

LÓVERSENYZÉS

Feladat:

Egymást követő hétvégeken feljegyeztük, hogy a lóversenyen mennyit nyertünk illetve veszítettünk (ez utóbbi egy negatív szám). Melyik időszakban (két hétvége között) volt az összesített nyereségünk (két hétvége között nyert illetve veszített pénzek összege) a legnagyobb?

Specifikáció:

$V = \text{Vect}(N, Z);$

$A = \underset{v}{V} \times \underset{i}{N} \times \underset{\max}{Z} \quad | \quad \underset{k}{N}$

$B = \underset{v'}{V}$

$Q = (v=v')$

$R = (Q \wedge i \in [v.lob..v.hib] \wedge \max = f(v, i) \wedge \forall j \in [v.lob..v.hib] : f(v, j) \leq f(v, i))$

$f : V \times N \rightarrow Z;$

$f(x, j) = \sum_{j=x.lob}^i x_j$

Visszavezetés: Maximumkeresés

Tétel	Feladat
m	$v.lob$
n	$v.hib$

Szigorítás	Nincs.
Plussz feltétel	Nincs.
Plussz paraméter	v vektor.

Megoldás:

$k, i, \max := v.lob, v.lob, f(v, v.lob)$	
$k \neq v.hib$	
$f(v, k+1) \geq \max$	$f(v, k+1) \leq \max$
$i, \max :=$ $k+1, f(v, k+1)$	SKIP
$k := k+1$	

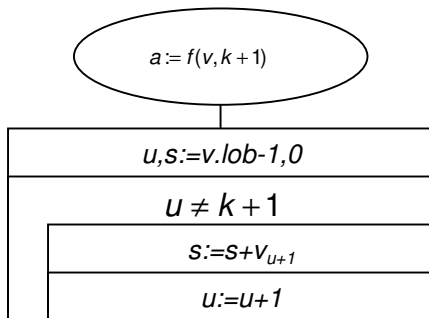
Program-transzformáció: Függvény helyettesítése változóval:

$a := f(v, k+1)$

Visszavezetés: Összegzés

Tétel	Feladat
m	$v.lob$
n	$k+1$
$f(j)$	$v.j$

Szigorítás	Nincs.
Plussz feltétel	Nincs.
Plussz paraméter	v vektor.

**Tétel:**

A stuktogramm formában megadott program megoldása a feladatnak.

Bizonyítás:

A maximumkeresés és az összegzés tételből következik.

C++ Forráskód:

```

#include <iostream>
#include <string>
#include <fstream>

using namespace std;

//Fuggvénydeklaráció
int f(int* v, int j);
//Eljárásdefiníció
void g(int* v, int j, int &eredm);

//Main függvény
int main()
{
    //Alapváltozók, deklarációja
    string ragain;
    string choice;
    string fname;
    const int maxn=100;
    int* v;
    int n,i,max;
    //INTERFACE
    do
    {
        cout << "LOVERSENY-KONYVELES" << endl;
        cout << "Adatok beolvasása" << endl;
        do
        {
            cout << "Fajlból [F]" << endl;
            cout << "Billentyűről [B]" << endl;
            cin >> choice;
        }
        while((choice!="F") && (choice!="f") && (choice!="B") && (choice!="b"));
        //Betöltés fajlból...
        if((choice=="F") || (choice=="f"))
        {
            cout << "Fajlnev: ";
            cin >> fname;
            ifstream inp(fname.c_str());
            if(inp.fail())
            {
                cout << "A megadott fajlt nem találok!" << endl;
                exit(1);
            }
            inp >> n;
        }
    }
}

```

```
        cout << "Hetek szama: " << n << endl;
        if(n<1)
        {
            cout << "Hibas adat!" << endl;
            exit(1);
        }
        v = new int[n];
        for(int j=0; j<n; ++j)
        {
            inp >> v[j];
            cout << (j+1) << ". het eredmenye: " << v[j] << endl;
        }
    }
    else
    {
        //Beolvasas billentyurol...
        cout << "Konyvelendo hetek szama: "; cin >> n;
        if(n<1)
        {
            cout << "HIBA: Legalabb 1 napnak lennie kell!" << endl;
            exit(1);
        }
        if(n>maxn)
        {
            cout << "HIBA: Maximum " << maxn << " db het kezelheto!" << endl;
            exit(1);
        }
        v = new int[n];
        for(int j=0; j<n; ++j)
        {
            cout << (j+1) << ". het eredmenye: "; cin >> v[j];
        }
    }
    //MAXIMUMKERESES
    i=0;
    max=f(v,0);
    for(int j=0; j<n; ++j)
    {
        if(f(v,j)>max)
        {
            i=j;
            max=f(v,j);
        }
    }
    //KIIRATAS, TAKARITAS
    cout << "Legjobb anyagi helyzetunk: " << (i+1) << ". het;" << endl;
    cout << "Ekkor osszesen " << max << " Forintunk volt." << endl;
    delete[] v;
    cout << "Futtassam ujra (I/N)? ";
    cin >> ragain;
}
while((ragain!="N")&&(ragain!="n"));
return 0;
}

//Fuggvenydefinialas
int f(int* v, int j)
{
    int eredm=0;
    for(int l=0; l<=j; ++l)
    {
        eredm=eredm+v[l];
    }
    return eredm;
}

//Eljarasdefinialas
void g(int* v, int j, int &eredm)
{
    eredm=0;
    for(int l=0; l<=j; ++l)
    {
        eredm=eredm+v[l];
    }
}
```

Tesztelési terv:

Tesztesek a feladat alapján (fekete doboz tesztelés):

- Fájlból olvasás kipróbálása (létező és nem létező fájlal is);
- Billentyűről olvasás kipróbálása;
- Vizsgálandó hetek megadása (1-nél kisebb és nem kisebb adatokkal is);
- Könyvelési adatok kitöltése;
- Újrafuttatás kipróbálása;

Tesztesek a kód alapján (fehér doboz tesztelés):

- Fájlból olvasás kipróbálása (létező és nem létező fájlal is);
- Billentyűről olvasás kipróbálása;
- Vizsgálandó hetek megadása (1-nél kisebb és nem kisebb egészekkel is);
- Tömb feltöltése egészekkel;
- Újrafuttatás kipróbálása;
- Kód átírása függvény helyett eljárás használatára, majd ezt követően az eddigi lépések megismétlése;