

ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ, ΠΟΛΙΤΙΣΜΟΥ, ΑΘΛΗΤΙΣΜΟΥ ΚΑΙ ΝΕΟΛΑΙΑΣ  
ΔΙΕΥΘΥΝΣΗ ΑΝΩΤΕΡΗΣ ΕΚΠΑΙΔΕΥΣΗΣ  
ΥΠΗΡΕΣΙΑ ΕΞΕΤΑΣΕΩΝ

ΠΑΓΚΥΠΡΙΕΣ ΕΞΕΤΑΣΕΙΣ ΠΡΟΣΒΑΣΗΣ 2022

ΠΛΗΡΟΦΟΡΙΚΗ - ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΛΥΣΕΙΣ

ΜΕΡΟΣ Α'

ΑΣΚΗΣΗ 1:

(α)

ΑΡΧΗ

Διάβασε  
kostos

(β)

$epix \leftarrow kostos * 30 / 100$

$epix < 1500$

ψ

$tepix \leftarrow 1500$

$tepix \leftarrow epix$

(γ)

Τύπωσε  
tepix

ΤΕΛΟΣ

Σημείωση: Η συνθήκη  $epix < 1500$  μπορεί να είναι και  $epix \leq 1500$ .

## ΑΣΚΗΣΗ 2:

(α) Μετατροπή του δεκαδικού αριθμού **A=45** στο δυαδικό σύστημα.

	Αριθμός	Πηλίκο	Υπόλοιπο
Βήμα 1:	45/2	22	1
Βήμα 2:	22/2	11	0
Βήμα 3:	11/2	5	1
Βήμα 4:	5/2	2	1
Βήμα 5:	2/2	1	0
Βήμα 6:	1/2	0	1

$$(45)_{10} = (101101)_2$$

(β) Μετατροπή του δυαδικού αριθμού **B=(10101001)<sub>2</sub>** στο δεκαδικό σύστημα.

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
128	64	32	16	8	4	2	1
1	0	1	0	1	0	0	1

$$\begin{aligned} B = (10101001)_2 &= 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 128 + 32 + 8 + 1 \\ &= (169)_{10} \end{aligned}$$

(γ) Το συμπλήρωμα ως προς 2 του  $\Gamma = (00111001)_2$  είναι 11000111

1	0	1	0	1	0	0	1
1	1	0	0	0	1	1	1
<del>0</del>	1	1	1	0	0	0	0

Αγνοούμε την υπερχείλιση του τελευταίου ψηφίου

$B - \Gamma = (01110000)_2$

## ΑΣΚΗΣΗ 3:

(α) `cout << round(5%6/2.0) * sqrt(8+trunc(1.99)) + 6*4/3*2;`

$$\begin{aligned} &= \text{round}(5/2.0) * \text{sqrt}(8+1.0) + 24/3*2 \\ &= \text{round}(2.5) * \text{sqrt}(9.0) + 8*2 \\ &= 3.0 * 3.0 + 16 \\ &= 9.0 + 16 \\ &= 25.0 \end{aligned}$$

(β) i) `x=0 (FALSE)`

ii) `y=1 (TRUE)`

(γ) i) `(str[0]=='M' && str[str.size()-1]=='Σ')`

ii) `(num<0 || num%10>=4 && num%10<=7)`

#### **ΑΣΚΗΣΗ 4:**

- (α) **Τ.Π.Τ:** string name  
**Τ.Π.Α:** string na[],int va[],int &vathmos
- (β) **Θέση Α:** search(names,vathmoi,onoma,vath);

#### **ΑΣΚΗΣΗ 5:**

- (α) **Θέση Α:** int num[3][4];  
**Θέση Β:** fin>>num[i][j];
- (β) **Θέση Γ:** if (num[i][j]>=10 && num[i][j]<=99)
- (γ) **Θέση Δ:** for (j=0;j<4;j++)  
sum3row+=num[2][j];

#### **ΑΣΚΗΣΗ 6:**

- (α) struct mathima{  
string name;  
int code,pl;  
bool exe;  
};
- (β) mathima subject;  
ή  
struct mathima{  
string name;  
int code,pl;  
bool exe;  
}subject;
- (γ) (i) cin>>subject.code>>subject.pl;  
(ii) subject.name="Πληροφορική";  
(iii) if (subject.exe)  
cout<<subject.name;

**ΤΕΛΟΣ Α' ΜΕΡΟΥΣ  
ΑΚΟΛΟΥΘΕΙ ΤΟ ΜΕΡΟΣ Β'**

## ΜΕΡΟΣ Β'

### ΑΣΚΗΣΗ 7:

(α) Πίνακας αληθείας για την πιο κάτω λογική συνάρτηση:

$$G(A, B, C) = AC + ABC' + AB'C$$

A	B	C	G
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Άθροισμα ελαχιστόρων:  $G(A, B, C) = \Sigma(5, 6, 7)$

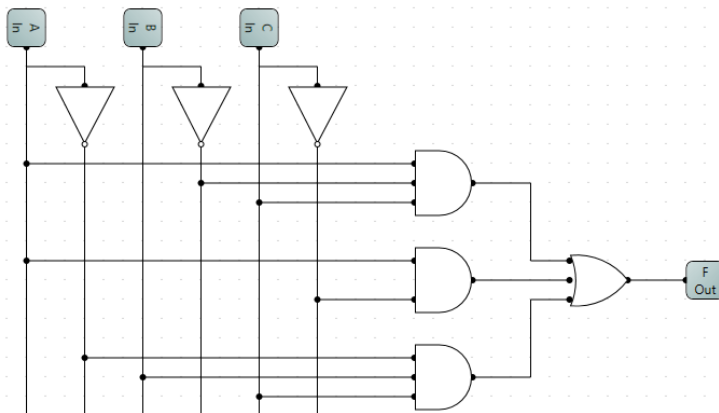
Γινόμενο Μεγιστόρων:  $G(A, B, C) = \Pi(0, 1, 2, 3, 4)$

(β) Λογική συνάρτηση που προκύπτει μετά την ομαδοποίηση των γειτονικών όρων του πιο κάτω χάρτη Karnaugh 3 μεταβλητών.

BC \ A	00	01	11	10
0	1	1	1	1
1	1	0	0	1

$$F(A, B, C) = A' + C'$$

(γ) Λογικό κύκλωμα για τη λογική συνάρτηση  $F(A, B, C) = AB'C + AC' + A'BC$



Τιμή της συνάρτησης F αν  $A=1$ ,  $B=1$  και  $C=0$ .

$$\begin{aligned} F(A, B, C) &= AB'C + AC' + A'BC \\ &= 1 \cdot 0 \cdot 0 + 1 \cdot 1 + 0 \cdot 1 \cdot 0 \\ &= 0 + 1 + 0 \\ &= 1 \text{ (TRUE)} \end{aligned}$$

## ΑΣΚΗΣΗ 8:

```
(α) #include<iostream>
using namespace std;
int main() {
    int i, sum=20, step=5, seires=1, tsum=20, N;
    for(i=2; i<=8; i++)
        sum+=step;
    cout<<"Σύνεδροι 8ης σειράς:"<<sum<<endl;
    cout<<"Δώσε πόσοι σύνεδροι:";
    cin>>N;
    sum=20;
    while (tsum<N) {
        sum+=step;
        tsum+=sum;
        seires++;
    }
    cout<<"Σειρές που χρειάζονται για να καθίσουν "<<N<<"
    σύνεδροι:"<<seires;
}
```

## (β) Προκαταρκτική εκτέλεση

### Κύρια συνάρτηση (main)

Μεταβλητές			Αποφάσεις		Παρουσίαση
a	b	c	b>0	A/Ψ	
3	4	3	4>0	A	x=7□y=5
	5		2>0	A	a=3□b=2□c=3
	2		0>0	Ψ	x=5□y=3
	3				a=3□b=0□c=3
	0				

### Συνάρτηση calc

Τ.Π.Τ	Τ.Π.Α	Αποφάσεις	
x	y	x>0	A/Ψ
3	4	3>0	A
7	5	3>0	A
3	2		
5	3		

## ΑΣΚΗΣΗ 9:

```
#include<iostream>
#include<fstream>

using namespace std;

#define row 5
#define col 5
```

```

int main(){
    int fishb[row][col],fisha[row][col],i,j,diff[row][col],maxdeft;
    ifstream fin1("before.txt");
    ifstream fin2("after.txt");
    ofstream fout("results.txt");

    for (i=0;i<row;i++)
        for (j=0;j<col;j++){
            fin1>>fishb[i][j];
            fin2>>fisha[i][j];
        }

    for (i=0;i<row;i++)
        for (j=0;j<col;j++){
            diff[i][j]=fisha[i][j]-fishb[i][j];
            if (diff[i][j]<=0)
                fout<<i<<" "<<j<<endl;
        }

    maxdeft=diff[0][4];
    for (i=0;i<row;i++)
        for (j=0;j<col;j++)
            if (i+j==4 && diff[i][j]>maxdeft)
                maxdeft=diff[i][j];

    cout<<"Συντεταγμένες περιοχών της δευτερεύουσας διαγωνίου με
τη μεγαλύτερη αύξηση:"<<endl;

    for (i=0;i<row;i++)
        for (j=0;j<col;j++)
            if (i+j==4 && diff[i][j]==maxdeft)
                cout<<i<<" "<<j<<endl;
    fin1.close();
    fin2.close();
    fout.close();
    return 0;
}

```

## **ΑΣΚΗΣΗ 10:**

### **Α' ΤΡΟΠΟΣ:**

```

#include<iostream>
#include<fstream>
#include <string>
using namespace std;

```

```

int main(){
    ifstream fin("text.txt");
    bool flag=true;
    string st,code;
    int pl=0;
    while (flag){
        fin>>st;
        if (st.size()>2 && st.size()%2==0){
            code=code+ st[st.size()/2-1] + st[st.size()/2];
            pl++;
        }
        if (pl==5) flag=false;
    }
    cout<<"Ο κωδικός είναι:"<<code;
    fin.close();
return 0;
}

```

### **Β' τρόπος:**

```

#include<iostream>
#include<fstream>
#include <string>
using namespace std;
int main(){
    string frasi,lexi,code;
    int i=0,middle,length=0;
    ifstream fin("text.txt");
    getline(fin,frasi);
    while(i<frasi.size()){
        lexi.clear();
        while(frasi[i]!=' ' && i<frasi.size()){
            lexi+=frasi[i];
            i++;
        }
        if (lexi.size()%2==0 && lexi.size()>2 && length<10){
            middle=lexi.size()/2;
            code=code+lexi[middle-1]+lexi[middle];
            length+=2;
        }
        i++;
    }
    cout<<"Ο κωδικός είναι:"<<code<<endl;
    fin.close();
return 0;
}

```

**ΤΕΛΟΣ Β' ΜΕΡΟΥΣ  
ΑΚΟΛΟΥΘΕΙ ΤΟ ΜΕΡΟΣ Γ'**

## ΜΕΡΟΣ Γ'

### **ΑΣΚΗΣΗ 11:**

```
#include<iostream>
using namespace std;
#define row 10
#define col 4

string maxtmima (string zooname[], int episk[][col]){
    int i,j,max=0,thesi,maxtmima;
    for (i=0;i<row;i++)
        if (zooname[i]=="ViennaZoo")
            thesi=i;
    for (j=0;j<col;j++)
        if (episk[thesi][j]>max){
            max=episk[thesi][j];
            maxtmima=j;
        }
    if (maxtmima==0)
        return ("Άγρια πτηνά");
    else if (maxtmima==1)
        return ("Θηλαστικά");
    else if (maxtmima==2)
        return ("Ερπετά");
    else
        return ("Έντομα");
}

int main(){
    string zooname[row],minzoo;
    int episk[row][col],total[row],sumrow=0,i,j,min;
    cout<<"Δώσε όνομα ζωολογικού κήπου και πλήθος επισκέψεων ανά
    τμήμα:"<<endl;
    for (i=0;i<row;i++){
        cin>>zooname[i];
        for (j=0;j<col;j++)
            cin>>episk[i][j];
    }
    for (i=0;i<row;i++){
        sumrow=0;
        for (j=0;j<col;j++)
            sumrow+=episk[i][j];
        total[i]=sumrow;
    }
}
```



```

cout<<"Συνολικός αριθμός επισκέψεων ανά ζωολογικό κήπο:"<<endl;
for (i=0;i<row;i++)
    cout<<total[i]<<" ";

min=episk[0][2];
minzoo=zooname[0];
for (i=0;i<row;i++)
    if (episk[i][2]<min){
        min=episk[i][2];
        minzoo=zooname[i];
    }
cout<<endl<<"Όνομα ζωολογικού κήπου με τις λιγότερες
επισκέψεις στο τμήμα των Ερπετών:"<<minzoo<<endl;

cout<<"Όνομα τμήματος με τη μεγαλύτερη επισκεψιμότητα του
ζωολογικού κήπου της Βιέννης:"<<maxtmima(zooname,episk)<<endl;

return 0;
}

```

## **ΑΣΚΗΣΗ 12:**

```

#include<iostream>
#include<fstream>
#include<iomanip>

using namespace std;

#define N 100

struct child{
    float age;
    char fylo;
    float ores;
    int gev;
    string ipef;
};

int main(){
    ifstream fin("askisi12IN.txt");
    child childSt[N],temp;
    int i,ploresA=0,maxg=0,plk=0;
    float sumoresA=0,mooresA;
    bool sorted;

```

```

for(i=0;i<N;i++)
fin>>childSt[i].age>>childSt[i].fylo>>childSt[i].ores>>childSt[i].gev
>>childSt[i].ipef;

    for(i=0;i<N;i++)
        if (childSt[i].fylo=='A'){
            sumoresA+=childSt[i].ores;
            ploresA++;
        }
    mooresA=sumoresA/ploresA;

    cout<<"Μέσος όρος των ωρών που κοιμούνται σ' ένα 24ωρο, τα
αγόρια:"<<fixed<<setprecision(2)<<mooresA<<endl;

    for(i=0;i<N;i++)
        if (childSt[i].ipef=="Τονείζ" && maxg<childSt[i].gev)
            maxg=childSt[i].gev;

    cout<<"Μέγιστος αριθμός γευμάτων των βρεφών που φροντίζουν οι
γονείς:"<<maxg<<endl;

    do{
        sorted=true;
        for (i=0;i<N-1;i++)
            if (childSt[i].ores<childSt[i+1].ores){
                temp=childSt[i];
                childSt[i]=childSt[i+1];
                childSt[i+1]=temp;
                sorted=false;
            }
    } while (sorted==false);

    cout<<"Ώρες και ηλικίες των 3 κοριτσιών που κοιμούνται
περισσότερο:"<<endl;

    for(i=0;i<N;i++)
        if (childSt[i].fylo=='K' && plk<3){
            cout<<fixed<<setprecision(1)<<childSt[i].ores<<"
"<<fixed<<setprecision(1)<<childSt[i].age<<endl;
            plk++;
        }
    fin.close();
    return 0;
}

```

**ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ**