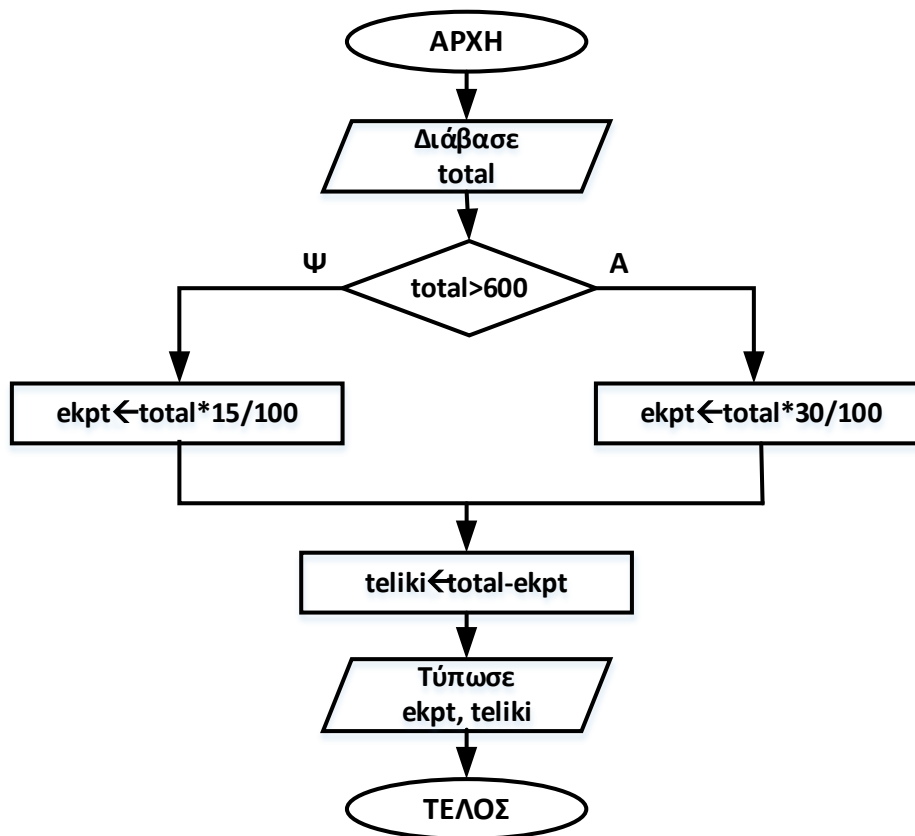


Ενδεικτικές Λύσεις  
Δοκιμίου Γ Κατεύθυνσης 2017-2018

Άσκηση 1

(α)



(βαθμοί 3)

(β)

```
if (total > 600)
```

(βαθμός 1)

(γ)

```
cout << teliki;
```

(βαθμός 1)

## Άσκηση 2

(α)

170

(Βαθμός 1)

(β)

A=01010110, B=10111111

(Βαθμοί 2)

(γ)

11110001

(Βαθμοί 2)

## Άσκηση 3

*(εδώ απαιτούνται και τα βήματα)*

(α) `round(-5.3) + 3*pow(2,3) + abs(-12) - trunc(6.2 + (5%2)) + sqrt(16)`  
`=-5+3*23+12-trunc(6.2+1)+4.0`  
`=-5+3*8+12-7+4.0`  
`=-5+24+9`  
`=-5+33`  
`=28`

(Βαθμοί 2)

*(εδώ απαιτείται μόνο το τελικό αποτέλεσμα )*

(β)

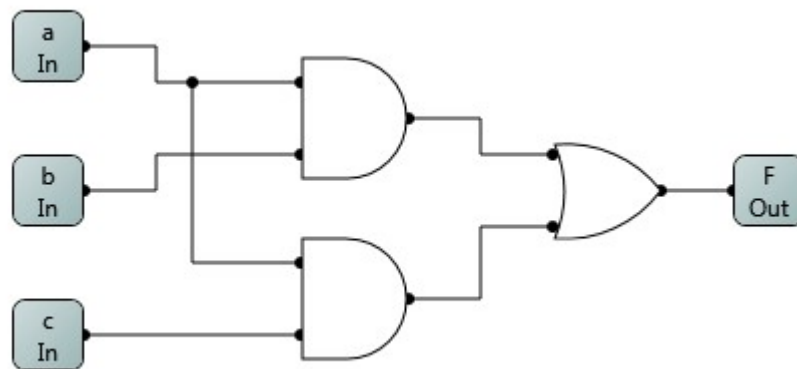
i) `x=(a==(2*a-1)) || ((abs(c)+2)!=a/b) && (a>=(a+ pow(c,2)))`  
`(2==(2*2-1)) || ((abs(-3)+2)!=2/1) && (2>=(2+pow(-3,2)))`  
`(2==3) || ((3+2)!=2) && (2>=(2+9))`  
`(2==3) || ((5!=2) && (2>=11))`  
`F || (T && F)`  
`F || F`  
`0 (FALSE)`

ii) `y=((a+b)<=2*c) && !((a-b)>=4)`  
`((2+1)<=2*(-3)) && !((2-1)>=4)`  
`(3<=-6) && !(1>=4)`  
`F && ! F`  
`F && T`  
`0 (FALSE)`

(Βαθμοί 2)

(γ)

$$F(a, b, c) = (a \ \&\& \ b) \ || \ (a \ \&\& \ c)$$



(βαθμός 1)

#### Άσκηση 4

(α)

$$(A==90 \ || \ B==90 \ || \ C==90 \ \&\& \ (A+B+C)==180)$$

(βαθμός 1)

(β)

```
i=10;
while(i>=0){
    cout << i << endl;
    i-=2;
}
```

(βαθμοί 2)

(γ)

```
/*2*/      #define N 10
/*5*/      int i, x, p=0;           (2 λάθη στην ίδια γραμμή)
/*9*/      if (x%2!=0)
```

(βαθμοί 2)

#### Άσκηση 5

(α)

```
ifstream fin ("askisi5IN.txt");
ofstream fout ("askisi5OUT.txt");
```

(βαθμοί 2)

(β)

```
while(!fin.eof())
```

(βαθμοί 2)

(γ)

```
sum += N;
```

(βαθμός 1)

## Άσκηση 6

(α)

```
int triple(int A);
```

(Βαθμός 1)

(β)

```
void print(int A, int B);
```

(Βαθμός 1)

(γ)

```
float calculate(int A, int B, int &D){  
    D = A%10 + B%10;  
    return (A+B)/2.0;  
}
```

(Βαθμοί 3)

**ΤΕΛΟΣ ΜΕΡΟΥΣ Α'**

## ΜΕΡΟΣ Β'

### Άσκηση 7

(α) Πίνακας αλήθειας

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

(βαθμοί 3)

(β) Πίνακας Karnaugh

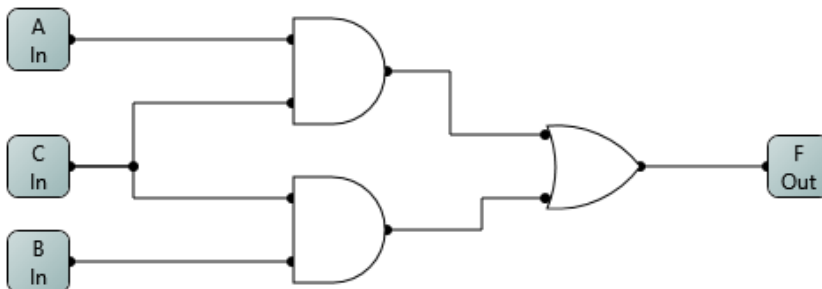
A	BC			
	00	01	11	10
0	0	0	1	0
1	0	1	1	0

(βαθμοί 3)

(γ) Απλοποίηση

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

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



(βαθμοί 4)

## Άσκηση 8

```
#include<iostream>
#define N 10
using namespace std;
int main(){
    int A[N],i,max,min,s,j,temp;

    cout<<"Δώσε 10 ακέραιους αριθμούς"<<endl;
    for(i=0;i<N;i++)
        cin>>A[i];
    max=A[0];
    min=A[0];
    for(i=1;i<N;i++){
        if(A[i]>max)
            max=A[i];
        if(A[i]<min)
            min=A[i];
    }

    cout<<" Μεγαλύτερος="<<max<<endl<<" Μικρότερος="<<min<<endl;
    cout<<"Δώσε αριθμό κυκλικών μετακινήσεων"<<endl;
    cin>>s;
    for(j=0;j<s;j++){
        temp=A[N-1];
        for(i=N-1;i>0;i--)
            A[i]=A[i-1];
        A[0]=temp;
    }

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

## Άσκηση 9

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

int main(){
    string stA,stB;
    int N,i,p=0,A[26]={},m;
    char cMax;
    cout<<"Δώσε συμβολοσειρές A, B και τον αριθμό N"<<endl;
    cin>>stA>>stB>>N;
    if(stA.size(>N){
```

```

        cout<<"Εμφάνιση συμβολοσειράς A μέχρι τον N-στο
        χαρακτήρα"<<endl;
        for(i=0;i<N;i++)
            cout<<stA[i];
        cout<<endl;
    }
    else {
        cout<<"Εμφάνιση ολόκληρης της συμβολοσειράς A "<<endl;
        cout<<stA<<endl;
    }
    for(i=0;i<stB.size();i++)
        if(stA[0]==stB[i])
            p++;
    cout<<"Πλήθος εμφανίσεων του 1ου χαρακτήρα της A μέσα στη
    B:"<<p<<endl;
    for(i=0;i<stB.size();i++)
        A[stB[i]-'A']++;
    m=A[0];
    cMax='A';
    for(i=1;i<N;i++)
        if(A[i]>m){
            m=A[i];
            cMax=i+'A';
        }
    cout<<"Χαρακτήρας με τις περισσότερες εμφανίσεις μέσα στη
    B:"<<cMax<<endl;
    return 0;
}

```

## Άσκηση 10

```

#include<iostream>
#define N 6
using namespace std;

int Max2(int A[][N]){
    int i,j,m;
    m=A[0][0];
    for(i=0;i<N;i++)
        for(j=0;j<N;j++)
            if(A[i][j]>m)
                m=A[i][j];
    return m;
}

int main(){

```

```

int arr1[N],arr2[N][N],i,j,ans;
bool check=false;
cout<<"Δώσε τα στοιχεία του δισδιάστατου πίνακα"<<endl;
for(i=0;i<N;i++)
    for(j=0;j<N;j++)
        cin>>arr2[i][j];
cout<<"Δώσε τα στοιχεία του μονοδιάστατου πίνακα"<<endl;
for(i=0;i<N;i++)
    cin>>arr1[i];

cout<<"Μεγιστός αριθμός δισδιάστατου πίνακα:"<<Max2(arr2)<<endl;

for(i=0;i<N;i++){
    ans=1;
    for(j=0;j<N;j++)
        if(arr2[j][i]!=arr1[j]){
            ans=0;
        }
    if(ans==1){
        check=true;
    }
}
if(check==true)
    cout<<1;
else
    cout<<0;
return 0;
}

```

**ΤΕΛΟΣ Β' ΜΕΡΟΥΣ**



## ΜΕΡΟΣ Γ'

### Άσκηση 11

```
#include<iostream>
#include<iomanip>
#include<string>
#define N 1000
#define M 12
using namespace std;

void TotalProfits(float pr[][M],float t[]){
    int i,j;
    for(i=0;i<N;i++){
        t[i]=0;
        for(j=0;j<M;j++)
            t[i]+=pr[i][j];
    }
}

int main(){
    string shops[N],key,temp1;
    float profits[N][M],totals[N],temp2;
    bool sorted, found=false;
    int i,j,x,first = 0, last = N - 1, mid;
    cout<<"Δώσε όνομα καταστήματος και έσοδα μήνα"<<endl;
    for (i=0;i<N;i++){
        cin>>shops[i];
        for(j=0;j<M;j++)
            cin>>profits[i][j];
    }

    TotalProfits(profits,totals);

    do{
        sorted = true;
        for (i=0; i<N-1; i++){
            if (shops[i]>shops[i+1]){
                temp1 =shops[i];
                shops[i] = shops[i+1];
                shops[i+1] = temp1;
                temp2 =totals[i];
                totals[i] = totals[i+1];
                totals[i+1] = temp2;
                sorted = false;
            }
        }
    } while (sorted==false);

    cout<<"Καταστήματα - Κέρδη"<<endl;
    for(i=0;i<N;i++)
```

```

cout<<shops[i]<<" "<<fixed<<setprecision(2)<< totals[i]<<endl;
cout<<"Δώσε όνομα καταστήματος προς αναζήτηση"<<endl;
cin>>key;

while (first <= last && !found) {
    mid = (first + last) / 2;
    if (shops[mid] == key)
        found=true;
    else if (shops[mid] > key)
        last = mid - 1;
    else
        first = mid + 1;
}
if(found==true)
    cout<<"Κέρδη:"<<fixed<<setprecision(2)<<totals[mid];
else
    cout<<-1;
return 0;
}

```

## Άσκηση 12

```

#include<fstream>
#include<iostream>
#include<string>
#include<iomanip>
using namespace std;
#define N 12

struct student{
    int AM;
    float eis;
};

float total(student s[], int n){
    float sum=0;
    int i;
    for(i=0;i<n;i++)
        sum+=s[i].eis;
return sum;
}

int main(){
    ifstream fin1("askisil2IN.txt");
    ifstream fin2("eisfores.txt");
    ofstream fout("askisil2OUT.txt");
    int i,am,j;
    float e;
    student studentsC1[N],temp;

```

```

for(i=0;i<N;i++){
    fin1>>studentsC1[i].AM>>studentsC1[i].eis;
}

while(!fin2.eof()){
    fin2>>am>>e;
    for(i=0;i<N;i++)
        if(studentsC1[i].AM==am){
            studentsC1[i].eis+=e;
        }
}
cout<<"Συνολική εισφορά:"
<<fixed<<setprecision(2)<<total(studentsC1,N)<<endl;

for (i=1; i<N; i++){

    temp = studentsC1[i];
    j = i - 1;

    while (j>=0 && studentsC1[j].eis>temp.eis){
        studentsC1[j+1] = studentsC1[j];
        j--;
    }
    studentsC1[j+1] = temp;
}

for(i=0;i<N;i++)
    fout<<studentsC1[i].AM<<" "
<<fixed<<setprecision(2)<<studentsC1[i].eis<<endl;
return 0;
}

```