

Temă pentru acasă. Expresii

Completați definițiile următoarelor funcții care prelucrează tablouri unidimensionale (vectori) de numere întregi, corespunzător cerințelor precizate. Tablourile au `dim` elemente.

- 1.) `int primulMax(int tab[], int dim){...}`  
returnează cel mai mic indice al elementelor de valoare maximă din tabloul `tab`;
- 2.) `int ultimulMin(int tab[], int dim){...}`  
returnează cel mai mare indice al elementelor de valoare minimă din tabloul `tab`;
- 3.) `int diferentaValorilorExtreme(int tab[], int dim){...}`  
returnează diferența dintre cea mai mare și cea mai mică valoare a elementelor vectorului `tab`;
- 4.) `int deltaMin(int tab[], int dim){...}`  
returnează cea mai mică diferență în valoare absolută dintre elementele distincte ale vectorului `tab`;
- 5.) `int produsulSumelor(int tab[], int dim){...}`  
returnează produsul dintre suma elementelor pozitive și suma elementelor negative ale vectorului `tab` (dacă nu există elemente pozitive suma lor este 0, analog pentru negative);
- 6.) `int sumaProduselor(int tab[], int dim){...}`  
returnează suma dintre produsul elementelor impare ale vectorului `tab` și produsul elementelor pare (dacă nu există elemente impare produsul este 1, analog pentru pare);
- 7.) `int lgMaxSecventaPara(int tab[], int dim){...}`  
returnează lungimea celei mai lungi subsecvențe de elemente consecutive din vectorul `tab` formată numai din numere pare;
- 8.) `int lgMaxSecventaCresc(int tab[], int dim){...}`  
returnează lungimea celei mai lungi subsecvențe strict crescătoare de elemente consecutive din vectorul `tab`;
- 9.) `void afisareValori(int tab[], int dim){...}`  
afișează mulțimea valorilor elementelor vectorului `tab` (fiecare valoare este afișată numai o singură dată);
- 10.) `void afisareValoriComune(int a[], int b[], int dim){...}`  
afișează valorile comune tablourilor `a` și `b` (fiecare valoare comună este afișată numai o singură dată);

- 11.) `void afisareValoriExclusive(int a[], int b[], int dim){...}`  
 afișează valorile care se găsesc numai în tabloul a sau numai în b (fiecare valoare afișată este afișată numai o singură dată);
- 12.) `int sumaMaximaPeSecvente(int a[], int dim){...}`  
 returnează valoarea maximă a sumelor subsecvențelor de elemente consecutive;

Exemplu de rezolvare:

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

int lgMaxSecventaPara1(int tab[], int dim){
    int lgMax=0, lg=0;
    for(int i=0; i<dim; i++){
        if(tab[i]%2==0)
            lg++;
        else //s-a terminat o subsecventa para
            lg=0;
        if(lgMax<lg) lgMax=lg;
    }
    return lgMax;
}

int lgMaxSecventaPara2(int tab[], int dim){
    int lgMax=0, lg=0;
    for(int i=0; i<dim; i++){
        if(tab[i]%2) lg=0;
        else if(lgMax<++lg) lgMax=lg;
    }
    return lgMax;
}

int lgMaxSecventaPara3(int tab[], int dim){
    int lgMax=0, lg=0;
    for(int i=0; i<dim; i++){
        (tab[i]%2 ? lg=0 : ++lg) && (lgMax<lg) && (lgMax=lg);
    }
    return lgMax;
}

int main(){
    int vect[10]={1, 2, 30, 4, 50, 6, 7, 8, 90, 10};
    cout<<lgMaxSecventaPara1(vect, 10)<<endl;
    cout<<lgMaxSecventaPara2(vect, 10)<<endl;
    cout<<lgMaxSecventaPara3(vect, 10)<<endl;
    return 0;
}
```