

Rezolvare + Indicatii Test nr 1 - Carte Vlad

Subiectul I

1. c
2. b
3. c
4. c
5. 8

Subiectul II

1. Raspuns:

- a. Afiseaza cifra de control adica suma cifrelor fiecarui raspuns intermediar => 25 => 7
- b. Cel mai mic 10004, cel mai mare 99995
- c.

```
#include <iostream>
using namespace std;
int main() {
    int n;
    cin >> n;
    while (n > 9) {
        int s = 0;
        while (n != 0) {
            s = s + n % 10;
            n = n/10;
        }
        n = s;
    }
    cout << n;
    return 0;
}
```

d.

```
citeste n
(numar natural nenul de cel mult 9 cifre)
cat timp n > 9
    s <- 0
    cat timp n <> 0
        daca (n > 0) atunci
            executa
                s <- s + n %10
                n <- [n/10]
        cat timp(n > 0)
```

```
n = s  
scrie s
```

2. Se va genera matrice si se va observa ca `a[3][3] = 16` si suma elementelor de pe prima linie este 4

3. Rezolvare:

```
float med = (x.nota1 + x.nota2) / 2.0;
```

- NOTA: este posibil ca in carte autorul sa se refere la variabila `e` pe care a declarat-o si sa fie gresit enuntul. In schimb, deoarece media este o valoare reala, pentru a nu pierde partea de dupa virgula, trebuie sa ne asiguram ca impartim la 2.0

Subiectul III

1.

- De retinut:
 - Divizorii proprii ai unui numar sunt toti divizorii mai putin 1 si numarul insusi.
 - Solutia va combina algoritmul pentru determinarea divizorilor unui numar cu algoritmul pentru a determina daca un numar este prim sau nu.
- Solutie:

```
#include <iostream>  
  
using namespace std;  
int calcul(int n);  
  
int main() {  
    cout << calcul(15);  
    return 0;  
}  
  
int calcul(int n) {  
    int suma = 0;  
    for (int i = 2; i <= n/2; i++) {  
        if (n % i == 0) {  
            int estePrim = 1;  
            for (int j = 2; j <= i/2; j++) {  
                if (i % j == 0) {  
                    estePrim = 0;  
                    break;  
                }  
            }  
            if (estePrim) {  
                suma += i;  
            }  
        }  
    }  
}
```

```

    }
}

return suma;
}

```

2.
 - Indicatii:
 - Chiar daca se poate afisa direct raspunsul, pentru a evita penalizari, se va construi o alta variabila ce va tine rezultatul
 - Solutie:

```

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

int main() {

    char text[31], vocale[31];
    int j = 0;
    cin >> text;
    for (int i = 0; i < strlen(text); i++) {
        if (strchr("aeiou", text[i]) != NULL) {
            vocale[j++] = text[i];
        }
    }
    cout << vocale;
    return 0;
}

```

3.
 - Explicatie eficienta:
 - Programul este eficient din punct de vedere al memoriei deoarece utilizam doar doua variabile pentru a citii si a analizat numere din fisier. Un alt atu al eficientei programului este faptul ca citim o singura data fisierul.
 - De retinut:
 - Atentie la ultimul element citit!!
 - Solutie:

```

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

int main() {
    ifstream fin ("bac.in");
    ofstream fout ("bac.out");

    long int numarAnterior, numarCurent;

```

```
int aparitii = 1;
fin >> numarAnterior;
while (fin >> numarCurent) {
    if (numarCurent == numarAnterior) {
        aparitii++;
    } else {
        fout << numarAnterior << " " << aparitii << endl;
        aparitii = 1;
    }
    numarAnterior = numarCurent;
}
fout << numarAnterior << " " << aparitii << endl;
fin.close();
fout.close();
return 0;
}
```