

ΜΕΡΟΣ 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)

```
float a,b,c,p;
```

Ερώτημα (β)

```
float e;  
e=a*b/2;  
cout<<"Εμβαδόν="<<e<<endl;
```

Ερώτημα (γ)

```
#include <iomanip>  
cout<<"Perimetros="<<fixed<<setprecision(2)<<p<<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 και το λογικό διάγραμμα της δομής διακλάδωσης If..Else.

Λύση

Ερώτημα (α)

```
avg=(temp1+temp2)/2.0; ή avg=float(temp1+temp2)/2;
```

Ερώτημα (β)

```
else  
    cout<<"Temperature OK";
```

Ερώτημα (γ)

```
(avg>35) || (temp1>40) || (temp2>40)
```

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)

S=9

Ερώτημα (β)

S=30

Ερώτημα (γ)

S=18

Άσκηση 4

Δείκτες Επιτυχίας

B7.6.6 Σχεδιάζουν το λογικό διάγραμμα και τις αντίστοιχες εντολές προγράμματος που να ανταποκρίνονται σε συγκεκριμένη εκφώνηση που αναφέρεται σε δομή επανάληψης.
B7.6.12 Ελέγχουν την ορθότητα της λύσης του προβλήματος χρησιμοποιώντας τη μέθοδο της προκαταρκτικής εκτέλεσης και της χρήσης του debugger για επαλήθευση.
B7.6.13 Μελετούν έτοιμο πρόγραμμα το οποίο περιλαμβάνει δομές επανάληψης και να εντοπίζουν βασικά μέρη του τα οποία συνδέονται με πτυχές του προβλήματος που επιλύει.

Λύση

Ερώτημα (α)

k=6

Ερώτημα (β)

A: $i \leftarrow 6$

B: $i \geq 0$

Γ: $k > 4$

Δ: $i \leftarrow i - 2$

Ερώτημα (γ)

k=6

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.

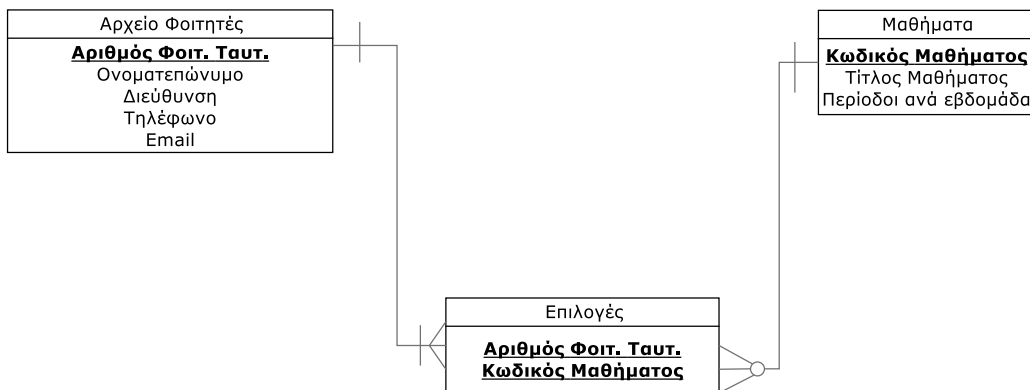
B6.8.14 Δημιουργούν ερωτήματα, επιλέγοντας πεδία από έναν ή περισσότερους σχετιζόμενους πίνακες και εφαρμόζουν κριτήρια και ταξινόμηση, σύμφωνα με τις απαιτήσεις της σχεδίασης του πληροφοριακού συστήματος.

Λύση

Ερώτημα (α)

- Παρατήρηση στον τόπο εργασίας (Observation)
- Συνεντεύξεις (Interviews)
- Συλλογή και Μελέτη Έντυπου Υλικού
- Γρήγορη Πρωτοτυποποίηση

Ερώτημα (β)



Ερώτημα (γ)

Field:	AFT	Onomateponimo	Kodikos_Mathimatos	Vathmos
Table:	Foitites	Foitites	Vathmoi	Vathmoi
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				<5
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.

Λύση

Ερώτημα (α)

PEGI 3

Ερώτημα (β)

Η ταχύτητα (speed) πρέπει να είναι μεγαλύτερη από 0.

Ερώτημα (γ)

Collision

ΜΕΡΟΣ Β'

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 αυξομείωσης (++ 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>
using namespace std;
int main(){
    int paidia[101]={};
    int i,N,M,S;
    bool flag=true;
    cin>>N>>M>>S;
    i=S;
    while(M>0){
        if(i>N)
            i=1;
        paidia[i]++;
        M--;
        i++;
    }
    cout<<"Last chocolate:"<<i-1<<endl;

    cout<<"No chocolate:";
    for (i=1;i<=N;i++)
        if(paidia[i]==0){
            cout<<i<<" ";
            flag=false;
        }
    if(flag)
        cout<<0<<endl;
    else
        cout<<endl;

    cout<<"Position/ Chocolates"<<endl;
```

```
    for (i=1;i<=N;i++)  
        cout<<i<<" "<<paidia[i]<<endl;  
  
    return 0;  
}
```


Άσκηση 2

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>
#include<cmath>
using namespace std;
int main(){
    int a,b,x,y,pA=1,d,dA,dB,jumps=0;
    cin >> a >> x >> b >> y;
    pA=1000 / x+1;
    cout<<"Frog A makes "<<pA<<" jumps"<<endl;
    d=(1000*x+a)-(1000*y+b);
    cout << "Distance after 1000 jumps: "<<abs(d)<<endl;
    dA=a;
    dB=b;
    while((dA!=dB)&&(jumps<=10000)){
        dA+=x;
        dB+=y;
        jumps++;
    }
    if(jumps>10000)
        cout << "No meeting point" << endl;
    else
        cout << "They meet after " << jumps << " jumps" << endl;
    return 0;
}
```

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)

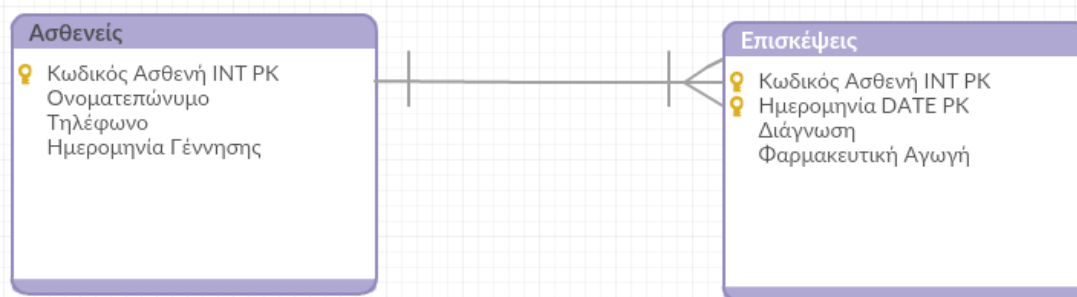
Πίνακας Ασθενείς	
Κωδικός Ασθενή	Number
Όνοματεπώνυμο	Text
Τηλέφωνο	Text
Ημερομηνία γέννησης	Date/Time

Πρωτεύον Κλειδί: Κωδικός Ασθενή

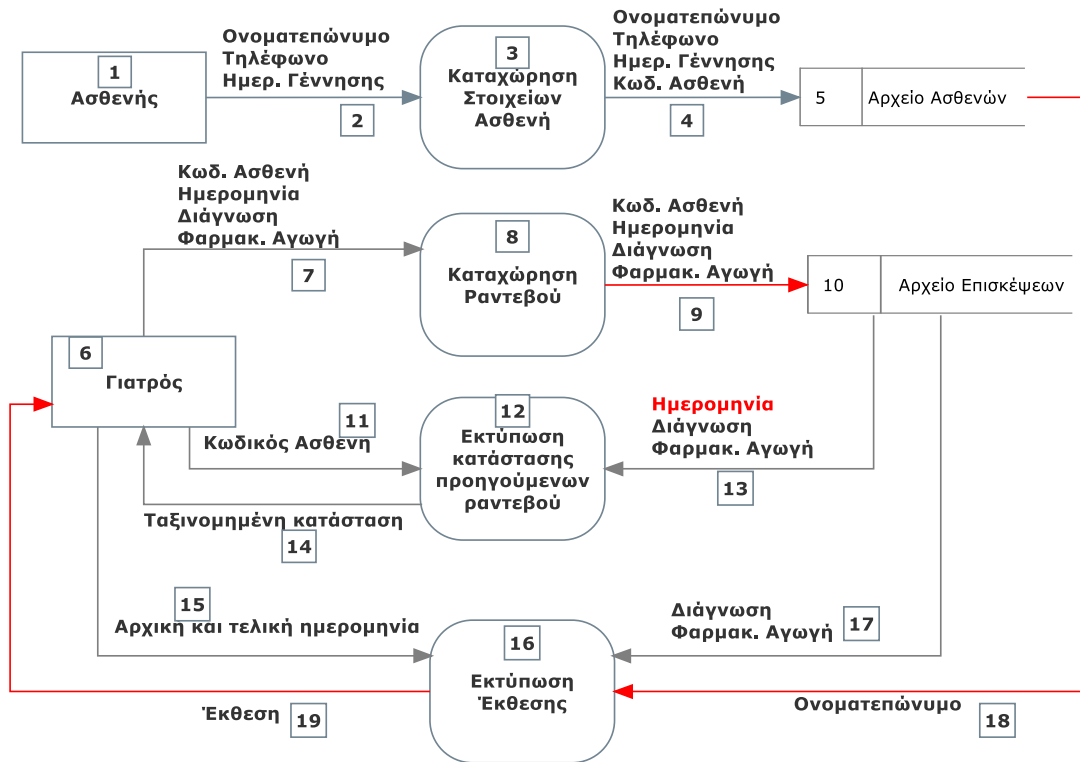
Πίνακας Επισκέψεις	
Κωδικός Ασθενή	Number
Ημερομηνία	Date/Time
Διάγνωση	Text
Φαρμακευτική Αγωγή	Text

Πρωτεύον Κλειδί: Κωδικός Ασθενή + Ημερομηνία

Ερώτημα (β)



Ερώτημα (γ)



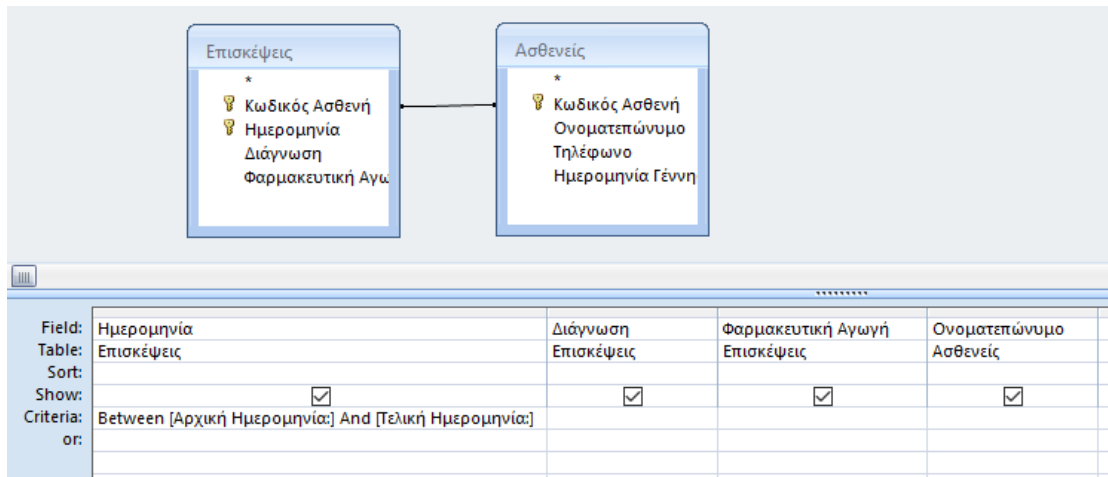
Ερώτημα (δ)

Επισκέψεις

- * Κωδικός Ασθενή
- * Ημερομηνία
- Διάγνωση
- Φαρμακευτική Αγω

Field:	Ημερομηνία	Διάγνωση	Φαρμακευτική Αγωγή	Κωδικός Ασθενή
Table:	Επισκέψεις	Επισκέψεις	Επισκέψεις	Επισκέψεις
Sort:	Ascending			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:				[Κωδικός Ασθενή:]
or:				

(ι)



(ι)

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)

Τα αντικείμενα (objects) που χρειάζονται είναι: ο σκύλος, η αράχνη, το κόκκαλο, η φωτιά και ο τοίχος.

Ερώτημα (β)

Πρέπει να εισάγουμε το action **Set the score relative to -5**, όταν ο σκύλος θα κάνει collision με την αράχνη.

Ερώτημα (γ)

Event: 1 Create **Actions:** 3 Set Alarm

Event: 6 Alarm **Actions:** 5 Display a message, 2 End the game

Ερώτημα (δ)

Event: 3 Step

Actions:

4 Test Instance Count



Start of a block

5 Display a message

2 End the game



End of a block

ΜΕΡΟΣ Γ'

Άσκηση 1

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

Λύση

Ερωτήματα (α) - (δ)

```
#include<iostream>
using namespace std;
int main(){
    int n,i,T=0,jumps=0;
    cin>>n;
    int A[n];
    for(i=0;i<n;i++){
        cin>>A[i];
        if(A[i]==1)
            T++;
    }
    i=0;
    while(i!=n-1){
        if(A[i+2]!=1)
            i+=2;
        else
            i++;
        jumps++;
    }
    cout << "Traps: " << T << endl;
    cout << "Minimum jumps: " << jumps << endl;
    return 0;
}
```

Ερώτημα (ε)

```
#include<iostream>
using namespace std;
int main(){
    int n,i,m,E=100,jumps;
    cin>>n>>m;
    int A[n];
    for(i=0;i<n;i++){
        cin>>A[i];
    }
    i=0;
    do{
        i=(i+m)%n;
        if(A[i]==1)
            E-=3;
        else
            E-=1;
    }while(i!=0);
    cout<<"E="<<E<<endl;
    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.

Λύση

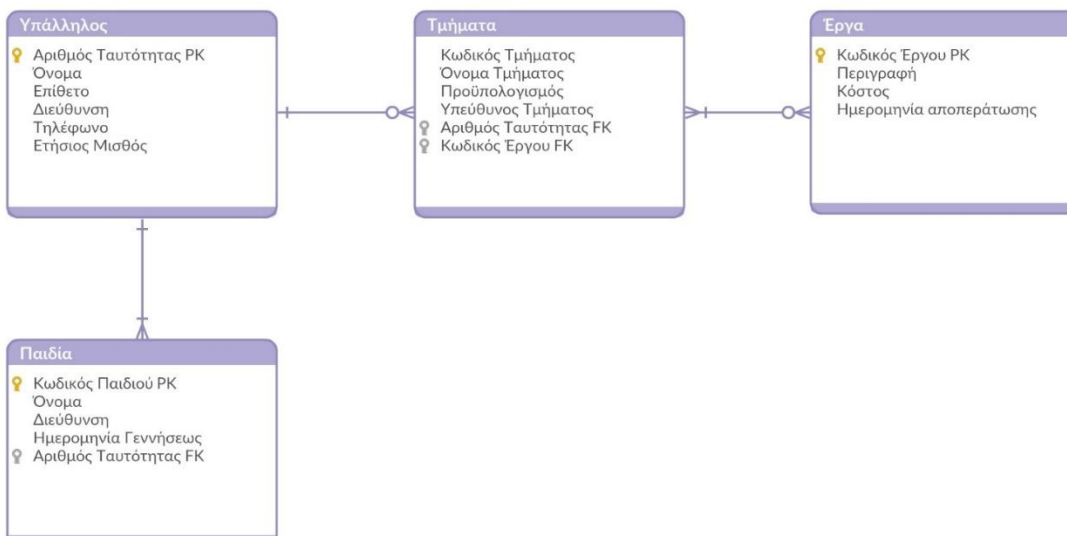
Ερώτημα (α)

Οντότητες: Διαχειριστής, Διευθυντής.

Διαδικασίες: Καταχώριση στοιχείων υπαλλήλου, αλλαγή στοιχείων υπαλλήλου, καταχώριση στοιχείων τμήματος, καταχώριση στοιχείων έργου, καταχώριση στοιχείων παιδιού, δημιουργία μηνιαίων καταλόγων / αναφορών.

Αρχεία: Αρχείο υπαλλήλων, αρχείο τμημάτων, αρχείο έργων, αρχείο παιδιών.

Ερώτημα (β)



Ερώτημα (γ)

The ER diagram shows two tables: Departments and Projects. Departments has a primary key Department_ID and foreign keys Emp_ID and Project_ID. Projects has a primary key Project_ID. A 1-to-many relationship exists between Departments and Projects, with a cardinality of 1:8.

Field:	Description	Date due	Cost	Budget
Table:	Projects	Projects	Projects	Departments
Sort:		Ascending		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Criteria:			> [Departments].[Budget]	
or:				

Ερώτημα (δ)

