose Apantisi"<<endl;
        cin>>apant;
        plall++;
        if (fylo=='K')
            plkor++;
        if(fylo=='A' && apant==3)
            plagor++;
        if (fylo=='K' && apant==1)
```

ερώτημα (α)

ερώτημα (β) 2μ.

ερώτημα (γ) 1μ + 2μ.

ερώτημα (δ) 2μ.

ερώτημα (ε) 1μ.

```

        korex++;
        cout<<"Dose fylo"<<endl;
        cin>>fylo;
    }
    cout<<plall<<" "<<plkor<<endl;
    cout<<plagor<<endl;
    cout<<korex;
    return 0;
}

```

ερώτημα (β) 1μ.

αφαιρούμε 0.25 ...

Άσκηση 2

Δείκτες Επιτυχίας
B7.5.3 Διατυπώνουν απλές συνθήκες με έναν συγκριτικό τελεστή που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση. B7.6.3 Χρησιμοποιούν τελεστές αυξομειώσης (++ και --) στη μεταβλητή ελέγχου της δομής For. B7.6.5 Επιλέγουν την κατάλληλη δομή επανάληψης με βάση την περιγραφή συγκεκριμένου προβλήματος. B7.7.2 Εντοπίζουν ποιοι πίνακες (arrays) χρειάζονται (Variables) χρειάζονται με βάση τις ανάγκες του αλγορίθμου/προγράμματος. B7.7.3 Δίνουν κατάλληλο όνομα (αναγνωριστικό) σε έναν πίνακα και να καθορίζουν το μέγεθός του (τον αριθμό στοιχείων που μπορεί να περιέχει). B7.7.4 Επιλέγουν κατάλληλο τύπο δεδομένων (ακέραιος/πραγματικός αριθμός, Boolean, B7.7.5 Δηλώνουν πίνακες (arrays) με βάση τις ανάγκες του προγράμματος. B7.7.6 Αναφέρονται σε στοιχεία του πίνακα χρησιμοποιώντας κατάλληλο δείκτη. B7.7.7 Χρησιμοποιούν ένα βασικό βρόγχο βρόγχος for... για είσοδο/εισαγωγή (εκχώρηση τιμής) στα στοιχεία ενός πίνακα.

Λύση

```

#include<iostream>
#include<iomanip>
using namespace std;
int main (){
    int A[20],i,B[20],sum50,min, pl50, antstr;
    float avg50;

    for (i=0; i<20; i++){
        cout<<"Dose ton "<<i+1<<" arithmo"<<endl;
        cin>>A[i];
        while ((A[i]<10)|| (A[i]>99)){
            cout<<"0 arithmos den einai dipsifios"<<endl;
            cin>>A[i];
        }
    }
}

```

ερώτημα (α) 4μ.

```

sum50=0; pl50=0;;
for (i=0; i<20; i++){
    if (A[i]>50)
    {
        sum50=sum50 + A[i]; //sum50+=A[i];
        pl50++;
    }
}
avg50=(float) sum50/pl50;
cout<<"Mesos oros arithmon pano apo 50: "<<fixed<<setprecision(2)<<avg50<<endl;

for (i=0; i<20; i++)
    B[i]= (A[i] % 10) * 10 + A[i]/10;

min=100;
for (i=0; i<20; i++)
    if (A[i]<min)
        min=A[i];
cout<<"O mikroteros ston pinaka A einai:"<<min;

return 0;
}

```

Ερώτημα(β) 2μ.

Ερώτημα(γ) 2μ.

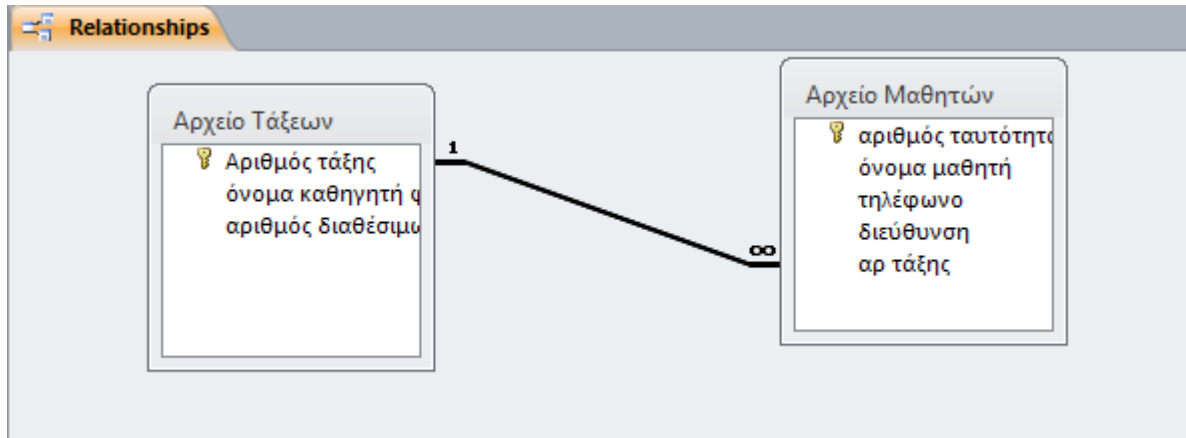
Ερώτημα(δ) 2μ.

Άσκηση 3

Δείκτες Επιτυχίας
B6.5.6 Να εντοπίζουν σφάλματα σε ένα Διάγραμμα ροής Δεδομένων και να προτείνουν τρόπους διόρθωσής τους. B6.6.1 Να αναγνωρίζουν ένα Διάγραμμα Οντοτήτων-Συσχετίσεων (ΔΟΣ). και να αναφέρουν ποιες πτυχές του πληροφοριακού συστήματος παρουσιάζει. B6.7.1 Να μετατρέπουν ένα διάγραμμα οντοτήτων-συσχετίσεων (ΔΟΣ) σε προδιαγραφές δημιουργίας σχετιζόμενων πινάκων, συγκεκριμένα: • Να διασπούν κάθε σχέση Πολλά-Προς-Πολλά σε πίνακα και σχέσεις Ένα-προς-Πολλά. • Να καθορίζουν τον τύπο δεδομένων για κάθε πεδίο και άλλες αναγκαίες απαιτήσεις (ιδιότητες). • Να ελέγχουν/καθορίζουν το πρωτεύον κλειδί σε κάθε πίνακα. • Να ελέγχουν/εισαγάγουν ξένο κλειδί για προσδιορισμό σχέσεων μεταξύ πινάκων. B6.8.14 Δημιουργούν ερωτήματα, επιλέγοντας πεδία από έναν ή περισσότερους σχετιζόμενους πίνακες και εφαρμόζουν κριτήρια και ταξινόμηση, σύμφωνα με τις απαιτήσεις της σχεδίασης του πληροφοριακού συστήματος.

Λύση

Ερώτημα (α) & Ερώτημα (β)



Ερώτημα (γ)

1. ΛΑΘΟΣ
2. ΛΑΘΟΣ
3. ΣΩΣΤΟ
4. ΣΩΣΤΟ
5. ΛΑΘΟΣ
6. ΣΩΣΤΟ

Ερώτημα (δ)

Τρία από τα ποιο κάτω τέσσερα λάθη:

1. Η ροή με τον αριθμό 6 δεν είναι έγκυρη διότι συνδέει απευθείας αποθηκευτικό χώρο με εξωτερική οντότητα. Πρέπει να πηγαίνει μέσω μιας διαδικασίας.
2. Η διαδικασία 8 δεν έχει ροή που να εξέρχεται από αυτή. Άρα αποτελεί Μαύρη Τρύπα. Έπρεπε να μεταφέρονται τα στοιχεία της ροής 7 στο αρχείο
3. Απουσιάζει η ροή που συνδέει τον αποθηκευτικό χώρο 5 (Αρχείο Τάξεων Φωνητικής) με την διαδικασία 11 (Δημιούργησε Κατάλογο Μαθητών ανά Τάξη) για την μεταφορά του ονόματος του καθηγητή φωνητικής που χρειάζεται για την δημιουργία του καταλόγου. Άρα δεν μπορούν να εξαχθούν οι ζητούμενες πληροφορίες με αποτέλεσμα να υπάρχει Γκρίζα Τρύπα.
4. Η διαδικασία 3 δεν έχει ροή που να εξέρχεται από αυτή. Άρα αποτελεί Μαύρη Τρύπα. Τα στοιχεία της ροής 4 έπρεπε να μεταφέρονται από την διαδικασία 3 στο μαθητή.

Άσκηση 4

Δείκτες Επιτυχίας
B7.8.9 Αναγνωρίζουν συστατικά αρχεία του παιχνιδιού και τον ρόλο του καθενός. B7.8.10 Αναγνωρίζουν τα βασικά δομικά στοιχεία και τη σύνταξη της γλώσσας προγραμματισμού του περιβάλλοντος ανάπτυξης ενός παιχνιδιού και να τα αντιστοιχούν με παρόμοια στοιχεία από το περιβάλλον/την κύρια γλώσσα προγραμματισμού που έχουν ήδη χρησιμοποιήσει. B7.8.11 Εντοπίζουν βασικά δομικά στοιχεία ενός παιχνιδιού (σε προγραμματιστικό επίπεδο). B7.8.12 Σχεδιάζουν ένα παιχνίδι (κόσμος, sprite, αντικείμενα, συμπεριφορές). B7.8.13 Υλοποιούν το παιχνίδι με τη χρήση του περιβάλλοντος ανάπτυξης.

Λύση

Ερώτημα (α) 2β.

1. Obj_ball
2. Collision with obj_ballon
3. Destroy the instance (other ticked)
4. Set lives -1 (relative ticked)

Ερώτημα (β) 2β.

1. Obj_controller
2. Draw
3. Draw the lives as image
4. x=0, y=0 Image: spr_lives

Ερώτημα (γ) 3β.

1. Obj_controller
2. Step
3. If number of instance is a value
Start of a block
 - Obj_bottle
 - Display a message
 - End of gameEnd of a block

Ερώτημα (δ) 3β.

1. obj_ball
2. Mouse-left pressed
3. Sound – sndclap
4. Display a message “Nikises”
5. End game

ΜΕΡΟΣ Γ'

Άσκηση 1

Δείκτες Επιτυχίας

B7.5.3 Διατυπώνουν απλές συνθήκες με έναν συγκριτικό τελεστή που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση.
B7.6.3 Χρησιμοποιούν τελεστές αυξομειώσης (++ και --) στη μεταβλητή ελέγχου της δομής For.
B7.6.5 Επιλέγουν την κατάλληλη δομή επανάληψης με βάση την περιγραφή συγκεκριμένου προβλήματος.
B7.7.2 Εντοπίζουν ποιοι πίνακες (arrays) χρειάζονται (Variables) χρειάζονται με βάση τις ανάγκες του αλγορίθμου/προγράμματος.
B7.7.3 Δίνουν κατάλληλο όνομα (αναγνωριστικό) σε έναν πίνακα και να καθορίζουν το μέγεθός του (τον αριθμό στοιχείων που μπορεί να περιέχει).
B7.7.4 Επιλέγουν κατάλληλο τύπο δεδομένων (ακέραιος/πραγματικός αριθμός, Boolean,
B7.7.5 Δηλώνουν πίνακες (arrays) με βάση τις ανάγκες του προγράμματος.
B7.7.6 Αναφέρονται σε στοιχεία του πίνακα χρησιμοποιώντας κατάλληλο δείκτη.
B7.7.7 Χρησιμοποιούν ένα βασικό βρόγχο βρόγχος for... για είσοδο/εισαγωγή (εκχώρηση τιμής) στα στοιχεία ενός πίνακα.
B7.7.9 Χρησιμοποιούν άλλους βρόγχους (π.χ. While, Do .. While) για είσοδο, έξοδο και επεξεργασία στοιχείων ενός πίνακα σύμφωνα με την περιγραφή συγκεκριμένου προβλήματος.
B7.7.13 Αποφασίζουν εάν χρειάζονται πίνακες για την επίλυση του προβλήματος και τους καθορίζουν.

```
#include <iostream>
#include <iomanip>
#include <string>
#define n 30
using namespace std;
int main()
{
    string onoma[n],mxanalm,mxprognm;
    int analysis[n], programmatismos[n], i,analmx,progmx;
    float telikoi[n];
```

Ερώτημα (α) 2μ.

```

for (i=0; i<n;i++)
{
    cout<<"Dose onoma aititi:"<<endl;
    cin>>onoma[i];
    cout<<"Dose vathmologia analysis:"<<endl;
    cin>>analysis[i];
    cout<<"Dose vathmologia programmatismou:"<<endl;
    cin>>programmatismos[i];
}

```

Ερώτημα (α) 2μ.

```

for (i=0; i<n;i++)
{
    telikoi[i]=(analysis[i] * 0.60)+(programmatismos[i]*0.40);
}

```

Ερώτημα (β) 3μ.

```

for (i=0;i<n;i++)
{
    cout<<onoma[i]<<" "<<fixed<<setprecision(2)<<telikoi[i]<<endl;
}

```

Ερώτημα (γ) 3μ.

```

mxanalnm=onoma[0];analmx=analysis[0];
mxprognm=onoma[0];progmx=programmatismos[0];
for (i=0; i<n;i++)
{

```

Ερώτημα (δ) 1μ

```

    if (analysis[i]>analmx)
    {
        analmx=analysis[i];
        mxanalnm=onoma[i];
    }

```

Ερώτημα (δ) 4μ
(μαζί με τα πιο κάτω)

```

    if (programmatismos[i]>progmx)
    {
        progmx=programmatismos[i];
    }
}

```

```

        mxprognm=onoma[i];
    }
}
cout<<"Psiloteri stin Analysi: "<<mxanalnm<<endl;
cout<<"Psiloteri ston programmatismo: "<<mxprognm<<endl;

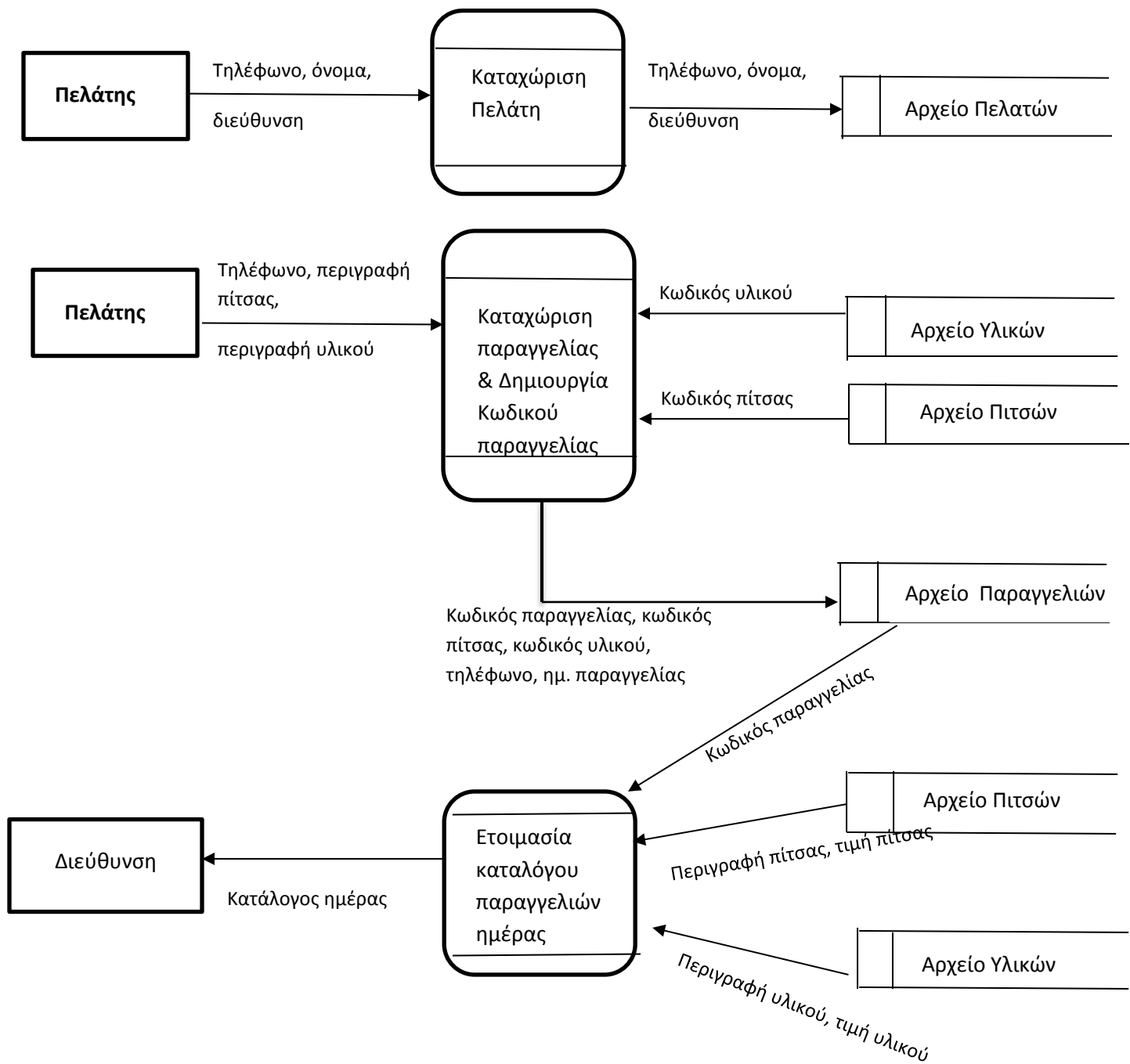
if (mxanalnm==mxprognm)
    cout<<"Exairetika";
return 0;
}

```

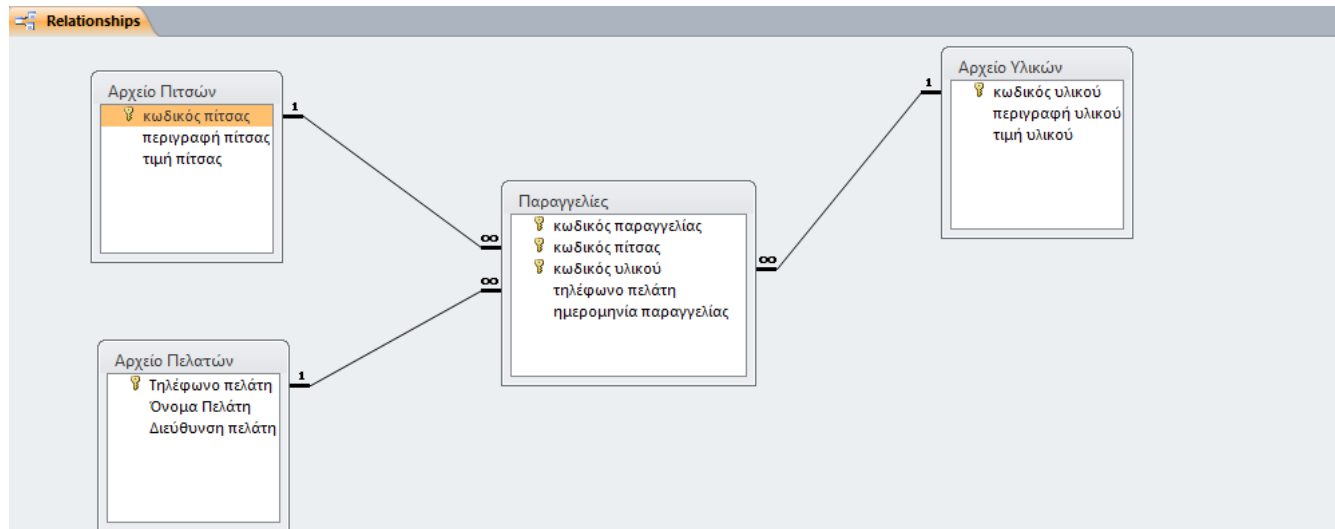
Άσκηση 2

Δείκτες Επιτυχίας
B6.5.3 Να αναγνωρίζουν τα βασικά δομικά στοιχεία στην περιγραφή ενός πληροφοριακού συστήματος και να σχεδιάζουν ένα Διάγραμμα Ροής Δεδομένων που να ταιριάζει στην περιγραφή. B6.6.1 Να αναγνωρίζουν ένα Διάγραμμα Οντοτήτων-Συσχετίσεων (ΔΟΣ). και να αναφέρουν ποιες πτυχές του πληροφοριακού συστήματος παρουσιάζει. B6.6.2 Να αναγνωρίζουν τα σύμβολα ενός Διαγράμματος Οντοτήτων-Συσχετίσεων καθώς και τα βασικά τους γνωρίσματα (ιδιότητες, πρωτεύον κλειδί) και να αναφέρουν τον ρόλο του καθενός. B6.6.3 Να αναγνωρίζουν τα βασικά δομικά στοιχεία στην περιγραφή ενός πληροφοριακού συστήματος και να σχεδιάζουν ένα Διάγραμμα Οντοτήτων-Συσχετισμών που να ταιριάζει στην περιγραφή. B6.7.6 Καθορίζουν τις προδιαγραφές ερωτημάτων που χρειάζονται για τη δημιουργία των εκθέσεων (ποια πεδία, από ποιους πίνακες, με ποια ταξινόμηση και ποια κριτήρια). B6.8.6 Ορίζουν το πρωτεύον κλειδί του πίνακα (Primary Key). B6.8.8 Δημιουργούν σχέσεις Ένα-Προς-Πολλά (One-To-Many, 1-∞) και Ένα-Προς- Ένα (One-To- One, 1-1) μεταξύ δύο πινάκων σε μια Βάση Δεδομένων. B6.8.14 Δημιουργούν ερωτήματα, επιλέγοντας πεδία από έναν ή περισσότερους σχετιζόμενους πίνακες και εφαρμόζουν κριτήρια και ταξινόμηση, σύμφωνα με τις απαιτήσεις της σχεδίασης του πληροφοριακού συστήματος.

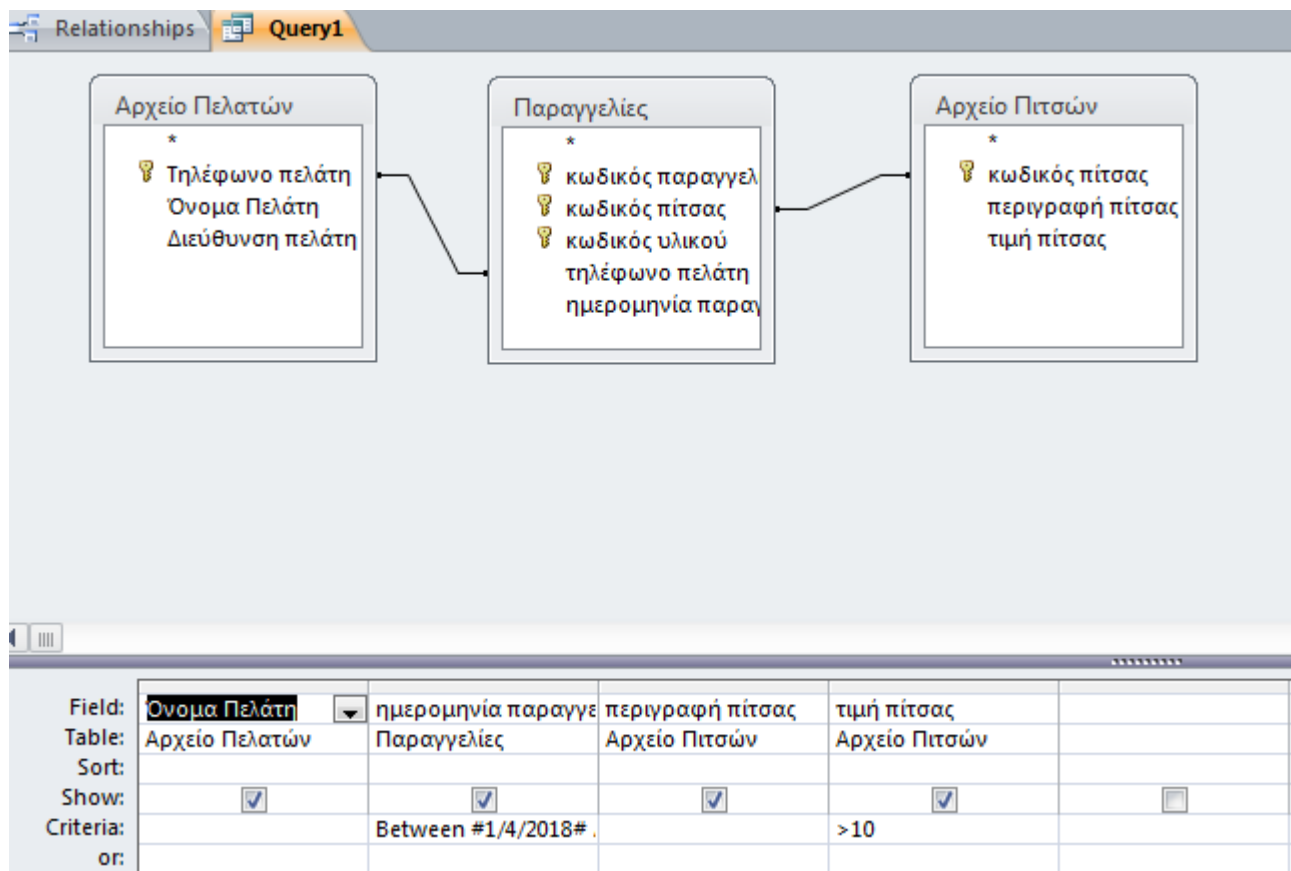
Ερώτημα (α) 5β.



Ερώτημα (β&γ) 3β. + 3β



Ερώτημα (δ) 4β.



ΠΕΡΙΦΕΡΕΙΑΚΟ ΛΥΚΕΙΟ Μ. ΚΟΥΤΣΟΦΤΑ – Α. ΠΑΝΑΓΙΔΗ ΠΑΛΑΙΟΜΕΤΟΧΟΥ

ΣΧΟΛΙΚΗ ΧΡΟΝΙΑ: 2017 - 2018

ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ ΜΑΪΟΥ – ΙΟΥΝΙΟΥ 2018

ΜΑΘΗΜΑ: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΤΕΥΘΥΝΣΗΣ

ΤΑΞΗ: Β' ΛΥΚΕΙΟΥ

ΗΜΕΡΟΜΗΝΙΑ: 30/05/2018

ΒΑΘΜΟΣ: Αριθμ.:

ΔΙΑΡΚΕΙΑ ΕΞΕΤΑΣΗΣ: 2.5 ώρες

Ολογρ.:

ΩΡΑ ΕΝΑΡΞΗΣ: 07:45 π.μ.

Υπογρ.:

ΟΝΟΜΑΤΕΠΩΝΥΜΟ:

ΤΜΗΜΑ:

ΑΡΙΘΜΟΣ:

ΛΥΣΕΙΣ

ΟΔΗΓΙΕΣ

- Το εξεταστικό δοκίμιο αποτελείται από είκοσι (20) σελίδες.
- Το εξεταστικό δοκίμιο αποτελείται από τρία μέρη Α', Β' και Γ'.
- Το μέρος Α' αποτελείται από έξι (6) ερωτήσεις και κάθε ερώτηση βαθμολογείται με πέντε (5) μονάδες.
- Το μέρος Β' αποτελείται από τέσσερις (4) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δέκα (10) μονάδες.
- Το μέρος Γ' αποτελείται από δύο (2) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δεκαπέντε (15) μονάδες.
- Να απαντήσετε όλες τις ερωτήσεις στο εξεταστικό δοκίμιο.
- Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.
- Τα Λογικά Διαγράμματα, τα Διαγράμματα Ροής Δεδομένων, τα Διαγράμματα Σχέσεων Οντοτήτων και τα περιεχόμενά τους, μπορούν να γίνουν με μολύβι.
- Κατοχή κινητού τηλεφώνου = Δολίευση.
- Απαγορεύεται η χρήση διορθωτικού υγρού.
- Οι μοναδικές βιβλιοθήκες που επιτρέπονται στη δημιουργία προγραμμάτων, είναι η `<iostream>` και η `<cmath>`, εκτός και αν ζητείται από την εκφώνηση.

ΜΕΡΟΣ Α'

Άσκηση 1

Ερώτημα (α) (Βαθμοί 2)

Να γράψετε τις αντίστοιχες εκφράσεις στην C++ για τις πιο κάτω μαθηματικές εκφράσεις:

- (i)
$$p = \frac{3x + |z-y|}{2+y}$$

`p = (3*x + abs(z-y)) / (2+y);`
- (ii)
$$p = 2x - \sqrt{y+x} + x^3$$

`α. p= 2*y - sqrt(y+x) + pow(x,3);`
`β. p= 2*y - sqrt(y+x) + x*x*x;`

(α) και (β) 3 μονάδες

Λάθος σύμβολο (π.χ. [x ή · αντί *], [: αντί /], λάθος διατύπωση π.χ των pow, sqrt, abs) ή αν λείπει ένα σύμβολο: -0,25 για κάθε λάθος
Λάθος θέση παρενθέσεων ή αν λείπουν οι (): -0,25 για κάθε λάθος

Αν δεν μπει ; στο τέλος: -0,25 μια φορά

γ) i) 1 μονάδα (ορθή απάντηση → 1 μ
λάθος απάντηση → 0 μ)
ii) 1 μονάδα (ορθή απάντηση → 1 μ
λάθος απάντηση → 0 μ)

Ερώτημα (β) (Βαθμοί 1)

Να μεταφράσετε την πιο κάτω πρόταση σε λογική έκφραση (συνθήκη) στη γλώσσα προγραμματισμού C++:

Η τιμή της μεταβλητής **num** να διαιρείται ακριβώς με το 4.

`α. (num % 4 == 0)`

`β. (num / 4 == num / 4.0)`

Ερώτημα (γ) (Βαθμοί 2)

Να μετατρέψετε:

- (i) τον αριθμό **(243)₁₀** από το δεκαδικό στο δυαδικό σύστημα αρίθμησης.

243 / 2 = 121	1	1
121 / 2 = 60	11	1
60 / 2 = 30	011	0
30 / 2 = 15	0011	0
15 / 2 = 7	10011	1
7 / 2 = 3	110011	1
3 / 2 = 1	1110011	1
1 / 2 = 0	11110011	1

ή

$$243 = 128 + 64 + 32 + 16 + 2 + 1$$

$$(243)_{10} = (11110011)_2$$

- (ii) τον αριθμό **(10101011)₂** από το δυαδικό στο δεκαδικό σύστημα αρίθμησης.

$$128 + 32 + 8 + 2 + 1 = 171$$

$$(10101011)_2 = (171)_{10}$$

Άσκηση 2

Ερώτημα (α) (Βαθμοί 3)

Να γράψετε τα αποτελέσματα των πιο κάτω εκφράσεων, οι οποίες είναι γραμμένες στην γλώσσα προγραμματισμού C++ :

(i) `cout << 15 / 3 * 2 % 4 / 2;`

`15 / 3 * 2 % 4 / 2 =`

`5 * 2 % 4 / 2 =`

`10 % 4 / 2 = 2 / 2 = 1`

`1`

<u>(α) i) 1 μονάδα</u>	(ορθή απάντηση → 1 μ λάθος απάντηση → 0 μ)
<u>ii) 1 μονάδα</u>	(ορθή απάντηση → 1 μ λάθος απάντηση → 0 μ)
<u>iii) 1 μονάδα</u>	(ορθή απάντηση → 1 μ λάθος απάντηση → 0 μ)

(ii) `cout << 2 * pow(3,2) + ( 7 - 2 * 3) - abs(10 - 5 / 2);`

`2 * pow(3,2) + ( 7 - 2 * 3) - abs(10 - 5 / 2) =`

`2 * 9 + (7 - 6) - abs(10 - 2) = 18 + 1 - abs(8) =`

`18 + 1 - 8 = 11`

`11`

(iii) `cout << 10 / 7 + 15 % 5 - round(5.6) + sqrt(3 * 6 - 2) * 2;`

`10 / 7 + 15 % 5 - round(5.6) + sqrt(3 * 6 - 2) * 2 =`

`1 + 0 - 6.0 + sqrt(16) * 2 = -5.0 + 4 * 2 = -5.0 + 8 = 3`

`3`

Ερώτημα (β) (Βαθμοί 2)

Οι μεταβλητές *x*, *y* και *z* έχουν τις ακόλουθες τιμές: *x* = 5.5, *y* = 6, *z* = 7.5.

Να γράψετε τις τιμές που θα έχουν οι λογικές μεταβλητές *a* και *b* (boolean), όταν εκτελεστούν οι πιο κάτω εντολές στη γλώσσα προγραμματισμού C++ :

(i) `a = ((1 > abs( x - z)) && (y > x));`

`(1 > abs(5.5 - 7.5)) && (6 > 5.5) =`

`(1 > 2) && (6 > 5.5) =`

`F && T = F`

<u>(β) i) 1 μονάδα</u>	(ορθή απάντηση → 1 μ λάθος απάντηση → 0 μ)
<u>ii) 1 μονάδα</u>	(ορθή απάντηση → 1 μ λάθος απάντηση → 0 μ)

(ii) `b = ((round(x) >= y) || (y != 6) || !(x > z) && (trunc(z) < y));`

`(round(5.5) >= 6) || (6 != 6) || !(5.5 > 7.5) && (trunc(7.5) < 6) =`

`(6 >= 6) || (6 != 6) || !(5.5 > 7.5) && (7 < 6) =`

`T || F || T && F = T || F || F = T || F = T`

Άσκηση 3

Σας δίνεται το πιο κάτω πρόγραμμα, που σκοπό έχει να υπολογίζει την τελική βαθμολογία του έτους ενός μαθητή στην Πληροφορική. Το πρόγραμμα δέχεται τις βαθμολογίες των δύο τετραμήνων του μαθητή (**vath1** και **vath2** που είναι ακέραιοι αριθμοί) και την βαθμολογία που πήρε στην τελική του εξέταση (**vath3** που είναι επίσης ακέραιος αριθμός). Ακολουθώντας υπολογίζει τον μέσο όρο των τριών βαθμολογιών που είναι η τελική βαθμολογία του έτους (**tel_vath** που είναι δεκαδικός αριθμός) και την παρουσιάζει με ακρίβεια ενός δεκαδικού ψηφίου.

```
#include<iostream>
#include<iomanip>
using namespace std;

int main(){
    cout<<"Δώσε τις τρεις βαθμολογίες του μαθητή: ";
    cin>>vath1>>vath2>>vath3;
    tel_vath=(float)(vath1 + vath2 + vath3)/3;
    cout<<"Τελ. Βαθμολογία:"<<fixed<<setprecision(1)<<tel_vath<<endl;
    return 0;
}
```

Ερώτημα (α) (Βαθμοί 1)

Να γράψετε την εντολή ή τις εντολές δήλωσης των απαραίτητων μεταβλητών.

```
int vath1, vath2, vath3;
float tel_vath;
```

(α) 1 μονάδα κάθε λάθος -0.25μ

Ερώτημα (β) (Βαθμοί 2)

Να δηλώσετε την απαραίτητη βιβλιοθήκη και να τροποποιήσετε την εντολή:

```
cout<<"Τελ. Βαθμολογία:"<<fixed<<setprecision(1)<<tel_vath<<endl;
```

ώστε η τελική βαθμολογία να εμφανίζεται στρογγυλεμένη στον πλησιέστερο ακέραιο αριθμό (χωρίς δεκαδικά).

(i) Βιβλιοθήκη : `include <cmath>`

(ii) Εντολή: `cout << "Τελ. Βαθμολογία:" << round(tel_vath) << endl;`

(β) (i) 0.5 μονάδα κάθε λάθος -0.25μ (max -0.5μ)

(ii) 1.5 μονάδες κάθε λάθος -0.25μ (max -1.5μ)

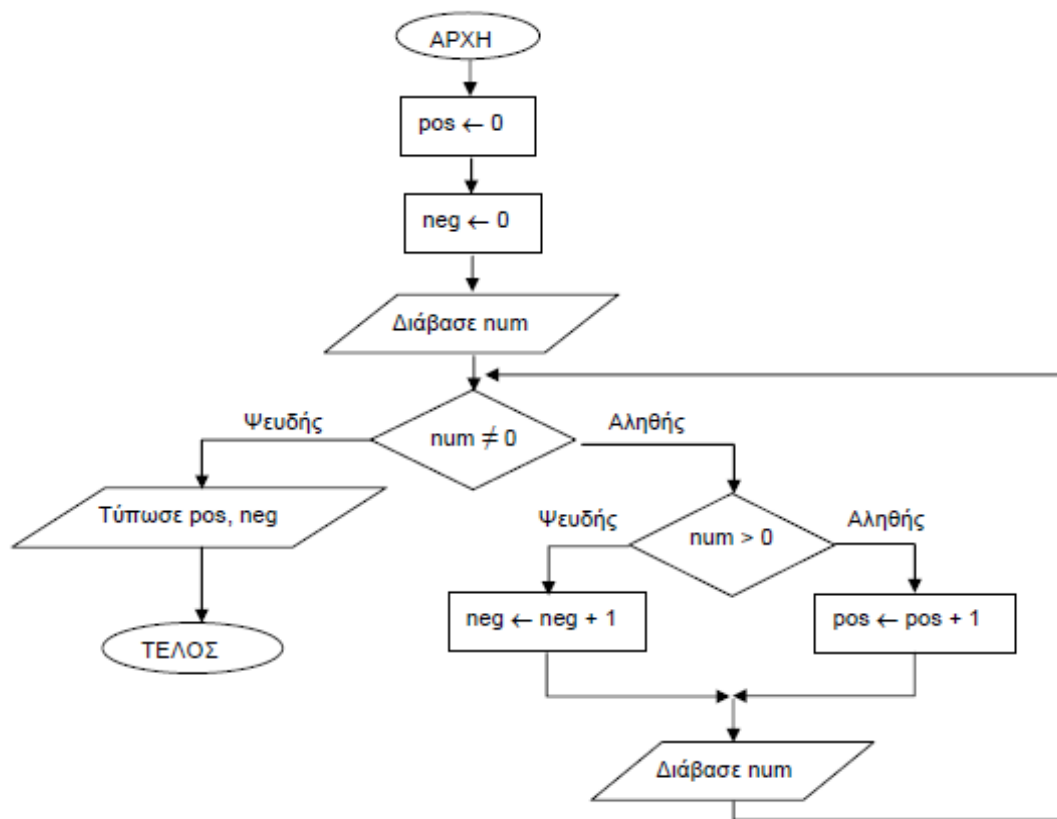
Ερώτημα (γ) (Βαθμοί 2)

Να γράψετε τις εντολές που απαιτούνται, ώστε το πρόγραμμα να παρουσιάζει το μήνυμα «Άριστα», αν η τελική βαθμολογία του μαθητή είναι 18.5 και πάνω.

```
if (tel_vath >= 18.5)    ← 1.25    Για κάθε λάθος: -0.25μ (max -1.25μ)
    cout << "Άριστα";    ← 0.75    Για κάθε λάθος: -0.25μ (max -0.75μ)
```

Άσκηση 4

Σας δίνεται το λογικό διάγραμμα:



Ερώτημα (α) (Βαθμοί 2)

Χρησιμοποιώντας τη μέθοδο της προκαταρκτικής εκτέλεσης (Dry Run), να παρουσιάσετε τα αποτελέσματα του πιο πάνω Λογικού Διαγράμματος όταν δοθούν στη μεταβλητή **num** κατά σειρά οι τιμές: **-5, -7, 10, 0**.

0.25 για όλες τις επικεφαλίδες

ΜΕΤΑΒΛΗΤΕΣ			ΑΠΟΦΑΣΕΙΣ (ΣΥΝΘΗΚΕΣ)				Παρουσίαση
pos	neg	num	num ≠ 0	Α/Ψ	num > 0	Α/Ψ	
0	0	-5	-5 ≠ 0	Α	-5 > 0	Ψ	1 2
1	1	-7	-7 ≠ 0	Α	-7 > 0	Ψ	
	2	10	10 ≠ 0	Α	10 > 0	Α	
		0	0 ≠ 0	Ψ			
0.25	0.25	0.25	0.25		0.25		0.5

Ερώτημα (Β) (Βαθμοί 3)

Να μετατρέψετε το πιο πάνω λογικό διάγραμμα σε πρόγραμμα στη C++.

```
#include <iostream>
using namespace std;
int main () {
    int pos, neg, num; /* Ορθά αποτελέσματα και με float */
    pos = 0;
    neg = 0;
    cin >> num;
    while (num != 0) {
        if (num > 0)
            pos++; /* ή pos += 1; ή pos = pos + 1; */
        else
            neg++; /* ή neg += 1; ή neg = neg + 1; */
        cin >> num;
    }
    cout << pos << " " << neg;
    return 0;
}
```

(Β) 3 μονάδα

κάθε λάθος εντολή -0.25μ (max -3μ)

Άσκηση 5

Ερώτημα (α) (Βαθμοί 1.5)

Να αναφέρετε την πρώτη και την τελευταία φάση του κύκλου ζωής και ανάπτυξης του πληροφοριακού συστήματος.

Πρώτη φάση: Προκαταρκτική Έρευνα - Μελέτη Σκοπιμότητας

Τελευταία φάση: Συντήρηση Συστήματος

(α) πρώτη φάση 0.75 μονάδα (ορθή απάντηση → 0.75 μ
λάθος απάντηση → 0 μ)
Δεύτερη φάση 0.75 μονάδα (ορθή απάντηση → 0.75 μ
λάθος απάντηση → 0 μ)

Ερώτημα (β) (Βαθμοί 1.5)

Να αναφέρετε σε ποια φάση του Κύκλου Ζωής και Ανάπτυξης Πληροφοριακών Συστημάτων λαμβάνει χώρα το κάθε ένα από τα ακόλουθα:

1. Η ανάπτυξη των διαφόρων μερών του συστήματος με κώδικα.
2. Ο σχεδιασμός του Διαγράμματος Ροής Δεδομένων.

1. Υλοποίησης Συστήματος

2. Καθορισμού Προδιαγραφών

(β) 1. 0.75 μονάδα (ορθή απάντηση → 0.75 μ
λάθος απάντηση → 0 μ)
2. 0.75 μονάδα (ορθή απάντηση → 0.75 μ
λάθος απάντηση → 0 μ)

Ερώτημα (γ) (Βαθμοί 2)

Κατά τη φάση της Προκαταρκτικής Έρευνας έχουν καταγραφεί εκτός των άλλων και τα πιο κάτω:

- 1) Η Ευρωπαϊκή Ένωση έδωσε οδηγίες για αλλαγή του τρόπου φορολογίας των εισαγόμενων προϊόντων από χώρες μέλη της Ένωσης. Το σύστημα πρέπει να τροποποιηθεί και να αλλαχθούν οθόνες, αναφορές κλπ.
- 2) Ο όγκος συναλλαγών των πελατών μιας εταιρείας έχει αυξηθεί κατά πολύ και η εύρεση των συναλλαγών που δεν έχουν εξοφληθεί ακόμη είναι χρονοβόρα.

Να γράψετε για το κάθε ένα από τα πιο πάνω αν είναι Πρόβλημα, Ευκαιρία ή Εντολή.

1. Έντολή

2. Πρόβλημα

(γ) 1. 1 μονάδα (ορθή απάντηση → 1 μ
λάθος απάντηση → 0 μ)
2. 1 μονάδα (ορθή απάντηση → 1 μ
λάθος απάντηση → 0 μ)

Άσκηση 6

Ερώτημα (α) (Βαθμοί 1)

Για ποια ηλικιακά επίπεδα είναι οι δύο πιο κάτω επισημάνσεις ενός βιντεοπαιχνιδιού;



PEGI 3

Για όλες τις ηλικιακές ομάδες



PEGI 16

Για ηλικίες άνω των 16 ετών

(α) 1 μονάδα

0.5 μονάδα για κάθε ορθή απάντηση ($2 \times 0.5 = 1\mu$)

Ερώτημα (β) (Βαθμοί 2)

Στο πίσω μέρος της συσκευασίας ενός βιντεοπαιχνιδιού υπάρχουν περιγραφικές ενδείξεις που αναφέρουν τους κύριους λόγους για τους οποίους ένα παιγνίδι έλαβε μια ορισμένη ηλικιακή διαβάθμιση. Υπάρχουν οι πιο κάτω τέσσερις ενδείξεις. Να νοιώσετε δίπλα από την καθεμιά τι ένδειξη είναι.

(β) 2 μονάδες

0.5 μονάδα για κάθε ορθή απάντηση ($4 \times 0.5 = 2\mu$)

Επισημάνση		Επισημάνση	
	Χυδαία Γλώσσα		Φόβος
	Ναρκωτικά		Βία

Ερώτημα (γ) (Βαθμοί 1)

Υπάρχουν πολλά είδη παιγνιδιών, τα οποία κατηγοριοποιούνται με βάση τα κύρια χαρακτηριστικά τους. Αναφέρετε δύο (2) από τα είδη παιχνιδιών που γνωρίζετε.

1. Περιπέτειας

(γ) 1 μονάδα

0.5 μονάδα για κάθε ορθή απάντηση ($2 \times 0.5 = 1\mu$)

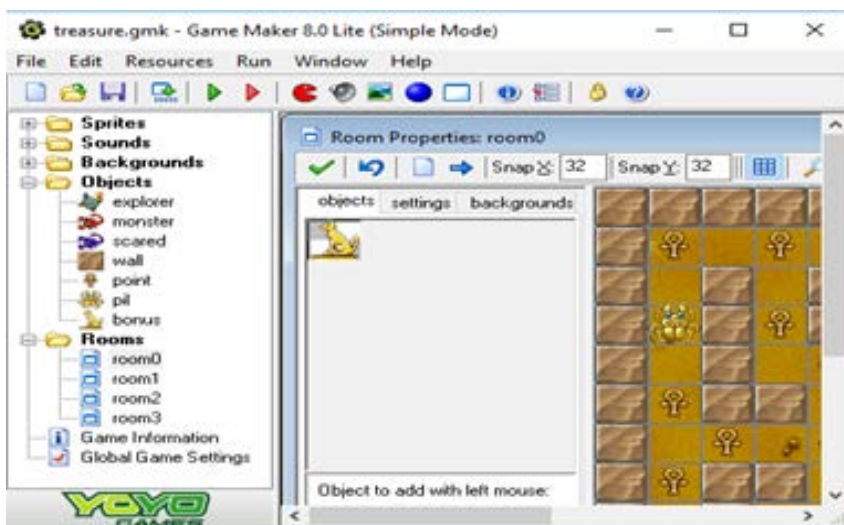
2. Αγώνων Ταχύτητας

Ερώτημα (δ) (Βαθμοί 1)

Στην πιο κάτω εικόνα φαίνεται ένα στιγμιότυπο από τη δημιουργία του παιχνιδιού **Treasure Island** από τη Yoyo Games στο GameMaker. Να γράψετε πόσα αντικείμενα (objects) και πόσα δωμάτια (rooms) χρησιμοποιούνται στο παιχνίδι αυτό.

(δ) 1 μονάδα

0.5 μονάδα για κάθε ορθή απάντηση ($2 \times 0.5 = 1\mu$)



Αντικείμενα (objects): 7

Δωμάτια (rooms): 4

ΤΕΛΟΣ Α' ΜΕΡΟΥΣ

ΜΕΡΟΣ Β'

Άσκηση 1

Ένα ιδιωτικό φροντιστήριο ξένων γλωσσών θα τοποθετήσει τους μαθητές που θα εγγραφούν φέτος στην Α' Τάξη σε τμήματα **10 μαθητών ακριβώς**.

Να δημιουργήσετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο:

- να διαβάζει από τον χρήστη τον αριθμό των μαθητών που έχουν εγγραφεί στην Α' Τάξη (**Βαθμοί 2.5 - με τη δήλωση των μεταβλητών και βιβλιοθηκών**)
- να υπολογίζει και να παρουσιάζει πόσα συμπληρωμένα τμήματα των 10 μαθητών μπορούν να δημιουργηθούν (**Βαθμοί 2.5**)
- αν υπολοίπονται κάποιοι μαθητές για να συμπληρωθεί ακόμη ένα τμήμα, να υπολογίζει και να παρουσιάζει τον αριθμό των μαθητών που χρειάζονται, για να συμπληρωθεί το τμήμα αυτό. Αν είναι ήδη συμπληρωμένο, να βγάζει το μήνυμα «Δεν χρειάζονται άλλοι μαθητές» (**Βαθμοί 5**)

Το πρόγραμμα πρέπει να εμφανίζει στην οθόνη τα κατάλληλα μηνύματα για την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.

Παράδειγμα Εξόδου: Αριθμός μαθητών: Τμήματα = 8 Μαθητές που χρειάζονται για να συμπληρωθεί ακόμη ένα τμήμα = 3	Παράδειγμα Εισόδου: 87
Παράδειγμα Εξόδου: Αριθμός μαθητών: Τμήματα = 11 Δεν χρειάζονται άλλοι μαθητές.	Παράδειγμα Εισόδου: 110

```
#include <iostream>
using namespace std;
int main() {
    int students;

    cout << "Αριθμός μαθητών: ";
    cin >> students;

    /* Μπορούν να χρησιμοποιηθούν ενδιάμεσες μεταβλητές
     * αντί να τυπωθεί απ' ευθείας το αποτέλεσμα */
    cout << "Τμήματα = " << students / 10 << endl;

    if (students % 10 == 0)
        cout << "Δεν χρειάζονται άλλοι μαθητές." << endl;
    else {
        cout << "Μαθητές που χρειάζονται για να συμπληρωθεί ακόμη ένα τμήμα=";
        cout << 10 - (students % 10) << endl;
    }
    return 0;
}
```

Αφαιρετικά: Αν λείπει έστω και ένα ; : - 0.5 μια φορά
Αν λείπει έστω και ένα endl : - 0.5 μια φορά
Αν λείπει έστω και ένα ή είναι λάθος { ή } : - 0.5 μια φορά
Αν λείπουν τα μηνύματα ή είναι λάθος : - 0.5 μια φορά

Άσκηση 2

Το πιο κάτω πρόγραμμα στη γλώσσα προγραμματισμού C++ διαβάζει τυχαίους ακέραιους αριθμούς που δίνονται από το χρήστη. Η επανάληψη τερματίζεται, όταν δοθεί αριθμός που είναι ίδιος με τον αμέσως προηγούμενο. Το πρόγραμμα υπολογίζει και παρουσιάζει πόσοι αριθμοί εδόθησαν και το άθροισμά τους. Στην περίπτωση που οι δυο πρώτοι αριθμοί που δόθηκαν είναι ίσοι, τότε παρουσιάζεται το μήνυμα 'ΑΡΙΘΜΟΙ ΙΣΟΙ' και το πρόγραμμα τερματίζεται. Στους υπολογισμούς δεν συμπεριλαμβάνεται ο τελευταίος αριθμός.

Στο πρόγραμμα υπάρχουν λογικά ή/και συντακτικά λάθη. Να γράψετε πέντε (5) από αυτά, αναφέροντας τον αριθμό της γραμμής στην οποία εμφανίζεται το κάθε λάθος μαζί με τη διορθωμένη εντολή. (2 Βαθμοί για κάθε λάθος που εντοπίζετε και διορθώνετε)

```
/*1*/ #include <iostream>
/*2*/ using namespace std;
/*3*/ int main() {
/*4*/     int a, b, sum;
/*5*/     cout<<"Δώστε έναν αριθμό: ";
/*6*/     cin<<a;
/*7*/     sum = a;
/*8*/     plithos = 0;
/*9*/     cout<<"Δώστε τον επόμενο αριθμό: ";
/*10*/    cin>>b;
/*11*/    if (a = b)
/*12*/        cout<<"ΑΡΙΘΜΟΙ ΙΣΟΙ";
/*13*/    else {
/*14*/        while (a != b) {
/*15*/            plithos = plithos + 1;
/*16*/            sum = sum + b;
/*17*/            b = a;
/*18*/            cout<<"Δώστε τον επόμενο αριθμό: ";
/*19*/            cin>>b;
/*20*/        }
/*21*/        cout<<"Δόθηκαν "<<plithos<<" και το άθροισμά τους είναι "<<sum;
/*22*/    }
/*23*/    return 0;
/*24*/ }
```

Λάθη:

- 1./* 4 */ **int a, b, sum, plithos;**
- 2./* 6 */ **cin >> a;**
- 3./* 8 */ **plithos = 1;**
- 4./* 11 */ **if (a == b)**
- 5./* 17 */ **a = b;**

5 * 2 = 10 Βαθμοί

0.5 για τον αριθμό γραμμής του λάθους και **1.5** για τη διόρθωση του λάθους.

Αν γράψει απλά αριθμό γραμμής και τίποτε άλλο τότε (αν έχει όντως λάθος η γραμμή) παίρνει 0.5μ

Αν γράψει την εντολή διορθωμένη χωρίς αριθμό γραμμής ή αριθμό γραμμής λάθος, τότε παίρνει και τις 2 μονάδες.

Άσκηση 3

Ένα σωματείο θέλει να καταγράφει τις εισφορές του μήνα από διάφορα άτομα ή φορείς και να βγάζει και κάποια στατιστικά στοιχεία. Να γράψετε πρόγραμμα στη C++ το οποίο:

- α) να διαβάζει από τον χρήστη, για κάθε εισφορά, το ποσό της εισφοράς σε Ευρώ και το όνομα του ατόμου ή του φορέα που έκανε την εισφορά. Η καταχώρηση των στοιχείων να σταματήσει, όταν ο χρήστης δώσει ποσό εισφοράς αρνητικό αριθμό ή 0 (Βαθμοί 4.5 - με δήλωση μεταβλητών και βιβλιοθηκών)
- β) να υπολογίζει και να παρουσιάζει το άθροισμα όλων των εισφορών που έγιναν προς το σωματείο τον συγκεκριμένο μήνα (Βαθμοί 1.5)
- γ) να υπολογίζει και να παρουσιάζει πόσες εισφορές ήταν κάτω από 10 Ευρώ και πόσες εισφορές ήταν πάνω από 500 Ευρώ (Βαθμοί 2)
- δ) να βρίσκει και να παρουσιάζει το όνομα του ατόμου ή του φορέα με το μεγαλύτερο ποσό εισφοράς καθώς και το ύψος της εισφοράς του. Να υποθέσετε ότι μόνο ένα άτομο ή φορέας έχει το μεγαλύτερο ποσό εισφοράς. (Βαθμοί 2)

Το πρόγραμμα πρέπει να εμφανίζει στην οθόνη τα κατάλληλα μηνύματα για την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.

<u>Παράδειγμα Εξόδου:</u>	<u>Παράδειγμα Εισόδου:</u>
Ποσό Εισφοράς:	80
Όνομα Χορηγού:	Κώστας
Ποσό Εισφοράς:	550
Όνομα Χορηγού:	Λεωνίδας
Ποσό Εισφοράς:	120
Όνομα Χορηγού:	Γεωργία
Ποσό Εισφοράς:	600
Όνομα Χορηγού:	ΑΜΜΟΧ
Ποσό Εισφοράς:	4
Όνομα Χορηγού:	Νικόλας
Ποσό Εισφοράς:	0
Συνολικό Ποσό Συνδρομών: 1354	
Εισφορές κάτω από 10 Ευρώ: 1	
Εισφορές πάνω από 500 Ευρώ: 2	
Ατομο με πιο ψηλή προσφορά: ΑΜΜΟΧ	
Ποσό πιο ψηλής προσφοράς: 600	

```

#include <iostream>
#include <string>
#include <iomanip>
using namespace std;
int main() {
    float eisfora, sum = 0, max_eisfora = 0;
    string onoma, max_onoma;
    int k500 = 0, a10 = 0;
    cout << "Ποσό Εισφοράς: ";
    cin >> eisfora;
    while (eisfora > 0) {
        cout << "Όνομα Χορηγού: ";
        cin >> onoma;
        sum += eisfora;
        if (eisfora < 10)
            a10++;
        else if (eisfora > 500)
            k500++;
        if (eisfora > max_eisfora) {
            max_eisfora = eisfora;
            max_onoma = onoma;
        }
        cout << "Ποσό Εισφοράς: ";
        cin >> eisfora;
    }
    cout << "Συνολικό Ποσό Συνδρομών: " << sum << endl;
    cout << "Εισφορές κάτω από 10 Ευρώ: " << a10 << endl;
    cout << "Εισφορές πάνω από 500 Ευρώ: " << k500 << endl;
    cout << "Ατομο με πιο ψηλή προσφορά: " << max_onoma << endl;
    cout << "Ποσό πιο ψηλής προσφοράς: " << max_eisfora << endl;
    return 0;
}

```

4.5

2 Για κάθε λάθος: -0.25μ (max -2μ)

0.5

0.5

1.5

1

0.75

2

0.75

2

1.5

0.5

0.5

0.25

0.25

0.25

0.25

Αφαιρετικά: Αν λείπει έστω και ένα ; : - 0.25 μια φορά
 Αν λείπει έστω και ένα endl : - 0.25 μια φορά
 Αν λείπει έστω και ένα ή είναι λάθος { ή } : - 0.25 μια φορά
 Αν λείπουν τα μηνύματα ή είναι λάθος : - 0.5 μια φορά

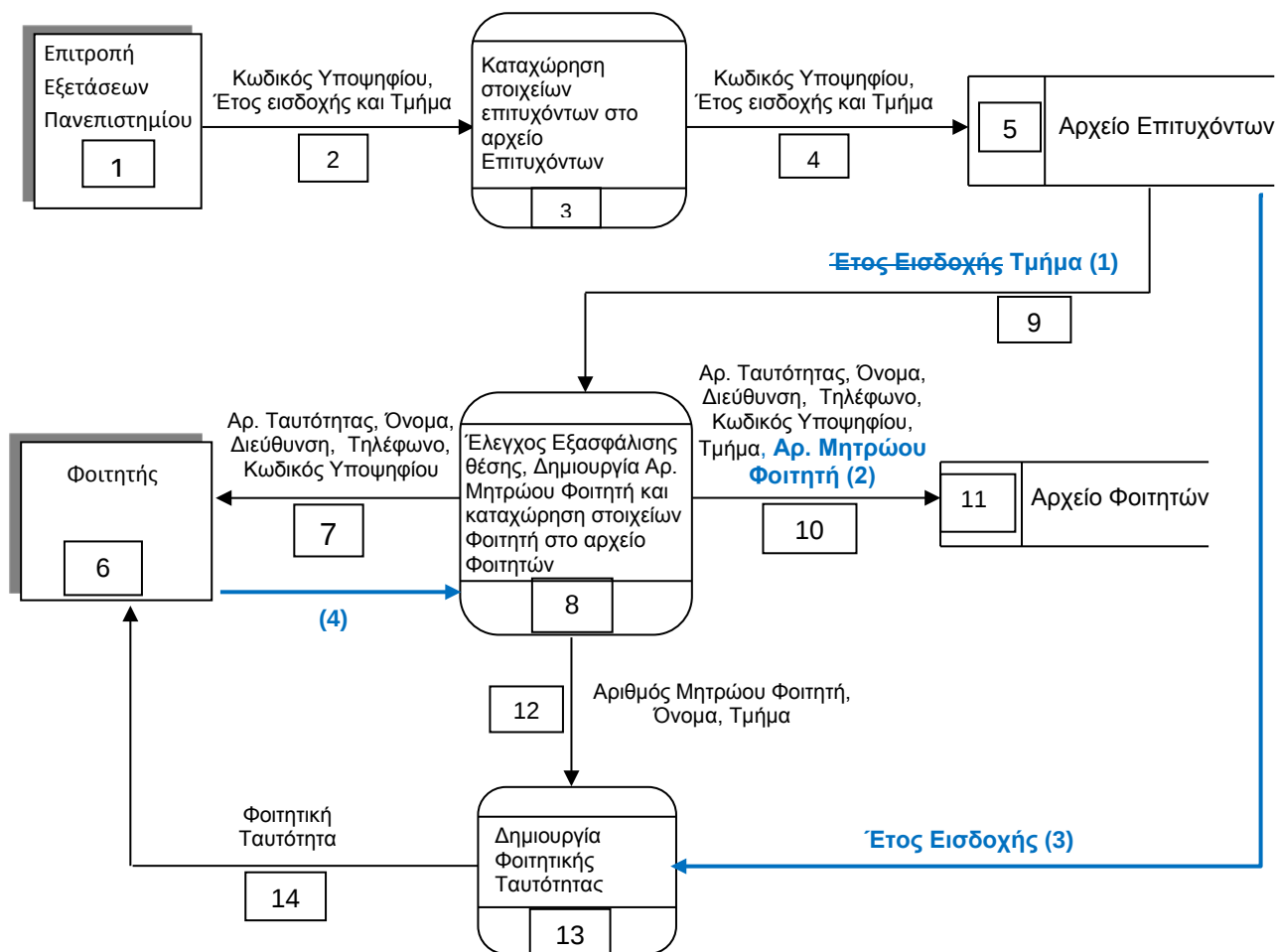
Άσκηση 4

Το πιο κάτω Διάγραμμα Ροής Δεδομένων περιγράφει την ακόλουθη λειτουργία:

Για την εισδοχή νέων φοιτητών σε ένα πανεπιστήμιο διεξάγονται εισαγωγικές εξετάσεις. Όταν βγουν τα αποτελέσματα, η επιτροπή εξετάσεων του πανεπιστημίου δίνει τα στοιχεία των επιτυχόντων (Κωδικός Υποψηφίου, Έτος εισδοχής και Τμήμα), για να καταχωρηθούν στο σύστημα, στο αρχείο επιτυχόντων.

Όταν ένας νέος φοιτητής, ο οποίος πέτυχε στις εξετάσεις, προσέλθει για εγγραφή, δίνει τα προσωπικά του στοιχεία (Αριθμό Ταυτότητας, Όνομα, Διεύθυνση, Τηλέφωνο) καθώς και τον κωδικό υποψηφίου. Με βάση τον κωδικό αυτό γίνεται έλεγχος στο αρχείο επιτυχόντων, για να εξακριβωθεί αν όντως το συγκεκριμένο άτομο έχει εξασφαλίσει θέση και αν ναι, επιστρέφει το τμήμα (π.χ. Πληροφορικής, Οικονομικών, κλπ) στο οποίο έχει εξασφαλίσει θέση. Το τμήμα αυτό μαζί με τα προσωπικά στοιχεία του νέου φοιτητή και έναν αριθμό μητρώου φοιτητή που δημιουργείται αυτόματα από το σύστημα καταχωρούνται στο αρχείο φοιτητών. Ακολούθως, δημιουργείται φοιτητική ταυτότητα η οποία περιέχει τον αριθμό μητρώου φοιτητή, το όνομα, το έτος εισδοχής και το τμήμα του και η οποία παραδίδεται στο νέο φοιτητή.

Να αναφέρετε και να δικαιολογήσετε **τέσσερα** από τα λάθη που υπάρχουν στο ΔΡΔ. (2.5 Βαθμοί για κάθε λάθος που εντοπίζετε και δικαιολογείτε)



- Λάθη:**
1. Στη ροή 9 θα πρέπει να επιστρέφεται το Τμήμα και όχι το Έτος Εισδοχής
 2. Στη ροή 10 θα πρέπει να συμπεριληφθεί και ο Αριθμός Μητρώου Φοιτητή αφού πρέπει να καταχωρηθεί στο Αρχείο Φοιτητών
 3. Θα πρέπει να δημιουργηθεί νέα ροή από το Αρχείο Επιτυχόντων προς τη διαδικασία 13 με την οποία να μεταφέρεται το Έτος Εισδοχής επειδή χρειάζεται για τη δημιουργία της Φοιτητικής Ταυτότητας.
 4. Η ροή 7 θα έπρεπε να κατευθύνεται από το Φοιτητή προς τη Διαδικασία 8 και όχι ανάποδα.

$$4 * 2.5 = 10$$

Εντοπισμός: **1 μ**

Επεξήγηση : **1.5 μ**

ΤΕΛΟΣ Β' ΜΕΡΟΥΣ

ΜΕΡΟΣ Γ'

Άσκηση 1

Κάθε σχολική χρονιά εγγράφεται στα 38 Λύκεια της Κύπρου ένας αριθμός μαθητών. Τα ονόματα των 38 Λυκείων καταχωρούνται σε ένα μονοδιάστατο πίνακα με το όνομα **lykeia** και σε έναν άλλο παράλληλο, επίσης μονοδιάστατο πίνακα, με το όνομα **mathites** καταχωρείται ο αριθμός των μαθητών κάθε Λυκείου. Να δημιουργήσετε πρόγραμμα στη γλώσσα προγραμματισμού C++, το οποίο:

- α) να καταχωρεί στο μονοδιάστατο πίνακα **lykeia** το όνομα του κάθε Λυκείου και στον παράλληλο μονοδιάστατο πίνακα **mathites** τον αριθμό των μαθητών που έχουν εγγραφεί στο συγκεκριμένο Λύκειο. **(Βαθμοί 3 - με δήλωση μεταβλητών και βιβλιοθηκών)**
- β) να υπολογίζει και να τυπώνει τον μέσο όρο των μαθητών όλων των Λυκείων (με ένα δεκαδικό) και ακολούθως να υπολογίζει και να τυπώνει το πλήθος των λυκείων που έχουν αριθμό μαθητών πάνω από τον μέσο όρο. **(Βαθμοί 3)**
- γ) να βρίσκει και να τυπώνει το όνομα του Λυκείου με τους περισσότερους μαθητές και τον αριθμό των μαθητών του, καθώς επίσης το όνομα του Λυκείου με τους λιγότερους μαθητές και τον αριθμό των μαθητών του. Να υποθέσετε ότι μόνο ένα Λύκειο έχει τους περισσότερους μαθητές και μόνο ένα Λύκειο έχει τους λιγότερους μαθητές. **(Βαθμοί 3)**
- δ) να ζητά από τον χρήστη το όνομα ενός Λυκείου και ακολούθως να τυπώνει τον αριθμό των μαθητών που έχει το συγκεκριμένο λύκειο. **(Βαθμοί 3)**
- ε) να παρουσιάζει τα περιεχόμενα των πινάκων **lykeia** και **mathites**, όπως στο παράδειγμα εξόδου πιο κάτω: **(Βαθμοί 3)**

Το πρόγραμμα πρέπει να εμφανίζει στην οθόνη τα κατάλληλα μηνύματα για την εισαγωγή των δεδομένων και την εξαγωγή των αποτελεσμάτων.

<u>Παράδειγμα Εξόδου για 5 Λύκεια:</u>	
Όνομα Λυκείου:	Λύκειο_yy
Αριθμός μαθητών:	450
Όνομα Λυκείου:	Λύκειο_xx
Αριθμός μαθητών:	650
Όνομα Λυκείου:	Λύκειο_zz
Αριθμός μαθητών:	500
Όνομα Λυκείου:	Λύκειο_mm
Αριθμός μαθητών:	700
Όνομα Λυκείου:	Λύκειο_nn
Αριθμός μαθητών:	400
Μέσος όρος μαθητών: 540.0	
Πιο πολλούς μαθητές: Λύκειο_mm	
Αριθμός μαθητών: 700	
Πιο λίγους μαθητές: Λύκειο_nn	
Αριθμός μαθητών: 400	
Λύκεια πάνω από το μέσο όρο: 2	
Δώσε όνομα Λυκείου:	Λύκειο_zz
Αριθμός μαθητών: 500	
Λύκειο	Μαθητές
Λύκειο_yy	450
Λύκειο_xx	650
Λύκειο_zz	500
Λύκειο_mm	700
Λύκειο_nn	400

```

#include <iostream>
#include <string>
#include <iomanip>
#define N 38
using namespace std;
int main() {
    string lykeia[N], maxLyk, lykToFind;
    int mathites[N], i, maxMath = -1, minMath, minPos;
    int sum = 0, lykAboveAv = 0;
    float moMath;
    for (i=0; i < N; i++) {
        cout << "Όνομα Λυκείου: ";
        cin >> lykeia[i];
        cout << "Αριθμός μαθητών: ";
        cin >> mathites[i];
        sum += mathites[i];
    }
    moMath = (float) sum / N;
    cout << "Μέσος όρος μαθητών: " << fixed << setprecision(1) << moMath << endl;

    minMath = mathites[0];
    minPos = 0;
    for (i=0; i < N; i++) {
        if (mathites[i] > maxMath) {
            maxMath = mathites[i];
            maxLyk = lykeia[i];
        }
        if (mathites[i] < minMath) {
            minMath = mathites[i];
            minPos = i;
        }
        if (mathites[i] > moMath)
            lykAboveAv++;
    }

    cout << "Πιο πολλούς μαθητές: " << maxLyk << endl;
    cout << "Αριθμός μαθητών: " << maxMath << endl;
    cout << "Πιο λίγους μαθητές: " << lykeia[minPos] << endl;
    cout << "Αριθμός μαθητών: " << mathites[minPos] << endl;
    cout << "Λύκεια πάνω από το μέσο όρο: " << lykAboveAv << endl;

    cout << "Δώσε όνομα Λυκείου: ";
    cin >> lykToFind;
    for (i=0; i < N; i++)
        if (lykeia[i] == lykToFind)
            cout << "Αριθμός Μαθητών: " << mathites[i] << endl;
    cout << setw(25) << "Λύκειο" << setw(25) << "Μαθητές" << endl;
    for (i=0; i < N; i++)
        cout << setw(25) << lykeia[i] << setw(25) << mathites[i] << endl;
    return 0;
}

```

1 Για κάθε λάθος: -0.25μ (max -1μ)

2

0.75

0.75

0.5

0.5

0.75

0.75

0.75

0.25

0.25

0.25

0.25

0.25

0.75

2

0.25

0.5

2.5

Αφαιρετικά: Αν λείπει έστω και ένα ; : - 0.25 μια φορά
 Αν λείπει έστω και ένα endl : - 0.25 μια φορά
 Αν λείπει έστω και ένα ή είναι λάθος { ή } : - 0.25 μια φορά
 Αν λείπουν τα μηνύματα ή είναι λάθος : - 0.5 μια φορά

Άσκηση 2

Το Τμήμα αποχετεύσεως ενός Δήμου πρόκειται να μηχανογραφηθεί. Κατά τη φάση της Εξακρίβωσης Αναγκών και Καθορισμού Απαιτήσεων έχουν εξακριβωθεί τα πιο κάτω χαρακτηριστικά, τα οποία πρέπει να διαθέτει ένα μέρος του Πληροφοριακού Συστήματος:

Κάθε δημότης που ολοκληρώνει τις εργασίες κατασκευής του αποχετευτικού στο υποστατικό του θα υποβάλει ηλεκτρονική αίτηση έγκρισης στο Δήμο δίνοντας τον Αριθμό Ρολογιού νερού που έχει το υποστατικό του. Με βάση τον Αριθμό Ρολογιού νερού θα γίνεται έλεγχος των στοιχείων του δημότη (Αριθμός Ταυτότητας, Ονοματεπώνυμο, Διεύθυνση) από το ήδη υφιστάμενο αρχείο Υποστατικών και θα καταχωρείται η αίτησή του στο αρχείο Αιτήσεων. Η κάθε αίτηση θα περιλαμβάνει έναν κωδικό αίτησης ο οποίος δημιουργείται αυτόματα από το σύστημα, τον αριθμό ρολογιού νερού, τον αριθμό ταυτότητας δημότη και την ημερομηνία της αίτησης (τρέχουσα ημερομηνία που καταχωρείται αυτόματα από το σύστημα). Αμέσως μετά θα δημιουργείται μια βεβαίωση αίτησης την οποία θα παίρνει τόσο ο Δημότης όσο και η Γραμματεία του Δήμου. Η βεβαίωση θα περιλαμβάνει τον Κωδικό Αίτησης, το Ονοματεπώνυμο Δημότη, τον Αριθμό Ρολογιού Νερού και την Ημερομηνία Αίτησης.

Σε μεταγενέστερο στάδιο ο Δήμος θα ορίζει την ημερομηνία επιθεώρησης για έλεγχο και έγκριση της κατασκευής και τον λειτουργό που θα την αναλάβει. Η Γραμματεία του Δήμου θα δίνει τον κωδικό αίτησης (με βάση τον οποίο θα ανακτώνται από το αρχείο αιτήσεων τα στοιχεία της αίτησης), την ημερομηνία επιθεώρησης και το ονοματεπώνυμο του λειτουργού που θα αναλάβει να κάνει τον έλεγχο, τα οποία θα καταχωρούνται και αυτά στο αρχείο Αιτήσεων. Στη συνέχεια θα δημιουργείται μια επιστολή Επιθεώρησης, η οποία θα περιλαμβάνει τον Κωδικό Αίτησης, τον Αριθμό Ταυτότητας, το Ονοματεπώνυμο δημότη, την Διεύθυνση και την ημερομηνία επιθεώρησης, η οποία θα αποστέλλεται στον δημότη και θα δίνεται και στον λειτουργό του Δήμου.

Ερώτημα (α) (Βαθμοί 10)

Να σχεδιάσετε το Διάγραμμα Ροής Δεδομένων (ΔΡΔ) για τη λειτουργία που περιγράφεται πιο πάνω. (Η απάντηση να δοθεί στην επόμενη σελίδα)

Διαδικασίες	: 4	μονάδες	(4 * 1)
Οντότητες	: 1.5	μονάδες	(3 * 0.5)
Αρχεία	: 1	μονάδα	(2 * 0.5)
Ροές	: 3.5	Κάθε ροή που θα κάνει λάθος ή ξεχάσει ή χωρίς δεδομένα:	- 0.25 (max -3.5μ)
Λάθος σχήματα:	-0.5	(μια φορά για κάθε είδος)	
Αν λείπει η αρίθμηση:	- 0.25		

**** Οι διαδικασίες 14 και 18 μπορούν να είναι μία. Σε αυτή την περίπτωση η διαδ. θα πάρει 2μ.**

Ερώτημα (β) (Βαθμοί 5)

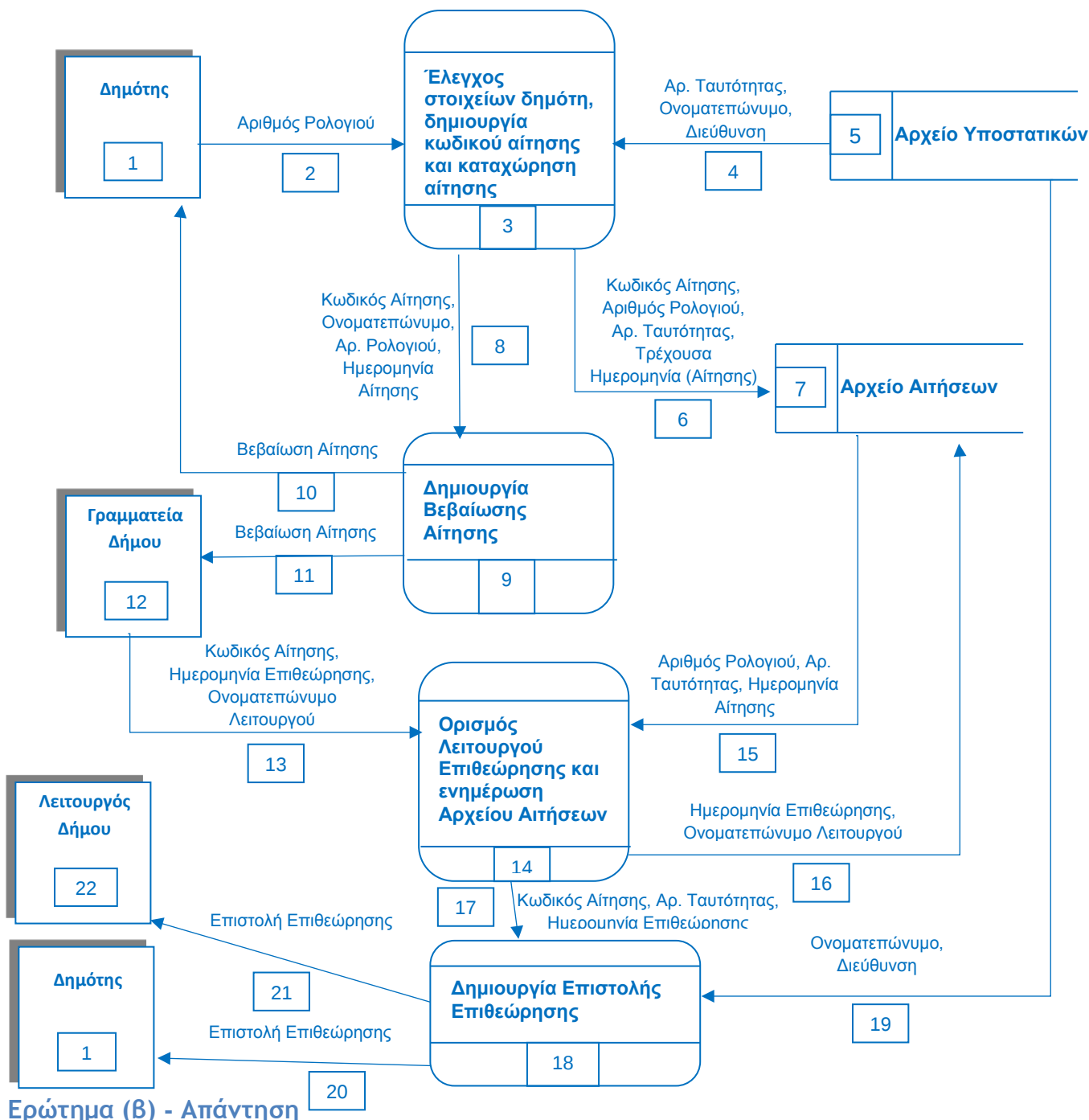
- (1) Να καθορίσετε τις δύο (2) αποθηκευτικές μονάδες (πίνακες), το αρχείο Υποστατικών και το αρχείο Αιτήσεων (όνομα, πρωτεύον κλειδί, ξένο κλειδί και πεδία).
- (2) Να σχεδιάσετε την σχέση μεταξύ των δύο πινάκων και να σημειώσετε την πληθυκότητα (cardinality).

(Η απάντηση να δοθεί στην επόμενη σελίδα)

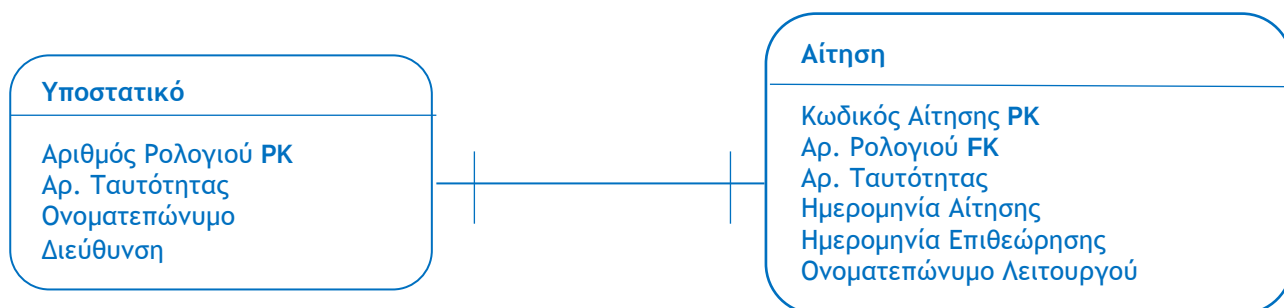
(β) (1) 3 μονάδες κάθε λάθος -0.5μ (max -3μ)

(2) 2 μονάδες κάθε λάθος -0.5μ (max -2μ)

Ερώτημα (α) - Απάντηση



Ερώτημα (β) - Απάντηση



ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ

Ο Διευθυντής

Νίκος Πρωτοπαπός

ΠΡΟΧΕΙΡΟ

ΓΡΑΠΤΕΣ ΠΡΟΑΓΩΓΙΚΕΣ ΕΞΕΤΑΣΕΙΣ

Λύσεις**ΜΑΪΟΥ 2018**

Μάθημα: ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΕΠΙΣΤΗΜΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
Τάξη: Β΄
ΗΜΕΡΟΜΗΝΙΑ: 25/05/2018
ΧΡΟΝΟΣ: 2.5 ΩΡΕΣ
ΩΡΑ: 7:45-10:15

ΟΔΗΓΙΕΣ

- Το εξεταστικό δοκίμιο αποτελείται από δεκαοχτώ (18) σελίδες.
- Το εξεταστικό δοκίμιο αποτελείται από τρία μέρη Α΄, Β΄ και Γ΄.
- Το μέρος Α΄ αποτελείται από έξι (6) ερωτήσεις και κάθε ερώτηση βαθμολογείται με πέντε (5) μονάδες.
- Το μέρος Β΄ αποτελείται από τέσσερις (4) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δέκα (10) μονάδες.
- Το μέρος Γ΄ αποτελείται από δύο (2) ερωτήσεις και κάθε ερώτηση βαθμολογείται με δεκαπέντε (15) μονάδες.
- Όλες οι απαντήσεις σας να γραφούν στο φύλλο εξέτασης που θα σας δοθεί.
- Επιτρέπεται η χρήση μη προγραμματιζόμενης υπολογιστικής μηχανής.
- Τα Λογικά Διαγράμματα και τα περιεχόμενά τους μπορούν να γίνουν με μολύβι.
- Να γράφετε μόνο με μπλε στυλό.
- Κατοχή κινητού τηλεφώνου = Δολίευση.
- Απαγορεύεται η χρήση διορθωτικού υγρού.
- Οι μοναδικές βιβλιοθήκες που επιτρέπονται στη δημιουργία προγραμμάτων, είναι η <iostream> και η <cmath>, εκτός αν αναφέρεται διαφορετικά στην εκφώνηση.
- Όλες οι απαντήσεις σας να γραφούν στις κόλλες εξέτασης που θα σας δοθούν.

ΚΑΛΗ ΕΠΙΤΥΧΙΑ

ΜΕΡΟΣ Α'

Άσκηση 1

Για το πιο κάτω λογικό διάγραμμα, να γράψετε το αντίστοιχο πρόγραμμα σε C++ και να υλοποιήσετε τα ακόλουθα:

Ερώτημα (α) (Βαθμός 1.75)

Να δηλώσετε τις απαραίτητες βιβλιοθήκες και τις κατάλληλες μεταβλητές.

`#include <iostream>` 0.25

`#include <cmath>` 0.25

`float a,b,c,res;` 1.25

Ερώτημα (β) (Βαθμοί 3.25)

Να χρησιμοποιήσετε τις κατάλληλες συναρτήσεις.

`cin>>a;` 0.5

`cin>>b;` 0.5

`c=sqrt((pow(b,2)-4*a));` 1

`res=c*(a+b);` 0.5

`cout<<c<<endl<<res;` 0.75

Άσκηση 2

Ερώτημα (α) (Βαθμοί 2)

Να προσθέσετε τις κατάλληλες εντολές, ώστε να εμφανίζεται το μήνυμα «Οι Αριθμοί δεν είναι ίσοι», όταν οι αριθμοί δεν είναι ίσοι.

`if (num1 !=num2)` 1β

`cout << "Οι Αριθμοί δεν είναι ίσοι"<<endl;` 1β

Ερώτημα (β) (Βαθμοί 3)

Να προσθέσετε τις κατάλληλες εντολές ώστε να εμφανίζεται το μήνυμα «Ο πρώτος αριθμός είναι περιττός», αν ο αριθμός num1 είναι περιττός.

```
If (num1 %2 != 0)                2β  
    cout << "Ο πρώτος αριθμός είναι περιττός" <<endl;  1β
```

Άσκηση 3

Ερώτημα (α) (Βαθμοί 1)

Να δηλώσετε τις βιβλιοθήκες που θα χρειαστούν για να τρέξει το πρόγραμμα.

```
# include <cmath>    0.5  
# include <iomanip>   0.5
```

Ερώτημα (β) (Βαθμοί 1)

Να δηλώσετε το αναγνωριστικό pi ως σταθερά.

```
#define pi 3.14159 ή const float pi = 3.14159;  1
```

Ερώτημα (γ) (Βαθμοί 1)

Να γράψετε την εντολή ή τις εντολές δήλωσης των απαραίτητων μεταβλητών.

```
float r, emvaddon;  1
```

Ερώτημα (δ) (Βαθμοί 2)

Να τροποποιήσετε την εντολή

```
cout<< "emvaddon=" << emvaddon ;
```

ώστε το εμβαδόν να εμφανίζεται με ακρίβεια δύο (2) δεκαδικών ψηφίων.

```
cout << "Emvaddon=" <<fixed<<setprecision(2)<<emvaddon<<endl;  2
```

Άσκηση 4

Ερώτημα (α) (Βαθμοί 1)

Να γράψετε το αποτέλεσμα της πιο κάτω έκφρασης, η οποία είναι γραμμένη στη γλώσσα προγραμματισμού C++:

```
cout<<pow(4,2) + round(3.5) * sqrt(25) – trunc(3.6) + abs(-4)
```

16 + 20 -3 +4 = 37

Ερώτημα (β) (Βαθμοί 2)

Να γράψετε τα αποτελέσματα των πιο κάτω λογικών εκφράσεων, οι οποίες είναι γραμμένες στη γλώσσα προγραμματισμού C++ αν a=4, b=1 και c=3:

i. $((c-1) == 2) \parallel (b \geq (a/c))$ **True** **1**

ii. $((a+c) == (a-c+b)) \&\& ((a\%c) == b)$ **False** **1**

Ερώτημα (γ) (Βαθμοί 2)

Να μετατρέψετε την πιο κάτω μαθηματική έκφραση στη γλώσσα προγραμματισμού C++.

$$x = \frac{|-b| + \sqrt{b^2 - 4a}}{2} / 2a$$

x= ((abs(-b)+sqrt(pow(b,2)-4*a)/2)/2*a **2**

Άσκηση 5:

Δίνεται το πιο κάτω λογικό διάγραμμα.

Ερώτημα (α) (Βαθμοί 1)

Ποιο θα είναι το αποτέλεσμα, αν δοθεί αρχικά η τιμή για το $Y=2$;

$X=13$ 0.5

$Y=3$ 0.5

Ερώτημα (β) (Βαθμοί 2)

Αν δοθεί αρχικά και πάλι η τιμή για το $Y=2$ και επιπρόσθετα αλλάξουμε στο αρχικό λογικό διάγραμμα την εντολή $X \leftarrow X + 5$ σε $X \leftarrow X - 3$ θα δημιουργηθεί ένα ανεπιθύμητο πρόβλημα. Ποιο είναι αυτό το πρόβλημα;

Infinite loop 2

Ερώτημα (γ) (Βαθμοί 2)

Το πιο πάνω λογικό διάγραμμα έχει μεταφραστεί στο πιο κάτω πρόγραμμα στη C++. Στο πρόγραμμα αυτό υπάρχουν δύο (2) λάθη. Να εντοπίσετε και να διορθώσετε τα λάθη αυτά αναφέροντας τον αριθμό της γραμμής στην οποία εμφανίζεται το κάθε λάθος και να γράψετε τη εντολή διορθωμένη, υπογραμμίζοντας το σημείο που έγινε η διόρθωση.

```
[1] #include<iostream>
[2] using namespace std;
[3] int main(){
[4] int x;
[5] cin>>y;
[6] x=y*4;
[7] if (x<=9){
[8] y=y+1;
[9] x=x+5;
[10] }
[11] cout<<x<<" "<<y;
[12] return 0;
[13] }
```

ΛΑΘΗ

[4] int x,y;	1
[7] <u>while</u> (x<=9) {	1

Άσκηση 6

Ερώτημα (α) (Βαθμοί 2)

Σε ποια ηλικιακή κατηγορία της PEGI (Pan European Game Information) θα κατατάσσατε το πιο πάνω παιχνίδι και γιατί;

PEGI 7 γιατί αν και παιδικό παιχνίδι, περιέχει κάποιες πιθανώς τρομακτικές σκηνές και ήχους.

Ερώτημα (β) (Βαθμοί 1)

Καθορίζοντας ταχύτητα μεγαλύτερη του 0.

Ερώτημα (β) (Βαθμοί 2)

Να συμπληρώσετε το παρακάτω πίνακα με την Περιγραφική Ένδειξη.

Επισήμανση	Περιγραφική Ένδειξη
A) 	Σεξ
B) 	Διακρίσεις
Γ) 	Φόβος
Δ) 	Βία

0.5 για το καθένα

ΜΕΡΟΣ Β'

Άσκηση 1

Ερώτημα (α) (Βαθμοί 2)

`int photos;` 0.5

`float xr;` 0.5

`cin>>photos` 1

Ερώτημα (β) (Βαθμοί 5)

`if (photos>=1 && photos<=10)` 0.5 β

`xr= photos * 12;` 0.5 β

`else if (photos>=11 && photos<=30)` 0.5 β

`xr= 10 * 12 + (photos -10)* 10.50;` 1β

`else if (photos >30)` 0.5 β

`xr= 10 * 12 + 20 * 10.50 + (photos-30) * 9.25;` 1.5β

`cout <<"XREOSI: "<<xr<<endl;` 1

Ερώτημα (γ) (Βαθμοί 3)

Δήλωση Σταθεράς

`# define disc 0.20 ή const float disc=0.20;` 0.5β

`float discposo =0;` 0.25β

`if (xr>150)` 0.5β

`{ discposo = xr*disc;` 0.5β

`float final = xr -discposo;` 0.5β

`cout <<"DISCOUNT: "<<discposo<<endl;` 0.25β

`cout << "FINAL : "<<final;` 0.25β

`}` 0.25 για { }

Άσκηση 2

Ερώτημα (α) (Βαθμοί 2)

```
int num;                0.25 β
cin >> num;             0.25 β
while (num !=0) {       1β
    cin >> num;          0.25β
}
- 0.25 για { }
```

Ερώτημα (β) (Βαθμοί 4)

```
#include <cmath>         0.5β
if (num>=10 && num<=99) { 1β
    int num2= num%10*10+num/10; 1.5β
    cout << num2<<" ";      0.5β
    cout << "SquareRoot: "<<sqrt(num2)<<endl; 0.5β
}
-0.25 για παράλειψη { }
```

Ερώτημα (γ) (Βαθμοί 4)

```
int sum=0, c=0;          1β

else                      0.25β
{
    sum+=num;             0.75β
    c++;                  0.75β
}

cin >> num;

}

float avg=float(sum)/c;   0.75β
cout <<"AVERAGE: "<<avg<<endl; 0.5β
```

Άσκηση 3

Ερώτημα (α) (Βαθμοί 3)

Ποια είναι τα αντικείμενα (objects), τα οποία χρειάζεται να δημιουργήσετε για το πιο πάνω παιχνίδι.

α) Αράχνη β) Αρκουδάκι γ) Διαμάντι δ) Σημαία ε) Κόκκινο Τούβλο στ) Τούβλο τοίχου

0.5 για το καθένα

Ερώτημα (β) (Βαθμοί 3)

Να σημειώσετε ποιες από τις πιο κάτω επιλογές από τα events και τα actions θα πρέπει να διαλέξετε έτσι ώστε, όταν το αρκουδάκι ακουμπήσει ένα διαμάντι να ακούγεται ένας ήχος και να προστίθενται 10 βαθμοί στους βαθμούς .

Events: A,

Collision Actions: A και Γ

1 βαθμό για το καθένα

Ερώτημα (γ) (Βαθμοί 1)

Πρέπει να επιλεγεί η επιλογή Relative.

Ερώτημα (δ) (Βαθμοί 2)

Events: 4 (1 β.) Collision Actions: 5 και 1 (0.5 το καθένα)

Ερώτημα (ε) (Βαθμοί 1)

To A

Άσκηση 4

Ερώτημα (α) (Βαθμοί 4)

Να αναφέρετε τέσσερα από τα λάθη που υπάρχουν στο διάγραμμα και να περιγράψετε τον τρόπο διόρθωσης τους.

- A) Στη ροή 4 απουσιάζει το δεδομένο Κωδικός Πελάτη
- B) Η ροή 6 πρέπει να φεύγει από την διαδικασία 3 και όχι από το αρχείο 5.
- Γ) Από το αρχείο 5 έπρεπε να υπάρχει ροή δεδομένων του επιθέτου, ονόματος και τηλ πελάτη για να μπορεί να γίνει ο έλεγχος πελάτη.
- Δ) Στη διαδικασία 12 απουσιάζει ροή δεδομένων από το Αρχείο πελατών με τα προσωπικά στοιχεία του πελάτη
- Ε) Από τη διαδικασία 12 έπρεπε να εξέρχεται ροή δεδομένων(επιστολή) προς τον πελάτη.

Ερώτημα (β) (Βαθμοί 3)

Ποιοι πίνακες χρειάζονται για την υλοποίηση του πιο πάνω ΔΡΔ. Για τον κάθε πίνακα να δηλώσετε:

- A) το όνομα του
- B) τα πεδία του
- Γ) το πρωτεύον κλειδί του

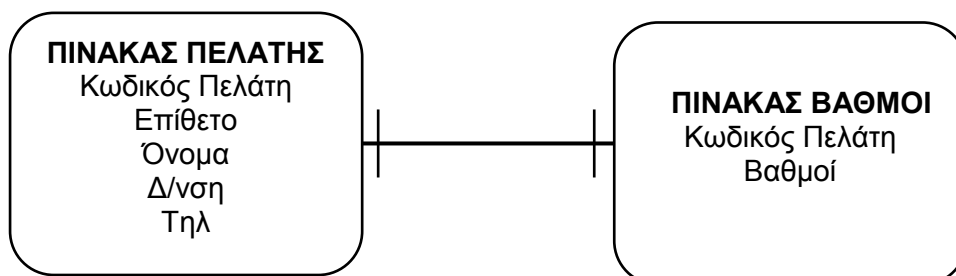
A) Πίνακες: Αρχείο Πελάτη και Αρχείο Βαθμών 0.5β

B) Πεδία Πελάτη: Κωδικός Πελάτη, Επίθετο, Όνομα, Δ/ση, Τηλ 2β
Πεδία Βαθμών: Κωδικός Πελάτη και Βαθμοί

Γ) Πρωτεύον Κλειδί και για τους 2 πίνακες: Κωδικός Πελάτη 0.5β

Ερώτημα (γ) (Βαθμοί 1)

Να σχεδιάσετε το διάγραμμα σχέσεων οντοτήτων(ΣΔΟ) για τους πιο πάνω πίνακες



Ερώτημα (δ) (βαθμοί 2)

Field:	Επίθετο	Όνομα	Τηλ	Βαθμοί
Table:	ΠΕΛΑΤΗΣ	ΠΕΛΑΤΗΣ	ΠΕΛΑΤΗΣ	ΒΑΘΜΟΙ
Sort:				Descending 
Show:	☒	☒	☒	☒
Criteria:				>1000
or:				

1β για πεδία - 0.25 για κάθε πεδίο
0.5 για πίνακες – 0.25 για κάθε πίνακα
0.25 για ταξινόμηση
0.25 για κριτήριο

ΜΕΡΟΣ Γ'

Άσκηση 1

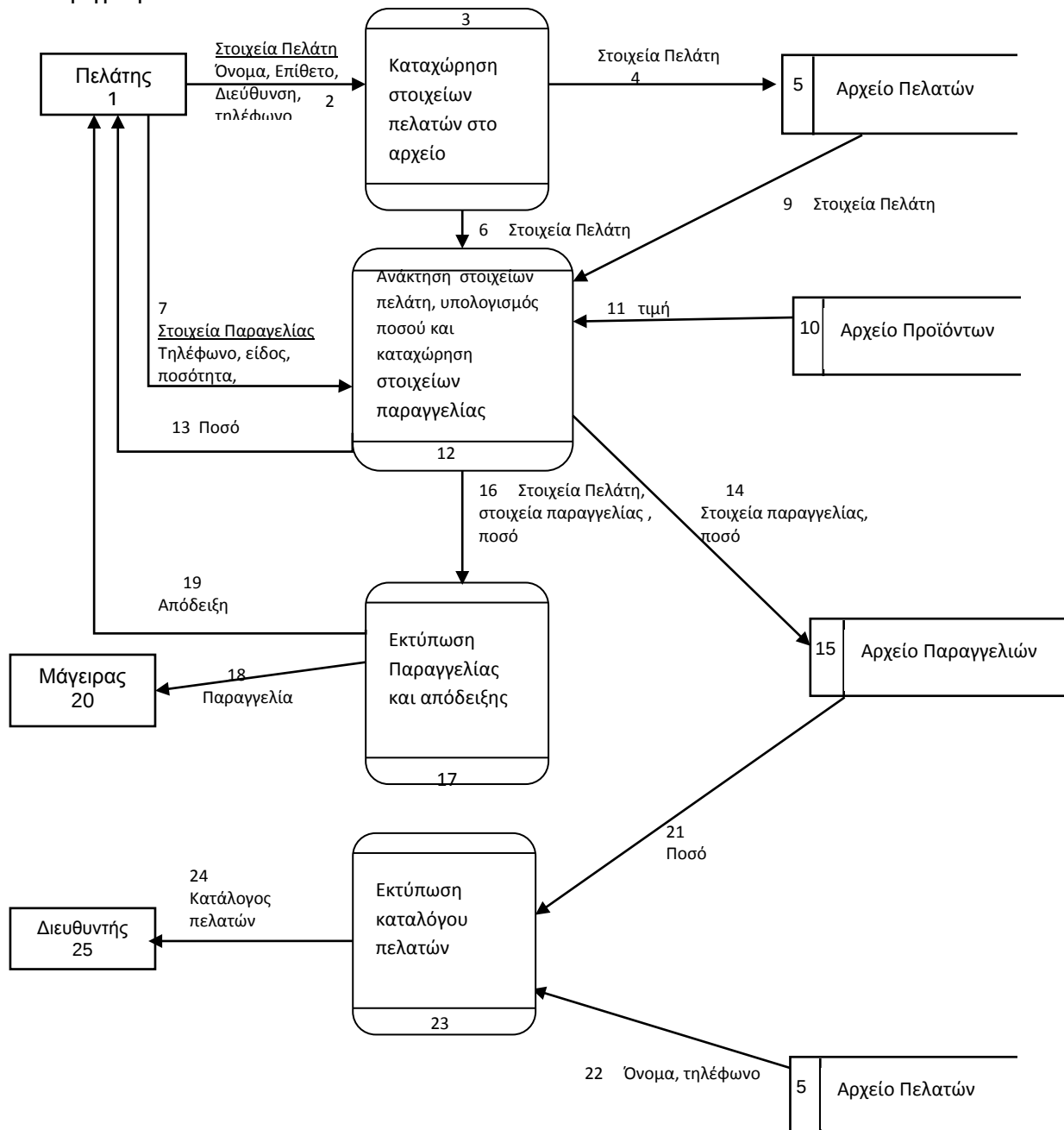
```
#include <iostream>      (0.25)
#include <string>         (0.5)
#include <iomanip>        (0.5)
Using namespace std;    (0.25)
int main (){            (0.25)
    string xores[28];    (0.25)
    int psifoi[28];      (0.25)
    int max=0, p=0, i;   (0.5)
    string maxN;         (0.25)
    for ( i=0; i<28; i++) (1.25) =(0.5 for)+ (0.75 = 0.25+.025+0.25)
        cin >> xores[i] >>psifoi[i]; (0.75= 0.25+.025+0.25)
        if psifoi[i]>max{ (0.75= if 0.25+.025+0.25)
            max = psifoi[i]; (0.5)
            maxN = xores[i]; (0.5)
        } (0.25 τα δύο άγκιστρα)
        if psifoi[i] == 0 (1.5) (if 0.5)+(psifoi[i] 0.5)+ (== 0.5)
            p++; (0.5)
    cout << "Νικήτρια χώρα" << maxN << "με" << max << "ψηφους" << endl; (1)
    cout << "το πλήθος των χωρών με 0 βαθμούς είναι:" << p << endl; (1)
    cout <<setw(20) <<"Χώρες"<<setw(20) <<"Ψήφοι" << endl; (1.5)
    for (i=0; i<28; i++) (1.25) =(0.5 for)+ (0.75 = 0.25+.025+0.25)
        cout <<setw(20) <<xores[i]<<setw(20) <<psifoi[i] << endl;
                                                    (1= 0.5 setw, 0.25 endl, )

    return 0; (0.25)
}
```

Άσκηση 2

Ερώτημα (α) (βαθμοί 10)

Να σχεδιάσετε το Διάγραμμα Ροής Δεδομένων για το πληροφοριακό σύστημα που περιγράφεται πιο πάνω.



10 βαθμοί

Διαδικασίες (1 β * 4)

Αρχεία (0.5 β * 3)

Οντότητες (0.5 β * 3)

Ροές (0.25 β * 13) σύνολο 3 β

Ερώτημα (β) (βαθμοί 3)

Η υλοποίηση του πιο πάνω συστήματος έχει ένα κύκλο ζωής με 6 φάσεις.

- i. Να δηλώσετε σε ποια φάση σχεδιάστηκε το πιο πάνω διάγραμμα.
Καθορισμός προδιαγραφών
- ii. Να ονομάσετε τις φάσεις που ακολουθούν μετά τη φάση Καθορισμός Προδιαγραφών.
Σχεδίαση, Υλοποίηση, Συντήρηση

(0.75 η κάθε φάση)

Ερώτημα (γ) (βαθμοί 2)

Να περιγράψετε τι γίνεται κατά τη φάση της Σχεδίασης. (0.5 β το καθένα)

1. Καθορίζονται τα μέρη που συστήματος και Λεπτομερής σχεδιασμός των μερών με χρήση αλγορίθμων
2. Σχεδιασμός Βάσεων δεδομένων
3. Σχεδιασμός οθονών
4. Σχεδιασμός Αναφορών