

UPT, Concursul "Alexandru Rogojan"

Model de subiect 2024

1. (3 p.) Care dintre expresiile de mai jos este echivalenta, din punct de vedere logic, cu expresia: $!(x > 0 \ \&\& \ y > 0)$?

- A. $x \leq 0 \ \&\& \ y \leq 0$
- B. $x > 0 \ \&\& \ y > 0$
- C. $x \leq 0 \ || \ y \leq 0$
- D. $!(x > 0) \ \&\& \ !(y > 0)$

2. (5 p.) Considerand ca variabilele x , y și z sunt de tip întreg și memorează numere naturale, indicați expresia aritmetico-logică care are valoarea 1 dacă și numai dacă valoarea variabilei x este strict mai mare decât valoarea oricăreia dintre variabilele y și z .

- A. $(x * z > x * y) \ \&\& \ (y * z > y * x)$
- B. $(y * z > y * x) \ \&\& \ (y * z > z * x)$
- C. $(y * z > x * z) \ \&\& \ (y * x > z * x)$
- D. $(x * y > y * z) \ \&\& \ (x * z > y * z)$

3. (2 p.) Pentru a verifica dacă în tabloul unidimensional $(0, 2, 5, 7, 16, 21, 49)$ există elementul cu valoarea $x=15$, se aplică metoda căutării binare. Indicați succesiunea de elemente din tablou ale căror valori se compară cu valoarea lui x pe parcursul aplicării metodei indicate.

- A. (7, 21, 49, 16)
- B. (0, 2, 5, 6, 16)
- C. (7, 21, 16)
- D. (7, 16)

4. (2 p.) Tablourile unidimensionale A și B au elementele: $A=(20, 17, 12, 10, 3)$, iar $B=(45, 16, 12, 7, 2)$. Selectați opțiunea ce reprezintă elementele tabloului obținut în urma interclasării celor două tablouri, în ordine descrescătoare.

- A. (45, 20, 17, 16, 12, 12, 7, 10, 3, 2)
- B. (45, 20, 17, 16, 12, 12, 10, 7, 3, 2)
- C. (45, 20, 16, 17, 12, 12, 7, 10, 2, 3)
- D. (45, 17, 12, 10, 3)

5. (3 p.) Variabilele i și j sunt de tip întreg. Indicați expresia care poate înlocui punctele de suspensie în algoritmul de mai jos, astfel încât, în urma executării secvenței obținute, să se afișeze valorile indicate. Se considera existente funcțiile $\min()$ și $\max()$ care efectuează calculul exprimat de numele lor.

```
for(int i=1; i<=5; i++){
    for (int j=1; j<=5; j++){
        printf("%u", ...);
    }
    printf("\n");
}
```

și se afișează:

```
4 4 4 4 4
4 3 3 3 3
4 3 2 2 2
4 3 2 1 1
4 3 2 1 0
```

- A. $5 - \min(i, j)$
- B. $5 - \max(i, j)$
- C. $4 - \min(i, j)$
- D. $4 - \max(i, j)$

6. (3 p.) Tabloul unidimensional v are elementele: $v[1] = 4$,

$v[2] = 3$, $v[3] = 2$, $v[4] = 1$. Știind că s-a notat cu $x \Leftrightarrow y$ interschimbarea valorilor variabilelor x și y , indicați cu ce se pot înlocui punctele de suspensie în algoritmul de sortare scris mai jos, pentru a ordona crescător cele patru elemente ale tabloului v .

```
for(int i=1; i<=3; i++){
    int m=i;
    for(int k=i+1; k<=4; k++){
        if (v[k]<v[m]){
            m=k;
            ...
        }
    }
}
```

- A. $v[i] \Leftrightarrow v[m+1]$
- B. $v[k] \Leftrightarrow v[m]$
- C. $v[k+1] \Leftrightarrow v[m]$
- D. $v[i] \Leftrightarrow v[m]$

7. (4 p.) Se consideră un graf neorientat cu 80 de noduri și 3160 muchii. Care este numărul de muchii ce pot fi eliminate astfel încât graful parțial obținut să fie arbore?

- A. 3080
- B. 3081
- C. 81
- D. 79

8. (5 p.) Câte grafuri neorientate, distincte, cu 5 vârfuri, se pot construi? Două grafuri se consideră distincte dacă matricele lor de adiacență sunt diferite.

- A. 2^5
- B. 2^{20}
- C. 2^{10}
- D. 2^4

9. (5 p.) În secvența de mai jos toate variabilele sunt întregi și memorează valori naturale. Indicați o expresie care poate înlocui punctele de suspensie, astfel încât, în urma executării secvenței obținute, variabila z să memoreze câtul împărțirii numărului memorat inițial în x la numărul nenul memorat în y .

```
int z=0;
while(x>=y){
    x=...;
    z=z+1;
}
```

- A. $x-y$
- B. $x/10$
- C. $x\%10$
- D. $x+y$

10. (4 p.) Se consideră un graf orientat cu 6 noduri numerotate de la 1 la 6 și cu mulțimea arcelor formată doar din arcele:

- de la fiecare nod numerotat cu un număr neprim i ($i > 1$) la toate nodurile numerotate cu numere ce aparțin mulțimii divizorilor proprii ai lui i (divizori diferiți de 1 și de i)
- de la nodul numerotat cu 1 la nodul numerotat cu 6

- de la fiecare nod numerotat cu un număr prim i la nodul numerotat cu $i - 1$

Pentru graful dat, care este lungimea celui mai mare drum, format doar din noduri distincte?

- A. 5
B. 4
C. 6
D. 3
11. (3 p.) Se consideră un graf orientat cu 6 noduri, numerotate 1, 2, ..., 6. Arcele grafului sunt de forma $(x, 2 \cdot x)$ pentru orice $x \in \{1, 2, 3\}$ și de forma $(x, x - 1)$ pentru orice $x \in \{2, 3, 4, 5, 6\}$. Care este numărul minim de arce ce trebuie adăugate astfel încât graful să fie tare conex?
- A. 3
B. 2
C. 0
D. 5
E. 4
F. 1
12. (5 p.) Precizați care sunt numărul minim și numărul maxim de arce ale unui graf orientat tare conex cu 15 vârfuri.
- A. 14 și 105
B. 15 și 15
C. 15 și 105
D. 14 și 15
E. 14 și 210
F. 15 și 210
13. (3 p.) Precizați care este numărul maxim de frunze ce apar într-un arbore cu 17 de noduri, dacă fiecare nod are gradul mai mic sau egal cu 4.
- A. 13
B. 12
C. 15
D. 14
E. 11
F. 16
14. (2 p.) Utilizând metoda backtracking se generează toate anagramele (permutări ale literelor) cuvântului **avion**. Precizați câte anagrama încep și se termină cu câte o consoană a alfabetului englez standard.
- A. 60
B. 120
C. 12
D. 6
E. 240
F. 40
15. (3 p.) Se utilizează metoda backtracking pentru a genera șiruri de câte 5 caractere din mulțimea $\{a, 1, b, 2, c, 3, d, 4\}$ cu proprietatea că nu pot apărea două cifre sau două litere alăturate. Știind că primul șir generat este $a1a1a$, iar al doilea este $a1a1b$, indicați șirul obținut imediat înainte de $2c1a1$.
- A. $2c4d4$
B. $2b1a1$
C. $2b4d4$
D. $2b4d3$
E. $3c4d4$
F. $1c4d4$
16. (4 p.) In cazul generării combinarilor de n elemente luate câte p , folosind metoda backtracking, într-o implementare iterativă, condiția necesară și suficientă pe care trebuie să o satisfacă un element de pe nivelul k al stivei pentru a fi considerat valid este:

- A. $st_{i-1} \neq st_i; i \in [1..k] \cap \mathbb{N}$
B. $st_i < st_k; i \in [1..k-1] \cap \mathbb{N}$
C. $st_{i-1} \leq st_i; i \in [1..k-1] \cap \mathbb{N}$
D. $st_{i-1} \leq st_{i+1}; i \in [1..k] \cap \mathbb{N}$
E. $st_{i-1} \neq st_k; i \in [1..k+1] \cap \mathbb{N}$
F. $st_{i-1} \neq st_k; i \in [1..k] \cap \mathbb{N}$

17. (4 p.) Care dintre următoarele probleme permite soluție optimă folosind metoda **greedy**?

- A. Amplasarea damelor pe o tabla de sah astfel încât să nu se atace reciproc.
B. Parcurgerea unui graf în lățime
C. Generarea permutărilor unei mulțimi
D. Parcurgerea unui graf în adâncime
E. Problema continuă a rucsacului
F. Problema discretă a rucsacului

18. (2 p.) Se considera funcția recursivă de mai jos, descrisă în notatie simplificată. Dacă apelul inițial nu se ia în considerare, precizați câte autoapeluri vor fi făcute pentru apelul $F(6)$.

```
int f(int t){
    if (t==1){
        return 0;
    }
    else{
        if (t==2){
            return 1;
        }
        else{
            return F(t-2)+F(t-1);
        }
    }
}
```

- A. 8
B. 6
C. 10
D. 14
E. 11
F. 5

19. (3 p.) Se definește funcția:

$$M_n^k = \begin{cases} M_{n-1}^k + k \cdot M_{n-1}^{k-1} & : k > 0 \\ 1 & : k \leq 0 \end{cases}$$

Dacă se dau numerele naturale n, k ($n \geq k$) și se apelează funcția recursivă scrisă într-un limbaj de programare cunoscut (C++/C sau Pascal) care evaluează funcția definită mai sus, în notatie matematică, valoarea calculată reprezintă:

- A. produsul cartezian
B. nici una dintre celelalte variante
C. afișarea tuturor combinațiilor mulțimii $\{1, 2, \dots, n\}$ luate câte k
D. numărul submulțimilor cu k elemente ale unei mulțimi cu n elemente
E. numărul submulțimilor unei mulțimi cu n elemente
F. afișarea tuturor aranjamentelor mulțimii $\{1, 2, \dots, n\}$ luate câte k

20. (3 p.) Se considera funcția recursivă de mai jos. În urma apelului `numar(824972345)` se va afișa:

```
void numar(int n){
    if (n<=100){
        printf("/");
    }
    else{
        if (n%10<5){
            printf("%d", n%10);
        }
    }
}
```

```

}
numar (n/10);
if (n%10>5) {
    printf("%d", n%10);
}
}
}

```

- A. 4234/97
- B. 4324/97
- C. 4324/79
- D. 3244/97
- E. 4234/79
- F. 2443/97

21. (5 p.) Se considera algoritmul de mai jos. Cate numere naturale impare, distincte, fiecare având exact două cifre, pot fi citite pentru variabila n astfel încât să se afișeze valoarea 3.

```

int n;
scanf("%d", &n);
int z=0;
int p=1;
while (n>0) {
    int c=n%10;
    n=n/10;
    if (c%3==0) {
        z=z+p*(9-c);
        p=p*10;
    }
}

```

- A. 6
- B. 5
- C. 2
- D. 4
- E. 3
- F. 7

22. (3 p.) Pentru afisarea celui mai mic numar natural, care se poate forma cu cifrele unui numar natural, avand k cifre, citit de la tastatura, exista un algoritm optim, care are complexitatea de timp:

- A. $\mathcal{O}(k)$; liniar raportat la numarul de cifre
- B. $\mathcal{O}(k^{10})$; polinomial raportat la numarul de cifre
- C. $\mathcal{O}(k^2)$; patrat raportat la numarul de cifre
- D. $\mathcal{O}(1)$; constant raportat la numarul de cifre
- E. $\mathcal{O}(\log_2(k))$; logaritmic raportat la numarul de cifre
- F. $\mathcal{O}(10^k)$; exponential raportat la numarul de cifre

23. (3 p.) Pe suprafata unui rezervor umplut cu apa se dezvolta o populatie de *Spirogyra* (matasea broastei). Stiind ca in prima zi luciul apei este ocupat de o colonie circulara cu raza de 10 cm, ca in fiecare zi aria suprafetei ocupate se dubleaza, iar in cea de-a 64-a zi colonia ocupa pentru prima data intrega suprafata a lacului, sa se determine dupa cate zile lacul este acoperit in proportie de 25% din suprafata.

- A. 63
- B. 65
- C. 62
- D. 16
- E. 128
- F. 32

24. (4 p.) Se considera algoritmul de mai jos, in notatie pseudocod, simplificata. Cate cifre distincte se vor afisa?

```

for(int i=1; i<=5 i++){
    for(int j=1; j<=5; j++){
        printf("%d", (i+j)%10);
    }
}

```

- A. 8
- B. 25
- C. 24
- D. 26
- E. 9
- F. 10

25. (2 p.) Cate numere naturale din intervalul $[1, 2^N]$ au exact $N - 1$ biti de 1 in reprezentarea in baza 2?

- A. 1
- B. $n \cdot (n - 1) / 2$
- C. N
- D. $N - 1$
- E. $2 \cdot n$

26. (2 p.) Fie o balanță și 27 de bile identice din punct de vedere al formei. Una dintre cele 27 este făcută dintr-un material diferit, aceasta fiind mai grea. De câte măsurători aveți nevoie pentru a identifica cu exactitate bila cea mai grea?

- A. 13
- B. 27
- C. 3
- D. 2
- E. 1

[3] Dacă n este un număr natural impar mai mare decât 2, atunci un graf neorientat cu n noduri, în care fiecare nod este adiacent cu exact $n-1$ noduri, este întotdeauna:

- A. neconex
- B. complet
- C. nu se poate determina
- D. aciclic
- E. eulerian

27. (2 p.) Se considera functia C de mai jos. Ce se afiseaza pentru $f(232493)$?

```

1 void f(unsigned n) {
2     unsigned nr;
3     nr=0;
4     for(int a=9;a>=0;a--){
5         unsigned m;
6         m=n;
7         while (m!=0 && m%10!=a) {
8             m=m/10;
9         }
10        if (m!=0) {
11            nr=nr*10+m%10;
12        }
13    }
14    printf("%u", nr);
15 }

```

- A. 8321
- B. 1238
- C. 9876
- D. 9432
- E. 2349

28. (2 p.) Se considera functia C de mai jos. Ce se afiseaza daca se citesc, in aceasta ordine, numerele 12 7 354 9 1630 0 ?

```

1 void f() {
2     unsigned x;
3     scanf("%u", &x);
4     unsigned n;
5     n=0;
6     while (x!=0) {
7         unsigned y;

```

```
8     y=x;
9     unsigned c;
10    c=0;
11    while(y>0){
12        if(y%10>c){
13            c=y%10;
14        }
15        y=y/10;
16    }
17    n=n*10+c;
18    scanf("%u", &x);
19 }
20 printf("%u", n);
21 }
```

-
- A. 27594
 - B. 27596
 - C. 27599
 - D. 27595
 - E. 275960

[2] Complexitatea de timp pentru algoritmul de sortare prin interclasare a doua tablouri avand m si respectiv n elemente este:

- A. $\mathcal{O}(m+n)$;
- B. $\mathcal{O}(\log(m*n))$;
- C. $\mathcal{O}(n^m)$;
- D. $\mathcal{O}(\log(m+n))$;
- E. $\mathcal{O}(m^n)$;

Model