

# Elemente de programare și gândire algoritmică

## Model de Examen de admitere UPT 2024; Setul 3

8 iulie 2024

- Pentru fiecare item este considerat corect un singur raspuns; Veti marca raspunsul considerat corect pe foaia de raspunsuri;
- In rezolvarea cerintelor se va considera limbajul C standard si programa scolara de liceu;

1. (3 p.) Ce valoarea are variabila `a` după executarea următoarei secvențe de cod?

```
1 int a=1011;
2 if ((a & (1 << 1)) == 0)
3     a=10;
4 else
5     a=11;
```

- A. 16
- B. 160
- C. 80
- D. 40
- E. 10
- F. 11

2. (3 p.) Ce se va afișa ca urmare a executării secvenței de mai jos?

```
1 for (int i = 0; i < 3; i++){
2     for (int j = i-1; j < 3; j++){
3         printf("*");
4     }
5     printf("/");
6 }
```

- A. \*\*\*\*/\*\*/\*\*/
- B. \*\*\*\*/\*\*\*\*/\*\*\*\*/
- C. \*\*\*\*\*/\*\*\*\*\*/\*\*/
- D. \*\*\*\*/\*\*/\*\*/\*\*/
- E. \*\*\*\*/\*\*\*\*/\*\*/
- F. \*\*\*\*/\*\*\*\*/\*\*\*\*/

3. (3 p.) Într-un graf orientat  $G$  cu 6 vârfuri numerotate cu numere distincte de la 1 la 6, există arc de la vârful  $i$  la vârful  $j$  dacă și numai dacă  $i < j$  și  $j - i > 1$ . Care sunt vârfurile din graf ce au gradul interior mai mare decât gradul exterior?

- A. 1, 2, 3
- B. 2, 3, 4
- C. 4, 5, 6
- D. 3, 4, 5
- E. 1, 4, 5
- F. 2, 3, 6

4. (3 p.) Ce valoare va avea variabila `r` după execuția următoarei secvențe de cod:

```
1 int v[] = { 12, 34, 56, 78, 9 };
2 int r = 0;
3 for (int i = 0; i < 5; i++){
4     while (v[i]){
5         r = r * 10 + v[i] / 10;
6         v[i] = v[i] / 10;
7     }
8 }
```

- A. 103050700
- B. 123456789
- C. 214365879
- D. 103355770
- E. 978563412
- F. 103055700

5. (3 p.) Fiecare dintre variabilele întregi  $x$  și  $y$  memorează câte un număr natural. Care dintre expresiile C/C++ de mai jos are valoarea 1 dacă și numai dacă numărul memorat în  $x$  este strict mai mare decât 0 și numărul memorat în  $y$  este strict mai mare decât 5?

- A.  $x * (y + 5) >= 0$
- B.  $x * (y - 25) <= 0$
- C.  $x * (y - 5) <= 0$
- D.  $x * (y - 5) >= 0$
- E.  $x * (y - 5) != 0$
- F.  $x * y - 5 != 0$

6. (3 p.) Care este numărul maxim de valori egale care pot să apară într-un vector de lagături de tip "tata" asociat unui arbore cu rădăcina care conține 10 noduri?

- A. 10
- B. 9
- C. 0
- D. 8
- E. 1
- F. 11

7. (3 p.) Considerând valorile de tip unsigned reprezentate pe 32 de biți, ce se va afișa ca urmare a executării următoarei secvențe de cod?

```
1 unsigned int x = 0xA5;
2 unsigned int y = 0x3C;
3 unsigned int result = (x & y) | ((~x) &
4     (~y));
5 printf("%x\n", result);
```

- A. 66ffffff
- B. 0xf6
- C. 66
- D. ffffffff66
- E. 0x66
- F. f6

8. (3 p.) Ce valori se vor regăsi în vectorul  $v$ , după executarea următoarei secvențe de cod?

```
1 int v[] = {1, 2, 3, 4, 5, 6};
2 while (v[0] < 15){
3     for (int i = 1; i <= 5; i++){
4         v[i] = 1 + v[i-1];
```

```

5     }
6     v[0]=v[5];
7 }

```

- A. 11 7 8 9 10 11  
 B. 22 18 19 20 21 22  
 C. 6 2 3 4 5 6  
 D. 21 17 18 19 20 21  
 E. 16 12 13 14 15 16  
 F. 17 13 14 15 16 17

9. (3 p.) Se consideră un graf neorientat cu nodurile: 1,2,3,4,5,6,7,8 și muchiile: [1,3], [1,7], [2,6], [3,7], [5,2], [5,6], [8,4]. Câte componente conexe are graful?

- A. 3  
 B. 5  
 C. 7  
 D. 1  
 E. 4  
 F. 0

10. (3 p.) Care este valoarea variabilei  $k$ , după execuția următoarei secvențe de cod C?

```

1 int n=100, k=0;
2 int i=n;
3 do{
4     for (int j=i+1; j<n; j++){
5         k++;
6     }
7     i--;
8 }while (i>=0);

```

- A. 100  
 B. 4950  
 C. 9900  
 D. 5000  
 E. 9000  
 F. 10000

11. (3 p.) Se consideră funcția de mai jos. Ce se afiseaza la apelul  $f(2024)$ ?

```

1 void f(int n){
2     if (n!=0){
3         printf("%d", n);
4         while (n){
5             f(n/10);
6             n=n/100;
7         }
8     }
9 }

```

- A. 20242022002  
 B. 20242022022  
 C. 20242022042  
 D. 2024202202  
 E. 20242022024  
 F. 20242022222

12. (3 p.) Ce valoare va avea variabila întregă  $x$  după execuția următoarelor instrucțiuni:

```

1 unsigned x = 13;
2 do {
3     if (x / 2) x = x - 7 ;
4     else x = x + 2 ;
5 } while (x >= 0) ;

```

- A. -2  
 B. -3

- C. -4  
 D. bucla infinita  
 E. 0  
 F. -1

13. (3 p.) Se consideră funcția de mai jos. Câte caractere \* se afisează la apelul  $f(20, 2024)$

```

1 int f(int x, int y) {
2     printf("*");
3     if(x<=1 || y<=1)
4         return 0;
5     if(x>y)
6         return 1+f(f(y/x,y), y);
7     return 1+f(x, f(x,y/x));
8 }

```

- A. 8  
 B. 10  
 C. recursivitate infinită  
 D. 9  
 E. 0  
 F. 7

14. (3 p.) Pentru a verifica dacă într-un tablou unidimensional există elementul cu valoarea  $x=0$ , se aplică metoda căutării binare, iar succesiunea de elemente ale tabloului a căror valoare se compară cu valoarea lui  $x$  pe parcursul aplicării metodei indicate este: **20, 2, 0**. Indicați succesiunea de valori care pot fi, în această ordine, elementele tabloului.

- A. 0,2,20,0,4,40,0,8,80  
 B. 0,401,2,32,4,20,30  
 C. 0,1,2,4,20,40,64  
 D. 0,2,4,8,16,20,32  
 E. 0,2,4,20,30,32,401  
 F. 0,2,4,30,20,32,401

15. (3 p.) Ce valoare va avea variabila  $r$ , după executarea următoarei secvențe de cod C?

```

1 int i = 1 ;
2 while (i <= 8)
3 {
4     printf("%d", i);
5     i=i+2;
6     if (i>= 6)
7         i=i-1;
8 }

```

- A. 13567  
 B. 135678  
 C. 135679  
 D. buclă infinită  
 E. 1356798  
 F. 1356789

16. (3 p.) Se consideră funcția C de mai jos. Ce se afisează dacă, după apel, se citesc, în această ordine, numerele 212 57 3254 91 6310 0?

```

1 void f(){
2     unsigned x;
3     scanf("%u", &x);
4     unsigned y=0;
5     while (x!=0){
6         while(x>9){
7             x=x/10;
8         }
9         y=y*10+x;
10        scanf("%u", &x);
11    }

```

```

12 printf("%u", y);
13 }

```

- A. 69351
- B. 69352
- C. 65936
- D. 25396
- E. 65935
- F. 15211

17. (3 p.) Utilizând metoda *backtracking*, se generează toate numerele impare de cel mult trei cifre din mulțimea 5, 6, 7, 8. Primele 8 soluții generate sunt, în această ordine: 5, 55, 555, 557, 565, 567, 57, 575. Cea de a 12-a soluție generată este:

- A. 67
- B. 55
- C. 65
- D. 655
- E. 587
- F. 658

18. (3 p.) În expresia C/C++ alăturată variabila  $x$  este de tip întreg. Expresia  $x \% 2 + (x + 1) \% 2$ :

- A. are valoarea 0 pentru orice număr natural  $x$ .
- B. are valoarea 1 dacă și numai dacă  $x$  este un număr par.
- C. are valoarea 1 pentru orice număr natural  $x$ .
- D. are valoarea 2 pentru orice număr natural  $x$ .
- E. are o valoare strict mai mare decât 2 pentru orice număr natural  $x$ .
- F. are valoarea 1 dacă și numai dacă  $x$  este un număr impar.

19. (3 p.) Ce efect are executia secvenței de cod C de mai jos?

```

1 unsigned s=0;
2 for (unsigned i=100; i>=1; i=i-3){
3     if (i%3==0)
4         s=s+i;
5 }
6 printf("%u", s);

```

- A. afiseaza suma numerelor divizibile cu 3 între 1 și 100;
- B. bucla infinita/nu afiseaza nimic datorata *overflow* in variabila  $b$ ;
- C. afiseaza -1
- D. afiseaza 99
- E. bucla infinita/nu afiseaza nimic datorata *undeflow* in variabila  $b$ ;
- F. afiseaza 0

20. (3 p.) Pentru rezolvarea careia dintre problemele de mai jos, metoda *backtracking* obtine solutia in numar minim de pasi?

- A. determinarea reuniunii a trei multimi;
- B. determinarea tuturor elementelor din sirul lui Fibonacci, mai mici decat o valoare  $X$ , data;
- C. determinarea tuturor factorilor primi ai unui numar natural
- D. determinarea tuturor modurilor in care se pot alege  $X$  elemente dintr-o multime astfel incat suma acestora sa fie mai mare decat o valoare  $S$
- E. determinarea tuturor divizorilor unui numar de 3 cifre;
- F. determinarea tuturor elementelor ce constituie intersectia a doua multimi de elemente, reprezentate sub forma de tablouri unidimensionale;

21. (3 p.) Se consideră funcția de mai jos. Ce valoare se returneaza la apelul  $g(1, 1)$  si cate caractere `'\n'` se afiseaza?

```

1 int g(int x, int y){
2     putchar('-');
3     if (x >= 0){

```

```

4         if (y == 0)
5             return g(x-1, 1);
6         if (y > 0)
7             return g(x-1, g(x-1, y-1));
8     }
9     return y+1;
10 }

```

- A. recursivitate infinită
- B. returneaza 3 ; 3 caractere `'\n'`
- C. returneaza 3 ; afiseaza 8 caractere `'\n'`
- D. returneaza 3 ; afiseaza 7 caractere `'\n'`
- E. returneaza 7 ; afiseaza 7 caractere `'\n'`
- F. returneaza 7 ; afiseaza 3 caractere `'\n'`

22. (3 p.) Pentru algoritmul afisare a intersectiei a doua multimi, reprezentate sub forma de tablouri unidimensionale, fiecare avand un numar de  $n$  elemente, iar elementele sunt valori naturale mai mici decat 100, se poate proiecta un algoritm optim de rezolvare, avand numarul de pasi egal cu:

- A.  $\mathcal{O}(2^n)$
- B.  $\mathcal{O}(n^2)$
- C.  $\mathcal{O}(3^n)$
- D.  $\mathcal{O}(n^3)$
- E.  $\mathcal{O}(n \cdot \log_2 n)$
- F.  $\mathcal{O}(n)$

23. (3 p.) Numărul de elemente nenule ale matricei de adiacență asociată unui arbore cu  $n$  noduri este:

- A.  $n^2 + 1$
- B.  $n^2 - (n - 1)$
- C.  $2^{n-1}$
- D.  $n^2$
- E.  $n(n - 1) + n$
- F.  $2 \cdot (n - 1)$

24. (3 p.) Ce se afiseaza in urma executarii urmatoarei secvențe C, daca toate datele se considera pe 32 de biti, fara semn?

```

1 unsigned int a = 0x5A;
2 unsigned int b = 0xC3;
3 unsigned int result;
4 result = (a ^ b) & (~a | b);
5 printf("%x\n", result);

```

- A. 80
- B. 128
- C. 81
- D. 99
- E. 129
- F. ff

25. (3 p.) Se utilizează metoda *backtracking* pentru a genera toate anagramele (permutări ale caracterelor) distincte ale cuvântului *scarp*. Câte dintre soluții vor avea primul și ultimul caracter o consoană. Se consideră consoane caracterele din mulțimea literelor alfabetului englez standard din care se elimina elementele din multimea  $\{a, e, i, o, u\}$ .

- A. 96
- B. 24
- C. 48
- D. 18
- E. 72
- F. 36

26. (3 p.) Se consideră un arbore binar complet cu 63 de noduri. Care este înălțimea acestui arbore?

- A. 4
- B. 62
- C. 64

- D. 5
- E. 6
- F. 1

27. (3 p.) Precizați complexitatea timp pentru următorul algoritm

```

1 void F(int n){
2     if (n<=1){
3         putchar('1');
4     }
5     else{
6         int m=n/2;
7         F(m);
8         F(m);
9     }
10 }
```

- A.  $\mathcal{O}(n)$
- B.  $\mathcal{O}(2^n)$
- C.  $\mathcal{O}(n \log_2 n)$
- D.  $\mathcal{O}(1)$
- E.  $\mathcal{O}(n^2)$
- F.  $\mathcal{O}(n \cdot 2^n)$

28. (3 p.) Tablourile unidimensionale A și B au valorile A=(18, 15, 9, 5, 1) și B=(40, 15, 13, 4, 1). Pentru a determina al 5-lea element obținut în urma interclasării tablourilor în ordine crescătoare, se compară elementul cu valoarea  $x_a$  din A cu elementul cu valoarea  $x_b$  din B. Indicați valorile lui  $x_a$  și  $x_b$ .

- A.  $x_a=9$  și  $x_b=13$
- B.  $x_a=9$  și  $x_b=5$
- C.  $x_a=9$  și  $x_b=4$
- D.  $x_a=5$  și  $x_b=4$
- E.  $x_a=5$  și  $x_b=13$
- F.  $x_a=4$  și  $x_b=13$

29. (3 p.) Ce se afișează, în urma executării secvenței de mai jos, dacă toate datele se consideră pe 32 de biți, fără semn?

```

1 unsigned int result = 0;
2 unsigned int x = 0xF0;
3 unsigned int y = 0x0F;
4 for (int i = 0; i < 4; i++) {
5     result = result | (x & (1 << i)) << 4;
6     result = result | (y & (1 << i)) << i;
7 }
8 printf("%x\n", result);
```

- A. ff
- B. 50
- C. 55
- D. 0
- E. fff
- F. 0x55

30. (3 p.) Ce se afișează ca urmare a executării secvenței de cod C de mai jos?

```

1 int count = 0;
2 int array[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
3 for (int i = 0; i < 10; i++) {
4     if (array[i] % 2 == 0) {
5         for (int j = 0; j < array[i]; j++) {
6             if (j % 3 == 0) {
7                 count++;
8             }
9         }
10    }
11 }
12 printf("%d\n", count);
```

- A. 15
- B. 18
- C. 10
- D. 16
- E. 14
- F. 12