

Evaluare M1

Nume elev.....

| | |
|----|---|
| 1. | <p>Variabilele x și y sunt de tip întreg, x memorând valoarea 8, iar y valoarea 6. Care dintre expresiile C/C++ de mai jos are valoarea 0?</p> <div style="display: flex; justify-content: space-between;"> <div>a. $3*x-4*y==0$</div> <div>b. $(x+y)/2 > x*y+1$</div> </div> <div style="display: flex; justify-content: space-between;"> <div>c. $!(x/2+2==y)$</div> <div>d. $x-y+3!=0$</div> </div> |
| 2 | <p>Variabila y memorează simultan datele fiecăruia dintre cei 30 de elevi dintr-o clasă: codul de identificare la un examen (un număr natural din intervalul $[1, 10^3]$) și două note obținute (numere reale). Expresiile C/C++ de mai jos au ca valori codul de identificare și cele două note ale celui de al treilea elev din clasă. Scrieți definiția unei structuri cu eticheta elev, care permite memorarea datelor despre un elev, și declarați corespunzător variabila y.</p> <pre>y[2].cod y[2].nota1 y[2].nota2</pre> |
| 3 | <p>Scrieți ce se afișează în urma executării secvenței de mai jos, în care variabilele s și t permit memorarea câte unui șir de cel mult 50 de caractere.</p> <pre>strcpy(s,"vorbeste"); s[3]=s[0]; s[5]=s[2]; s[0]=s[1]+1; s[2]=s[1]-2; s[6]=s[4]-1; strcpy(t,s); t[3]='\0'; cout<<t<<endl<<s+3; printf("%s\n%s",t,s+3);</pre> |
| 4 | <div style="display: flex;"> <div style="flex: 1;"> <p>Pentru funcția f definită alăturat, stabiliți care este valoarea f(5). Dar f(23159)?</p> </div> <div style="flex: 1;"> <pre>int f(int n){ int c; if (n==0) return 9; else {c=f(n/10); if (n%10<c) return n%10; else return c;} }</pre> </div> </div> |
| 5 | <div style="display: flex;"> <div style="flex: 1;"> <p>Algoritmul alăturat este reprezentat în pseudocod. S-a notat cu a÷b restul împărțirii numărului natural a la numărul natural nenul b și cu [c] partea întreagă a numărului real c.</p> <ol style="list-style-type: none"> Scrieți ce se afișează dacă se citește, în această ordine, numerele 5 și 2. (6p.) Dacă pentru variabila k se citește valoarea 6, scrieți cea mai mică și cea mai mare valoare din intervalul $[1, 30]$ care pot fi citite pentru variabila n astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, ultimul număr afișat să fie 12. (6p.) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.) Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat ultima structură pentru...execută cu o structură repetitivă de alt tip. </div> <div style="flex: 1;"> <pre>citește n,k (numere naturale nenule) t←1 pentru i←1,[n/k] execută pentru j←1,k execută scrie 2*t,' ' t←t+1 pentru i←n÷k,1,-1 execută scrie 3*t,' '</pre> </div> </div> |
| 6 | <p>Subprogramul nrfp are doi parametri:</p> <ul style="list-style-type: none"> n, prin care primește un număr natural ($n \in [2, 10^5]$); m, prin care furnizează numărul din intervalul închis $[2, n]$ care are cei mai mulți factori primi; dacă există mai multe numere cu această proprietate, subprogramul îl returnează pe cel mai mare dintre ele. <p>Scrieți definiția completă a subprogramului. Exemplu: dacă n=100 atunci, în urma apelului, m=90.</p> |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--|--|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | <p>Scrieți un program C/C++ care citește de la tastatură numărul natural n ($n \in [5, 50]$) și elementele unui tablou bidimensional cu n linii și n coloane, numere naturale din intervalul $[0, 10^2]$. Programul afișează pe ecran suma numerelor din zona delimitată de cele două diagonale și ultima coloană a tabloului, ca în exemplu.</p> <p>Exemplu: pentru $n=7$ și tabloul alăturat, se afișează pe ecran 12.</p> | <table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>5</td><td>6</td></tr><tr><td>7</td><td>8</td><td>9</td><td>0</td><td>3</td><td>1</td><td>2</td></tr><tr><td>4</td><td>6</td><td>8</td><td>0</td><td>1</td><td>1</td><td>3</td></tr><tr><td>8</td><td>6</td><td>3</td><td>6</td><td>2</td><td>4</td><td>7</td></tr><tr><td>5</td><td>7</td><td>9</td><td>2</td><td>2</td><td>5</td><td>8</td></tr><tr><td>1</td><td>4</td><td>7</td><td>0</td><td>5</td><td>3</td><td>6</td></tr><tr><td>9</td><td>2</td><td>5</td><td>8</td><td>5</td><td>9</td><td>1</td></tr></table> | 1 | 2 | 3 | 4 | 5 | 5 | 6 | 7 | 8 | 9 | 0 | 3 | 1 | 2 | 4 | 6 | 8 | 0 | 1 | 1 | 3 | 8 | 6 | 3 | 6 | 2 | 4 | 7 | 5 | 7 | 9 | 2 | 2 | 5 | 8 | 1 | 4 | 7 | 0 | 5 | 3 | 6 | 9 | 2 | 5 | 8 | 5 | 9 | 1 |
| 1 | 2 | 3 | 4 | 5 | 5 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | 8 | 9 | 0 | 3 | 1 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | 6 | 8 | 0 | 1 | 1 | 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | 6 | 3 | 6 | 2 | 4 | 7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | 7 | 9 | 2 | 2 | 5 | 8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 4 | 7 | 0 | 5 | 3 | 6 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | 2 | 5 | 8 | 5 | 9 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 8 | <p>Fișierul <code>bac.txt</code> conține un șir de cel mult 10^6 numere naturale din intervalul $[0, 10^9]$. Se cere să se determine și să se afișeze pe ecran, separate printr-un spațiu, ultimele două numere impare (nu neapărat distincte) din șirul aflat în fișier, sau mesajul <code>nu exista</code>, dacă nu există două astfel de numere. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.</p> <p>Exemplu: dacă fișierul conține valorile 122 <u>1635</u> 628 <u>1413</u> <u>1647</u> 900 <u>3001</u> 4252 se afișează pe ecran 1647 3001</p> <p>a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.</p> <p>b. Scrieți programul C/C++ corespunzător algoritmului proiectat.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

Evaluare M1

Nume elev.....

| | | |
|---|---|--|
| 1 | <p>1. Variabile x, y și z sunt de tip întreg. Indicați o expresie C/C++ care are valoarea 1 dacă și numai dacă x, y și z au valori identice.</p> <p>a. x==y && x==z</p> <p>b. x==y==z</p> <p>c. !(x!=y && x!=z)</p> <p>d. !(x!=y!=z)</p> | |
| 2 | <p>Variabila c memorează simultan următoarele date despre o carte dintr-o bibliotecă: titlul (un șir de maximum 50 de caractere), numele autorului/autorilor și numărul de exemplare (maximum 10²); o carte poate avea cel mult 10 autori, fiecare având un nume de maximum 50 de caractere. Știind că expresiile C/C++ de mai jos au ca valori două șiruri de caractere, reprezentând titlul, respectiv numele primului autor al cărții, și un număr natural reprezentând numărul de exemplare ale acestei cărți, scrieți definiția unei structuri cu eticheta carte, care permite memorarea datelor despre o carte, și declarați corespunzător variabila c.</p> <p>c.titlu c.autor[0] c.nrExemplare</p> | |
| 3 | <p>Variabila i este de tip întreg, iar variabilele s și aux permit memorarea câte unui șir cu cel mult 15 caractere. Scrieți ce se afișează pe ecran în urma executării secvenței de program alăturate.</p> <pre>strcpy(s,"voalata"); cout<<strlen(s); printf("%d",strlen(s)); i=0; while (i<strlen(s)) if (strchr("aeiou",s[i])!=NULL) { strcpy(aux,s+i+1);strcpy(s+i,aux); i=i+1; } else i=i+2; cout<<s; printf("%s",s);</pre> | |
| 4 | <p>Subprogramul afis este definit alăturat. Indicați ce se afișează în urma apelului de mai jos. afis(12345);</p> <pre>void afis (int a) { cout<<" "; printf("+"); if(a>0) { afis(a/1000); cout<<a; printf("%d",a); } cout<<" "; printf("+"); }</pre> | |
| 5 | <p>Algoritmul alăturat este reprezentat în pseudocod. S-a notat cu [c] partea întreagă a numărului real c.</p> <p>a. Scrieți valoarea afișată dacă se citesc, în această ordine, numerele 12, 7, 354, 9, 630, 0.</p> <p>b. Scrieți o secvență de numere din intervalul [0, 10⁴) care pot fi citite, în această ordine, astfel încât, în urma executării algoritmului, să se afișeze 321.</p> <p>c. Scrieți programul C/C++ corespunzător algoritmului dat.</p> <p>d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat a doua structură repetă...până când cu o structură repetitivă cu test inițial.</p> <pre>citește x (număr natural nenul) y←0 repetă dacă x>9 atunci repetă x←[x/10] până când x≤9 y←y*10+x citește x (număr natural) până când x=0 scrie y</pre> | |
| 6 | <p>Subprogramul produs are doi parametri:</p> <ul style="list-style-type: none"> n, prin care primește un număr natural ($n \in [0, 10^9]$); p, prin care furnizează produsul cifrelor pare distincte din scrierea acestuia, sau -1 dacă nu există astfel de cifre. <p>Scrieți definiția completă a subprogramului. Exemplu: dacă n=1622325, după apel p=12 (12=6·2), iar dacă n=122325, după apel p=2.</p> | |

| | |
|---|--|
| 7 | |
| 8 | <p>Fișierul bac.in conține un șir de numere naturale distincte, din intervalul $[1, 10^9]$. Numerele din șir sunt separate prin câte un spațiu și cel puțin trei dintre ele au penultima cifră 2 și ultima cifră 0.</p> <p>Se cere să se afișeze pe ecran cele mai mici trei numere din șir cu proprietatea că au penultima cifră 2 și ultima cifră 0. Numerele determinate sunt afișate în ordine crescătoare, separate prin câte un spațiu. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.</p> <p>Exemplu: dacă fișierul conține numerele 9731 50 112 <u>20</u> 8 16 85<u>20</u> 3 25<u>20</u> 15<u>20</u> pe ecran se vor afișa, în această ordine, numerele: 20 1520 2520</p> <p>a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)</p> <p>b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)</p> |

Evaluare M1

Nume elev.....

| | |
|---|---|
| 1 | <p>Variabila x este de tip întreg și memorează un număr nenul. Indicați expresia C/C++ cu valoarea 1 dacă și numai dacă 20 este multiplu al lui x.</p> <p>a. $20 / (20/x) == 0$ b. $20 / (20\%x) == 0$ c. $20\% (20\%x) == 0$ d. $20\% (20/x) == 0$</p> |
| 2 | <p>Variabila s memorează simultan următoarele date despre fiecare dintre cele 20 de spectacole din cadrul unei stagiuni: un cod, numărul de actori și vârstele acestora. La un spectacol sunt maximum 10 actori. Expresiile C/C++ de mai jos au ca valori numere naturale și reprezintă codul primului spectacol, numărul de actori care joacă în acest spectacol, respectiv vârsta celui de al 5-lea astfel de actor. Scrieți definiția unei structuri cu eticheta spectacol, care permite memorarea datelor despre un spectacol, și declarați corespunzător variabila s.</p> <p>s[0].cod s[0].nrActori s[0].varsta[4] (6p.)</p> |
| 3 | <p>Variabila i este de tip întreg, iar variabila s poate memora un șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)</p> <pre>strcpy(s, "stilou"+4); cout<<s<<endl; printf("%s\n", s); strncpy(s, "stilou", 4); s[4]='\0'; for(i=0; i<4; i++) if(i%2==0) s[i]=s[0]+i-1; else s[i]=s[0]+3*(2*i/3-1); cout<<s; printf("%s", s);</pre> |
| 4 | <p>Subprogramul f este definit alăturat. Indicați ce se va afișa în urma apelului de mai jos. f(54321);</p> <pre>void f(int n) { if (n!=0) { f(n/10); if (n%2==0) cout<<n%10; printf("%d", n%10); } }</pre> |
| 5 | <p>1. Algoritmul alăturat este reprezentat în pseudocod.</p> <p>a. Scrieți numărul afișat în urma executării algoritmului dacă pentru n se citește valoarea 7. (6p.)</p> <p>b. Scrieți două numere din intervalul $[10, 10^2]$ care pot fi citite astfel încât, pentru fiecare dintre acestea, în urma executării algoritmului, să se afișeze 10. (6p.)</p> <p>c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)</p> <p>d. Scrieți în pseudocod un algoritm, echivalent cu cel dat, înlocuind adecvat una dintre structurile cât timp...execută cu o structură repetitivă de tip pentru...execută. (6p.)</p> <pre> citește n (număr natural nenul) nr ← 0; i ← 1 cât timp i ≤ n execută x ← 0; y ← 1; j ← 1 cât timp j < i execută r ← 2*x - y; x ← y; y ← r j ← j + 1 i ← i + 1 dacă y > 0 atunci nr ← nr + 1 scrie nr</pre> |
| 6 | <p>Subprogramul produs are doi parametri:</p> <ul style="list-style-type: none"> n, prin care primește un număr natural ($n \in [0, 10^9]$); p, prin care furnizează produsul cifrelor pare distincte din scrierea acestuia, sau -1 dacă nu există astfel de cifre. <p>Scrieți definiția completă a subprogramului.</p> <p>Exemplu: dacă n=1622325, după apel p=12 ($12=6 \cdot 2$), iar dacă n=122325, după apel p=2.</p> |

| | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|--|---|-----------------|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 7 | <p>Scrieți un program C/C++ care citește de la tastatură un număr natural, n ($n \in [2, 20]$), și construiește în memorie un tablou bidimensional cu n linii și n coloane, astfel încât fiecare linie a sa are proprietățile:</p> <ul style="list-style-type: none">• elementul situat pe diagonala secundară are valoarea n;• începând de la diagonala secundară, spre stânga, elementele formează un șir strict descrescător de numere consecutive, iar începând de la diagonala secundară, spre dreapta, elementele formează tot un șir strict descrescător de numere consecutive. <p>Programul afișează pe ecran tabloul construit, fiecare linie a tabloului pe câte o linie a ecranului, cu elementele aflate pe aceeași linie separate prin câte un spațiu.</p> <p>Exemplu: dacă $n=5$ se afișează pe ecran tabloul alăturat.</p> <table><tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr><tr><td>2</td><td>3</td><td>4</td><td>5</td><td>4</td></tr><tr><td>3</td><td>4</td><td>5</td><td>4</td><td>3</td></tr><tr><td>4</td><td>5</td><td>4</td><td>3</td><td>2</td></tr><tr><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td></tr></table> | 1 | 2 | 3 | 4 | 5 | 2 | 3 | 4 | 5 | 4 | 3 | 4 | 5 | 4 | 3 | 4 | 5 | 4 | 3 | 2 | 5 | 4 | 3 | 2 | 1 |
| 1 | 2 | 3 | 4 | 5 | | | | | | | | | | | | | | | | | | | | | | |
| 2 | 3 | 4 | 5 | 4 | | | | | | | | | | | | | | | | | | | | | | |
| 3 | 4 | 5 | 4 | 3 | | | | | | | | | | | | | | | | | | | | | | |
| 4 | 5 | 4 | 3 | 2 | | | | | | | | | | | | | | | | | | | | | | |
| 5 | 4 | 3 | 2 | 1 | | | | | | | | | | | | | | | | | | | | | | |
| 8 | <p>Fișierele bac1.txt și bac2.txt conțin numere naturale din intervalul $[1, 10^5]$: fișierul bac1.tx conține pe prima linie un număr n_1, iar pe a doua linie un șir de n_1 numere, iar fișierul bac2.tx conține pe prima linie un număr n_2, iar pe a doua linie un șir de n_2 numere. Numerele aflate p aceeași linie a unui fișier sunt ordonate crescător și sunt separate prin câte un spațiu.</p> <p>Se cere să se afișeze pe ecran, în ordine crescătoare, separate prin câte un spațiu, numerele divizibil cu 5 care se găsesc doar în unul dintre șirurile aflate în cele două fișiere. Dacă nu există niciun astfel d număr, se afișează pe ecran mesajul nu exista. Proiectați un algoritm eficient din punctul de veder al memoriei utilizate și al timpului de executare.</p> <p>Exemplu: dacă fișierul bac1.txt conține numerele</p> <table><tr><td>7</td></tr><tr><td>1 2 3 4 7 20 60</td></tr></table> <p>iar fișierul bac2.txt conține numerele</p> <table><tr><td>9</td></tr><tr><td>3 5 7 8 9 10 12 20 24</td></tr></table> <p>pe ecran se afișează, în această ordine, numerele 5 10 60.</p> <p>a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.</p> <p>b. Scrieți programul C/C++ corespunzător algoritmului proiectat.</p> | 7 | 1 2 3 4 7 20 60 | 9 | 3 5 7 8 9 10 12 20 24 | | | | | | | | | | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 2 3 4 7 20 60 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9 | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 5 7 8 9 10 12 20 24 | | | | | | | | | | | | | | | | | | | | | | | | | | |