

Lucrarea conține 8 pagini scrise

PROBĂ SCRISĂ

Pentru*) EXAMENUL NAȚIONAL
DE BACALAUREAT

Disciplina INFORMATICA
LIMBAJUL C/C++

Sesiunea Iunie - Iulie 2023
(luna, anul)

*) Se completează după caz: examen de bacalaureat, evaluare națională, admitere, certificare a calificării profesionale, concurs sau olimpiadă școlară

Numele: PODESCU
Inițiala prenumelui tatălui: C.A.
Prenumele: MĂDĂLIN COSTEL
Școala de proveniență: COL. NAȚ. DE
INFO. "TUDOR VIANU"
Centrul de examen: COL. NAȚ. DE
INFO. "TUDOR VIANU"
Loc. BUCUREȘTI jud. SECTOR 1

Nume și prenume asistent	Semnătura
<u>GABRIEL LORENA</u>	<u>[semnătură]</u>
<u>IONESCU TOMA</u>	<u>[semnătură]</u>

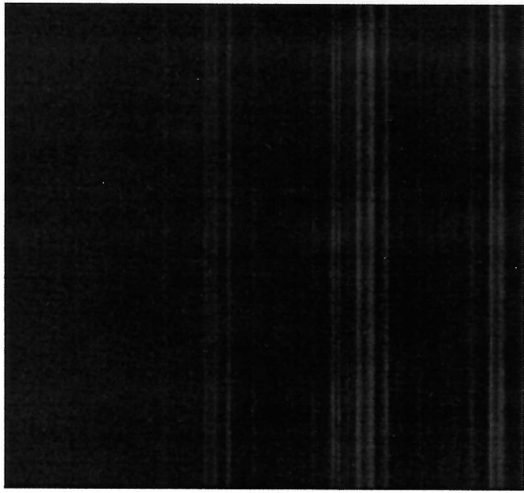
A	COMISIA DE EVALUARE	REZULTATUL EVALUĂRII**	NUMELE ȘI PRENUMELE PROFESORULUI	SEMNĂTURA
	EVALUATOR I			
	EVALUATOR II			
	EVALUATOR III			
	EVALUATOR IV			
	NOTA FINALĂ/PUNCTAJ ȘI/SAU NIVEL DE COMPETENȚĂ			

B	COMISIA DE CONTESTAȚII	NOTĂ (CIFRE ȘI LITERE)	NUMELE ȘI PRENUMELE PROFESORULUI	SEMNĂTURA
	EVALUATOR I			
	EVALUATOR II			
	EVALUATOR III			
	EVALUATOR IV			
	NOTA FINALĂ			

C	COMISIA DE REEVALUARE LA CONTESTAȚII	NOTĂ (CIFRE ȘI LITERE)	NUMELE ȘI PRENUMELE PROFESORULUI	SEMNĂTURA
	EVALUATOR I			
	EVALUATOR II			
	EVALUATOR III			
	EVALUATOR IV			
	NOTA FINALĂ			

**) Se va completa, după caz, cu:

- Nota (cifre și litere) - pentru probele scrise din cadrul examenului de bacalaureat, evaluării naționale, examenului de admitere, examenului de certificare a calificării profesionale, concursurilor sau olimpiadelor școlare;
- Punctajul - pentru proba de evaluare a competențelor digitale din cadrul examenului de bacalaureat;
- Nivelul de competență, pentru proba de evaluare a competenței lingvistice într-o limbă de circulație internațională din cadrul examenului de bacalaureat.



Subiectul I

- 1) d
- 2) c
- 3) c
- 4) b
- 5) a

Subiectul al II-lea

1) a) 9752

b) 975 și 952

c) #include <iostream>
using namespace std;

```
int main() {  
    int x;  
    cin >> x;  
    int p = 1;  
    int m = -1;  
    while (p <= x) {  
        int c = (x / p) % 10;  
        if (c > m) {  
            m = c;  
            p = p * 10;  
        }  
    }
```

```

else {
    x = (x / (p * 10)) * p + (x % p);
}

}

if (m >= 0) cout << x;
else cout << "nul";

return 0;
}

```

d) citește x (număr natural)

$p \leftarrow 1$; $m \leftarrow -1$

execută

dacă $p \leq x$ atunci

$c \leftarrow [x / p] \% 10$

dacă $c > m$ atunci

$m \leftarrow c$; $p \leftarrow p * 10$

altfel

$x \leftarrow [x / (p * 10)] * p + x \% p$

cât timp $p \leq x$

dacă $m \geq 0$ atunci scrie x

altfel scrie "nul"

2) Soluția generată imediat înainte de (ocean, ploaie) este (mare, râu), iar soluția generată imediat după (ocean, ploaie) este (ocean, ploaie, râu).

```

3) struct lala {
    char denumire[20];
    struct {
        int nrFire;
        int pretFir;
    } stoc;
} f[10];

```

Subiectul al III - lea

```

1) int abundant (int m) {
    int sumN = 0;
    for (int i = 1; i < m; i++) {
        if (m % i == 0) sumN += i;
    }
    int valN =
    float valN = sumN / m;

```

```

    for (int i = 1; i < m; i++) {
        for (int j = 1; j < i; j++) {
        int sum = 0;
        for (int j = 1; j < i; j++) {
            if (i % j == 0) sum += j;
        }
        if (sum > i) {

```

Numele: POPESCU

Inițiala prenumelui tatălui: C.A.

Prenumele: MĂDĂLIN COSTEL

Școala de proveniență: COL. NAT. DE INFO. „TUDOR VIANU”

Centrul de examen: COL. NAT. DE INFO. „TUDOR VIANU”

Loc. BUCUREȘTI jud. SECTOR 1

Nume și prenume asistent	Semnătura
GABRIEL LORENA	<u>GL</u>
IONESCU TOMA	<u>IV</u>

```

float val = sum / i;
if (val >= valN) return 0;
}
return 1;
}

```

2) #include <iostream>
using namespace std;

int m, n;

int main() {

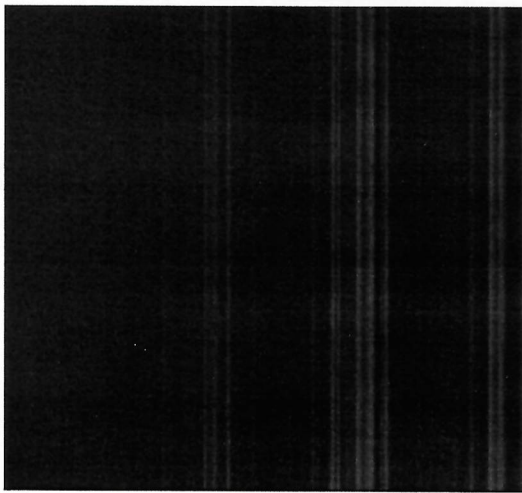
cin >> m >> n;

for (int i = 1; i <= m; i++) {

int maxVal;

maxVal = -1;

int idx;



```
for (int j = 1; j <= n; j++) {  
    int x;  
    cin >> x;  
    if (x > maxVal) {  
        maxVal = x;  
        idx = j;  
    }  
}
```

```
cout << i << ": " << idx << " ";
```

```
return 0;
```

```
3) #include <iostream>  
#include <fstream>  
using namespace std;
```

```
int n[105];  
int p[105];
```

```
ifstream fin("bac.txt");
```

```
int main() {
```

```
    int x;
```

```
    while (fin >> x) {
```

```
        if (x > 99 && x < 1000) {
```

```
            suf[x % 100]++;
```

```
            pref[x / 10]++;
```

```
        }
```

```
        if (x > 999 && x < 10000) {
```

```
            suf[x % 100]++;
```

```
            pref[x / 100]++;
```

```
        }
```

```
    }
```

```
    int ans = 0;
```

```
    for (int i = 10; i < 100; i++) {
```

```
        if (suf[i] != 0 && suf[i] == pref[i]) {
```

```
            ans++;
```

```
        }
```

```
    }
```

```
    cout << ans;
```

```
    fin.close();
```

```
    return 0;
```

```
}
```

a) Algoritmul proiectat are o abordare liniară. Acesta citește pe rând numerele din fișier, iar dacă acestea au mai mult de 2 cifre le marchează sufixul și prefixul într-un doi vectori de frecvență. Apoi, acesta verifică pentru toate sufixele și prefixele valide dacă apar de același număr de ori. Complexitatea de timp este $O(n)$, datorită faptului că programul stoează o singură dată prin toate numerele. Complexitatea de spațiu este $O(1)$, deoarece numărul variabilelor este constant. Astfel, programul este eficient din punctul de vedere al timpului și al memoriei utilizate.