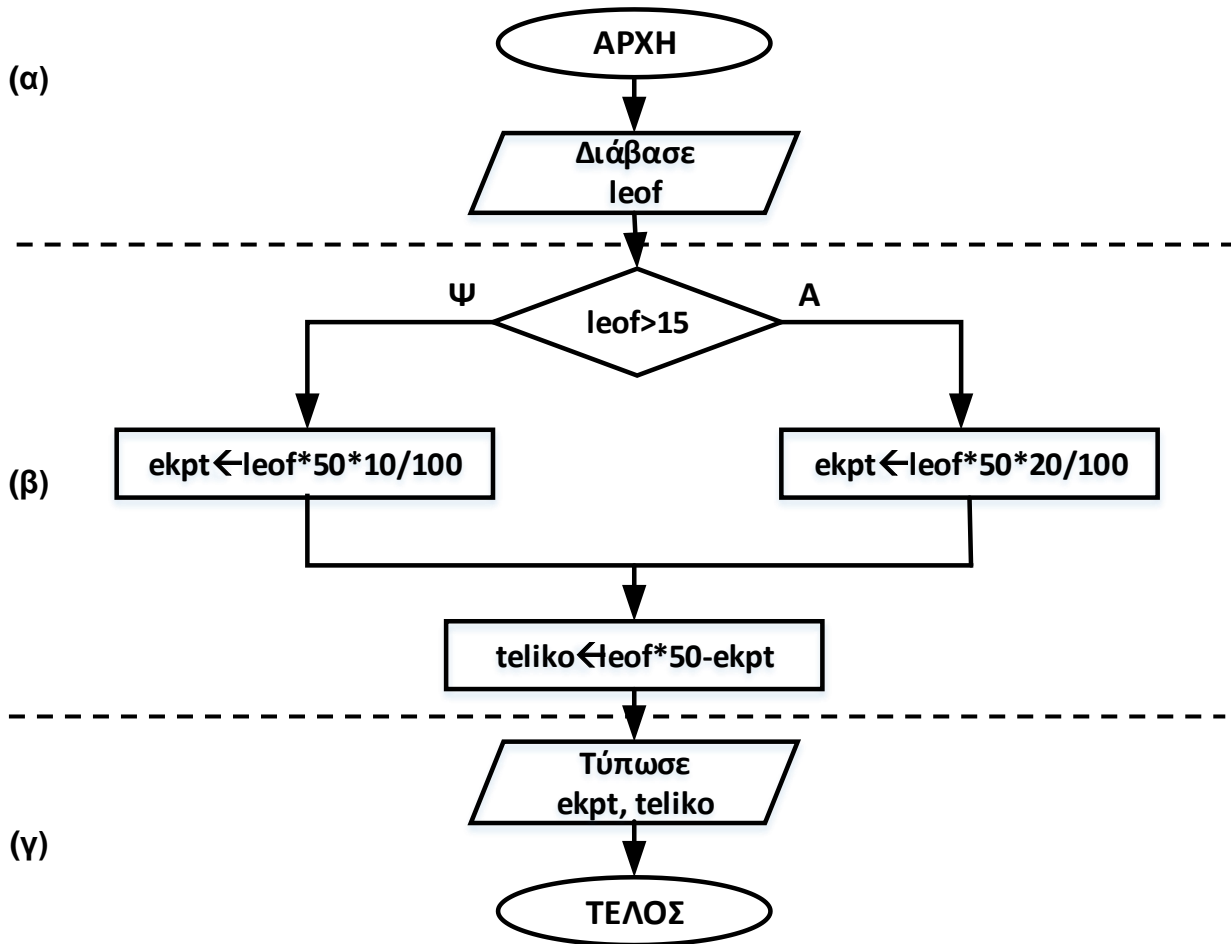


ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΠΟΛΙΤΙΣΜΟΥ  
ΥΠΗΡΕΣΙΑ ΕΞΕΤΑΣΕΩΝ

ΠΑΓΚΥΠΡΙΕΣ ΕΞΕΤΑΣΕΙΣ 2018  
ΠΛΗΡΟΦΟΡΙΚΗ - ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΛΥΣΕΙΣ

ΜΕΡΟΣ Α'

ΑΣΚΗΣΗ 1:



ΑΣΚΗΣΗ 2:

(α)  $A=(36)_{10} \rightarrow A=(100100)_2$

(β)  $B=(01010101)_2 \rightarrow B=(85)_{10}$

	Πηλίκο	Υπόλοιπο
Βήμα 1:	36/2	18
Βήμα 2:	18/2	9
Βήμα 3:	9/2	4
Βήμα 4:	4/2	2
Βήμα 5:	2/2	1
Βήμα 6:	1/2	0

$(36)_{10}=(100100)_2$

$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
128	64	32	16	8	4	2	1
0	1	0	1	0	1	0	1

$(01010101)_2=0*2^7+1*2^6+0*2^5+1*2^4+0*2^3+1*2^2+0*2^1+1*2^0$   
 $= 64+16+4+1=(85)_{10}$

(β) Το συμπλήρωμα ως προς 2 του  $\Gamma=(01010011)_2$  είναι **10101101**

### ΑΣΚΗΣΗ 3:

(α) `cout<<trunc(-6.5*pow(2,2))+abs(round(-sqrt(67)));`  
=trunc(-6.5\*pow(2,2))+abs(round(-sqrt(67)))  
=trunc(-6.5+4)+abs(round(-8.185))  
=trunc(-2.5)+abs(-8)  
=-2+8  
=6

(β) Αν  $a=-3$ ,  $b=2$  και  $c=4$ , τότε:

i)  $x = (b!=(3*\text{pow}(a,2)-1)) \ \&\& \ ((\text{abs}(a)+\text{trunc}(\text{float}b/c))\text{==}b);$   
 $x=0$  (FALSE)

ii)  $y = !((b+c)\text{>}=a) \ || \ (\text{sqrt}(b*8)\text{<}=\text{round}(a+7.3));$   
 $y=1$  (TRUE)

(γ)  $F(x,y,z) = (x+y)(x'+z)$  ή

$F(x,y,z) = (x+y)*(x'+z)$  ή

$F(x,y,z) = (x \text{ or } y) \text{ and } (\text{not}(x) \text{ or } z)$  ή

$F(x,y,z) = (x \ || \ y) \ \&\& \ (!x \ || \ z)$

Σημείωση: Ένα οποιοδήποτε από τα πιο πάνω θεωρείται σωστό.

### ΑΣΚΗΣΗ 4:

(α) Οι αντίστοιχες λογικές εκφράσεις για τις πιο κάτω λεκτικές εκφράσεις είναι:

i) Η μεταβλητή **grade** να είναι μεταξύ του **1** και **100 συμπεριλαμβανομένων**.

$(\text{grade}\text{>}=1 \ \&\& \ \text{grade}\text{<}=100)$  ή  $((\text{grade}\text{>}=1) \ \&\& \ (\text{grade}\text{<}=100))$

ii) Η μεταβλητή **z** να είναι ζυγός αριθμός και μην ισούται με **8**.

$(z\%2\text{==}0 \ \&\& \ z!\text{=}8)$  ή  $((z\%2\text{==}0) \ \&\& \ (z!\text{=}8))$

Σημείωση: Τόσο στο i) όσο και στο ii), ένα οποιοδήποτε από τα πιο πάνω θεωρείται σωστό.

(β) Το αντίστοιχο τμήμα προγράμματος με τη χρήση της περιπτωσιακής δομής **switch** είναι:

```
switch (flag) {
    case 'Z': cout<<"ΖΕΣΤΟ"<<endl;
              break;
    case 'X': cout<<"ΧΛΙΑΡΟ"<<endl;
              break;
    case 'K': cout<<"ΚΡΥΟ"<<endl;
              break;
    default : cout<<"Λανθασμένη τιμή"<<endl;
              break;
}
```

(γ) Τα τέσσερα (4) λάθη είναι τα πιο κάτω:

```
/*4*/ int i=0, count=0, num;           {λάθος 1}  
/*8*/ if (num%5==0) {                 {λάθος 2}  
/*13*/ while (i<10);                 {λάθος 3}  
/*14*/ cout<<count<<endl;           {λάθος 4}
```

#### ΑΣΚΗΣΗ 5:

(α) Αποτέλεσμα εντολής 8: **Εξετάσεις Πληροφορικής**  
Αποτέλεσμα εντολής 10: **Προγραμματισμός**

(β) Αποτέλεσμα εντολής 13: **22**  
Αποτέλεσμα εντολής 14: **15**

(γ) Εντολή θέσης A (εντολή 15): **cout<<st1[0]<<st1[1]<<st1[2];**

#### ΑΣΚΗΣΗ 6:

(α) **Θέση Α:** float ipotinousa(int a, int b)

(β) **Θέση Γ:** evperim (plevra1,plevra2,ipotin,evadon,perimeter);

(γ) **Θέση Β:**

```
string check (float evad, float perim){  
    string msg;  
    if (evad==perim)  
        msg="Είναί ίσα";  
    else  
        msg="Άνισα";  
    return msg;  
}
```

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

## ΜΕΡΟΣ Β'

### ΑΣΚΗΣΗ 7:

$$F(A,B,C) = A'B'C' + AB'C + ABC + A'BC'$$

(α) Πίνακας αληθείας της συνάρτησης F.

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

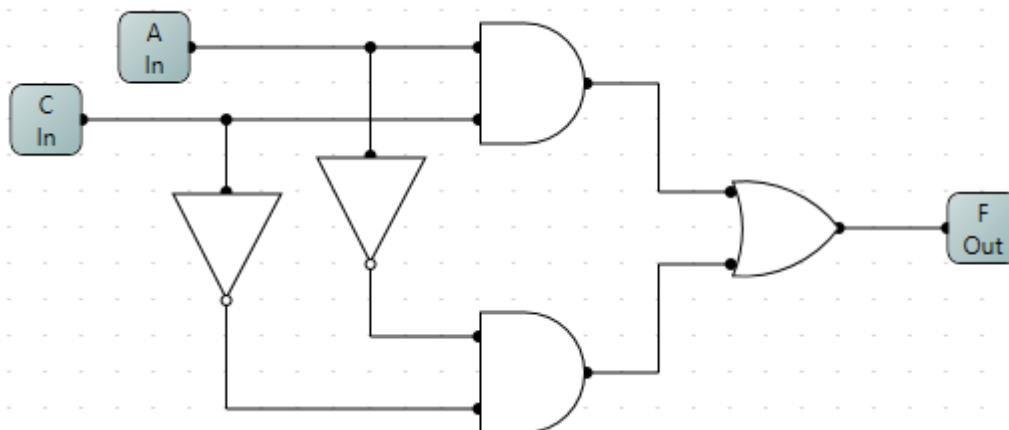
(β) Πίνακας Karnaugh που αντιστοιχεί στη συνάρτηση F.

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

(γ) Απλοποίηση F και αντίστοιχο λογικό κύκλωμα

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

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



## ΑΣΚΗΣΗ 8:

```
#include<iostream>
#include<iomanip>

using namespace std;

struct mathitis {
    string name,tmima;
    int apousia;
    bool aporos;
};

int main(){
    const int limit=25;
    mathitis s_class[limit];

    int i,sum=0,p=0,pB5=0;
    float mo;

    cout<<" Δώσε στοιχεία μαθητή: Όνομα, τμήμα, αριθμό απουσιών,
άπορος:" <<endl;
    for (i=0;i<limit;i++){
        cin>>s_class[i].name>>s_class[i].tmima>>
        s_class[i].apousia>>s_class[i].aporos;

        if (s_class[i].aporos==1) {
            sum+=s_class[i].apousia;
            p++;
        }
    }

    mo= (float) sum/p;
    cout<<"Μέσος όρος απουσιών άπορων μαθητών:" <<fixed <<
        setprecision(2)<<mo<<endl;

    for (i=0;i<limit;i++){
        if ((s_class[i].aporos==1) && (s_class[i].tmima=="B5")){
            cout<<s_class[i].name<<endl;
            pB5++;
        }
    }

    if (pB5==0)
        cout<<"Δεν υπάρχουν άποροι μαθητές στο B5";

    return 0;
}
```

### ΑΣΚΗΣΗ 9:

```
#include<iostream>
using namespace std;
void pzeros (int code[][4], int &sumdiag, int &prest){

int pall=0,i,j,pdiag1=0,pdiag2=0;

    for (i=0;i<4;i++)
        for (j=0;j<4;j++){
            if (code[i][j]==0)
                pall++;

            if ((i==j)&&(code[i][j]==0))
                pdiag1++;

            if ((i==4-j-1)&&(code[i][j]==0))
                pdiag2++;
        }
    sumdiag=pdiag1+pdiag2;
    prest=pall-sumdiag;
}

int main(){
    int code[4][4];
    int i,j,sdiag,rest;
    cout << "Δώσε 16 ακέραιους αριθμούς από 0 μέχρι 9:"<<endl;
    for (i=0;i<4;i++)
        for (j=0;j<4;j++)
            cin>>code[i][j];

    pzeros (code,sdiag,rest);

    if (sdiag==rest)
        cout<<"Ελεύθερη πρόσβαση"<<endl;
    else
        cout<<"Λαθος κωδικός"<<endl;
    return 0;
}
```

**ΑΣΚΗΣΗ 10:**

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

int main() {
    int n,sum=0,i,j,m;
    cout<<"Δώσε ένα ακέραιο αριθμό (N): ";
    cin>>n;
    cout<<"Αριθμός μυρμηγκιών:"<<n*n<<endl;
    cout<<"Αριθμός άδειων θαλάμων:"<<n*n-n<<endl;
    m=n;
    for (i=1;i<=n;i++) {
        for (j=1;j<=n-1;j++){
            sum=sum+m;
            m=m+1;
        }
        m=m-1;
    }
    cout<<"Συνολικός χρόνος:"<<sum;
    return 0;
}
```

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

## ΜΕΡΟΣ Γ'

### **ΑΣΚΗΣΗ 11:**

```
#include<iostream>
using namespace std;
#define row 20
#define col 5
int m10 (float tot[]){
    int i,p10=0;
    float mo;
    for (i=0;i<row;i++) {
        mo=tot[i] / col;
        if (mo<10)
            p10+=1;
    }
return p10;
}
int main(){
    string names[row],tempname;
    float prices[row][col],total[row];
    int i,j,k,count10;
    float sumrow,temp;

    cout<<"Δώσε το όνομα κάθε φυτωρίου και τις τιμές για το κάθε
ένα από τα 5 διαφορετικά είδη φυτών: "<<endl;
    for (i=0;i<row;i++){
        cin>>names[i];
        for (j=0;j<col;j++)
            cin>>prices[i][j];
    }

    for (i=0;i<row;i++) {
        sumrow=0;
        for (j=0;j<col;j++){
            sumrow+=prices[i][j];
        }
        total[i]=sumrow;
    }
    count10 = m10(total);
    cout<<"Πλήθος φυτωρίων με μέσο όρο τιμών <10:"<<count10<<endl;
```



```

    for (i=1;i<row;i++){
        temp=total[i];
        tempname=names[i];
        k=i-1;
        while (k>=0 && total[k]<temp) {
            total[k+1]=total[k];
            names[k+1]=names[k];
            k--;
        }
        total[k+1]=temp;
        names[k+1]=tempname;
    }
    cout<<"Τα 3 Φυτώρια με την χαμηλότερη συνολική τιμή..."<<endl;
    for(i=row-3;i<row;i++)
        cout<<names[i]<<endl;
return 0;
}

```

### **ΑΣΚΗΣΗ 12:**

```

#include<fstream>
#include<iostream>
using namespace std;
#define n 100
struct car{
    string marka,reg,color;
    int timi;
};
int main(){
    ifstream fin1("askisi12IN.txt");
    ifstream fin2("afxisi.txt");
    ofstream fout("askisi12OUT.txt");
    int i,incr,p,maxtimi;
    string ar_eg,maxcolor;
    car carSt[n],temp;

    // διάβασμα στοιχείων αυτοκινήτων από το αρχείο askisi12IN.txt
    for(i=0;i<n;i++){
fin1>>carSt[i].marka>>carSt[i].reg>>carSt[i].color>>carSt[i].timi;
    }
}

```

```

// προσθήκη αύξησης από το αρχείο afxisi.txt
while(!fin2.eof()){
    fin2>>ar_eg>>incr;
    for(i=0;i<n;i++)
        if(carSt[i].reg==ar_eg){
            carSt[i].timi+=incr;
        }
    }
//καταχώρηση των στοιχείων στο αρχείο askisi12OUT και των 100
αυτοκινήτων μετά την προσθήκη της αύξησης.
for(i=0;i<n;i++)
    fout<<carSt[i].marka<<" "<<carSt[i].reg<<" "<<carSt[i].color<<"
"<<carSt[i].timi<<endl;
// υπολογισμός και παρουσίαση της τιμής και του χρώματος του
ακριβότερου αυτοκινήτου
maxtimi=0;
for(i=0;i<n;i++)
    if (carSt[i].timi>maxtimi) {
        maxtimi=carSt[i].timi;
        maxcolor=carSt[i].color;
    }
    cout<<"Τιμή και χρώμα ακριβότερου αυτοκινήτου:" <<maxtimi<<"
"<<maxcolor<<" "<<endl;
// εύρεση και παρουσίαση της μάρκας του αυτοκινήτου βάσει του
αριθμού εγγραφής του (σειριακή αναζήτηση)
cout<<"Δώσε αριθμό εγγραφής αυτοκινήτου:";
cin>>ar_eg;
p=0;
for (i=0; i<n; i++)
    if (carSt[i].reg==ar_eg){
        p+=1;
        cout<<carSt[i].marka;
    }
if (p==0)
    cout<<"Δεν υπάρχει τέτοιο αυτοκίνητο!";
fin1.close();
fin2.close();
fout.close();
return 0;
}

```

**ΤΕΛΟΣ ΕΞΕΤΑΣΤΙΚΟΥ ΔΟΚΙΜΙΟΥ**