

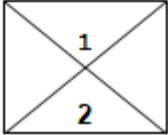
## VARIANTA 8 - Rezolvare

### Subiectul 1

1.	Se ține cont de prioritatea operatorilor în evaluarea expresiei. Pentru răspunsul <b>b)</b> se acordă <b>4p</b> .
2.	Accesarea denumirii celui de-al treilea element al tabloului unidimensional este <code>m[2].denumire</code> (C++/C), <code>m[3].denumire</code> (Pascal), iar pentru accesarea primei litere utilizăm notația <code>m[2].denumire[0]</code> (C++/C) / <code>m[3].denumire[1]</code> (Pascal). Pentru răspunsul <b>b)</b> se acordă <b>4p</b> .
3.	Se generează șirurile: 1111, 1110, 1101, 1100, 1011 etc. După șirul 1001 se află șirurile 1000, 0111. Pentru răspunsul <b>a)</b> se acordă <b>4p</b> .
4.	Numărul minim de muchii ce se pot elimina este 3. Pentru precizarea răspunsului <b>c)</b> se acordă <b>4p</b> .
5.	Pentru ca numărul nodurilor terminale să fie minim trebuie ales ca rădăcină unul din nodurile terminale actuale. Deci oricare din nodurile 8, 9, 4 sau 7 dacă sunt alese ca rădăcină arborele va avea numai 3 noduri terminale. Pentru precizarea răspunsului <b>a)</b> se acordă <b>4p</b> .

### Subiectul 2

1.	a)	Numim cifra de control a unui număr cifra obținută prin adunarea repetată a cifrelor sale până când numărul obținut este mai mic decât 10. Exemplu: cifra de control a numărului 557 este 8 deoarece $5+5+7=17$ . Cum $17>10$ adunăm cifrele numărului 17, $1+7=8$ , și $8<10$ deci cifra de control a numărului 557 este 8. Algoritmul verifică dacă un multiplu al cifrei de control corespunzător numărului $n$ se găsește în intervalul $[a, b]$ . În caz afirmativ se afișează 1. Pentru datele de intrare 33, 30, 10 se verifică dacă un multiplu al cifrei de control pentru numărul 33 (care este 6) se află în intervalul $[10, 30]$ . Cum 12 aparține intervalului $[10, 30]$ , se va afișa 1, răspuns pentru care se acordă (6p.)
	b)	Pentru afișarea valorii 0, în cazul în care $n=99$ și $a=55$ trebuie căutată o valoare pentru $b$ astfel încât nici un multiplu al cifrei de control să nu aparțină intervalului $[a, b]$ . Prin urmare cea mai mică valoare a lui $b$ este 55. Pentru precizarea corectă a valorii variabilei $b$ se vor acorda (6p.)
	c)	Ultima structură <b>cât timp ... execută</b> poate fi înlocuită cu structura liniară: $a \leftarrow ([a/n] + 1) * n$ (6p.)
	d)	Pentru declararea variabilelor se acordă 1p, pentru citirea datelor se acordă 1p, pentru afișarea rezultatului se acordă 1p, pentru instrucțiunea de decizie se acordă 1p., pentru cele trei instrucțiuni repetitive se acordă 3p., iar pentru instrucțiuni de atribuire se acordă 2p. Pentru structura corectă a programului se acordă 1p., în total 10p.

	<p>O posibilă implementare în limbajul de programare studiat este:</p> <table border="1"> <thead> <tr> <th>Limbajul Pascal</th><th>Limbajul C/C++</th></tr> </thead> <tbody> <tr> <td> <pre> var n,a,b,s:integer; begin   read(n,a,b);   if(a&gt;b) then     begin       b:=b+a;       a:=b-a;       b:=b-a;     end;   while (n&gt;9) do   begin     s:=0;     while (n&gt;0) do       begin         s:=s+n mod 10;         n:=n div 10;       end;     n:=s;   end;   while (a&lt;=b) and (a mod n&lt;&gt;0)   do     a:=a+1;     if(a&lt;=b) then write('1')       else write('0');   end. </pre> </td><td> <pre> #include &lt;iostream&gt; #include &lt;stdio.h&gt; using namespace std; int main() {   int n,a,b;   cin&gt;&gt;n&gt;&gt;a&gt;&gt;b;   scanf("%d %d %d",n,a,b);   if(a&gt;b)   {     b=b+a;     a=b-a;     b=b-a;   }   while(n&gt;9)   {     int s=0;     while(n)     {       s=s+n%10;       n=n/10;     }     n=s;   }   while (a%n!=0)     a++;   if(a&lt;=b) cout &lt;&lt; 1;   else cout&lt;&lt;0; } </pre> </td></tr> </tbody> </table>	Limbajul Pascal	Limbajul C/C++	<pre> var n,a,b,s:integer; begin   read(n,a,b);   if(a&gt;b) then     begin       b:=b+a;       a:=b-a;       b:=b-a;     end;   while (n&gt;9) do   begin     s:=0;     while (n&gt;0) do       begin         s:=s+n mod 10;         n:=n div 10;       end;     n:=s;   end;   while (a&lt;=b) and (a mod n&lt;&gt;0)   do     a:=a+1;     if(a&lt;=b) then write('1')       else write('0');   end. </pre>	<pre> #include &lt;iostream&gt; #include &lt;stdio.h&gt; using namespace std; int main() {   int n,a,b;   cin&gt;&gt;n&gt;&gt;a&gt;&gt;b;   scanf("%d %d %d",n,a,b);   if(a&gt;b)   {     b=b+a;     a=b-a;     b=b-a;   }   while(n&gt;9)   {     int s=0;     while(n)     {       s=s+n%10;       n=n/10;     }     n=s;   }   while (a%n!=0)     a++;   if(a&lt;=b) cout &lt;&lt; 1;   else cout&lt;&lt;0; } </pre>
Limbajul Pascal	Limbajul C/C++				
<pre> var n,a,b,s:integer; begin   read(n,a,b);   if(a&gt;b) then     begin       b:=b+a;       a:=b-a;       b:=b-a;     end;   while (n&gt;9) do   begin     s:=0;     while (n&gt;0) do       begin         s:=s+n mod 10;         n:=n div 10;       end;     n:=s;   end;   while (a&lt;=b) and (a mod n&lt;&gt;0)   do     a:=a+1;     if(a&lt;=b) then write('1')       else write('0');   end. </pre>	<pre> #include &lt;iostream&gt; #include &lt;stdio.h&gt; using namespace std; int main() {   int n,a,b;   cin&gt;&gt;n&gt;&gt;a&gt;&gt;b;   scanf("%d %d %d",n,a,b);   if(a&gt;b)   {     b=b+a;     a=b-a;     b=b-a;   }   while(n&gt;9)   {     int s=0;     while(n)     {       s=s+n%10;       n=n/10;     }     n=s;   }   while (a%n!=0)     a++;   if(a&lt;=b) cout &lt;&lt; 1;   else cout&lt;&lt;0; } </pre>				
2.	<p>Secvența interschimbă elementele triunghiurilor <b>unu</b> și <b>doi</b> inclusiv elementele de pe diagonale (vezi desenul de mai jos) din tabloul bidimensional, celelalte elemente rămânând neschimbate:</p>  <pre> 5 5 5 5 5 2 4 4 4 2 3 3 3 3 3 4 2 2 2 4 1 1 1 1 1 </pre> <p>Pentru răspuns corect se acordă <b>(6p.)</b>.</p> <p>Pentru răspuns parțial corect (interschimbă doar elementele celor două diagonale sau numai elementele dintre diagonale) se acordă doar <b>(3p.)</b></p>				
3.	<p>a) La apelul <math>f(2, 7, 5)</math> variabila <math>m</math> primește valoarea 4 și cum <math>c &gt; 4</math> se execută apelul recursiv <math>f(5, 7, 5)</math>. La acest apel variabila <math>m</math> primește valoarea 6. Cum <math>c &lt; 6</math> noul apel recursive va fi <math>f(5, 5, 5)</math>. La acest apel <math>m = c</math> deci se va returna valoarea 10. Pentru răspunsul 10 se acordă <b>(3p.)</b></p>				

- b) Dacă  $a=4$  și  $b=9$  variabila  $m$  primește valoarea 6. În acest caz variabila  $c$  trebuie să fie mai mică decât 6 sau mai mare decât 6 (în caz contrar se va returna 13). Dacă  $c>6$  va avea loc apelul recursiv  $f(7,9,c)$  caz în care se va returna o valoare  $>10$ . Prin urmare  $c<6$ . Dacă  $c<5$  atunci se va returna o valoare mai mică decât 10 deci  $c=5$ , răspuns pentru care se primește (3p.)

### Subiectul 3

1. O posibila implementare a subprogramului este:

Limbajul Pascal	Limbajul C/C++
<pre>function suma (n,s:integer;a:array[1..100] of integer):integer; var i,lmax,lc,ic:integer; begin   lmax:=-1;   for i:=1 to n do   begin     ic:=i;lc:=0;     while(lc&lt;&gt;s) and (ic&lt;=n)do     begin       lc:=lc+a[ic];inc(ic);     end;     if(lc=s) then       if(ic-i&gt;lmax)then         lmax:=ic-i;   end;   suma:=lmax; end;</pre>	<pre>int suma_x(int n,int a[],int s) {   int lc,ic,lmax=-1;   for(int i=0; i&lt;n; i++)   {ic=i;lc=0;     while(lc!=s&amp;&amp;ic&lt;n)     {lc=lc+a[ic];ic++;}     if(lc==s)       if(ic-i&gt;lmax)lmax=ic-i;   }   return lmax; }</pre>

Pentru antet subprogram (se acordă câte 1p. pentru fiecare aspect al antetului: structură, parametri de intrare, conform cerinței.), 3p; determinare a valorii cerute 3p; declarare a tuturor variabilelor locale 1p; corectitudine globală a subprogramului 3p. Total 10p.

2. Pentru declararea corectă a variabilelor se acordă 1p., pentru citirea corectă a șirului de caractere se acordă 1p., pentru identificarea cuvintelor în care vocalele apar în ordine alfabetică 4p., pentru utilizarea corectă a funcțiilor specifice șirurilor de caractere se acordă 2p., pentru afișarea cuvintelor în care vocalele apar în ordine alfabetică se acordă 1p. Pentru corectitudinea sintactică a programului se acordă 1p - în total 10p.

Un exemplu de implementare fi:

Limbajul Pascal	Limbajul C/C++
<pre>var   s:string;   p:string[20];   i,k,j:integer;   vb:boolean;   vocale:set of char; begin   vocale:   =['a','e','i','o','u'];   readln(s);</pre>	<pre>#include &lt;iostream&gt; #include &lt;string.h&gt; using namespace std;  int main() {   char s[256],*p,v[10];   cin.get(s,256);   p=strtok(s," ");</pre>

	<pre>repeat   k:=pos(' ',s);   p:=copy(s,1,k-1);   vb:=true;   if(k=0) then     p:=s;   for i:=1 to length(p)-1 do     for j:=i+1 to length(p) do       if(p[i] in vocale) and       (p[j] in vocale) and       (p[i]&gt;=p[j]) then         vb:=false;   if(vb) then write(p, ' ');   delete(s,1,length(p)+1); until k=0; end.</pre>	<pre>while(p) {   int vb=1;   for(int i=0; i&lt;strlen(p)-1     &amp;&amp; vb; i++)     for(int j=i+1; j&lt;strlen(p)       &amp;&amp;vb; j++)       if(strchr("aeiou",p[i])         &amp;&amp;         strchr("aeiou",p[j])         &amp;&amp; p[i]&gt;=p[j])         vb=0;   if(vb)     cout&lt;&lt;p&lt;&lt;" ";   p=strtok(NULL, " "); }</pre>				
3.	<p>a) Se acordă <b>2p.</b> pentru operațiile cu fișiere (declarare, nume corect și deschidere pentru citire și scriere), <b>1p.</b> pentru citirea tuturor numerelor din fișier, <b>2p.</b> pentru un algoritm de sortare corect, <b>1p.</b> pentru determinarea valorilor cerute, <b>1p.</b> pentru afișarea valorilor în ordine crescătoare în fișier <b>1p.</b> pentru corectitudinea globală (declararea variabilelor, structura programului, sintaxa instrucțiunilor etc.).<b>Total 8 puncte.</b></p> <p>Un exemplu de implementare poate fi:</p> <table><tr><th>Limbajul Pascal</th><th>Limbajul C/C++</th></tr><tr><td><pre>var     f,g:text;     v:array[1..10] of integer;     i,x,y:integer; begin     assign(f,'date.in');     reset(f);     assign(g,'date.out');     rewrite(g);     while not eof(f) do         begin             read(f,x);             y:=x mod 10;             while(x&gt;9) do                 x:=x div 10;             if(x=y) then                 inc(v[x]);             end;             for i:=1 to 9 do                 write(g, v[i], ' ');             close(f);             close(g);             readln;         end.</pre></td><td><pre>#include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("date.in"); FILE *f=fopen("date.in" ,"r"); ofstream g("date.out"); int v[10]; int main() {     int x,i;     while(f&gt;&gt;x)     {         int y=x%10;         while(x&gt;9)             x/=10;         if(x==y)             v[x]++;     }     for(i=1;i&lt;=9;i++)         g&lt;&lt;v[i]&lt;&lt;" ";     g.close();     f.close(); }</pre></td></tr></table>	Limbajul Pascal	Limbajul C/C++	<pre>var     f,g:text;     v:array[1..10] of integer;     i,x,y:integer; begin     assign(f,'date.in');     reset(f);     assign(g,'date.out');     rewrite(g);     while not eof(f) do         begin             read(f,x);             y:=x mod 10;             while(x&gt;9) do                 x:=x div 10;             if(x=y) then                 inc(v[x]);             end;             for i:=1 to 9 do                 write(g, v[i], ' ');             close(f);             close(g);             readln;         end.</pre>	<pre>#include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("date.in"); FILE *f=fopen("date.in" ,"r"); ofstream g("date.out"); int v[10]; int main() {     int x,i;     while(f&gt;&gt;x)     {         int y=x%10;         while(x&gt;9)             x/=10;         if(x==y)             v[x]++;     }     for(i=1;i&lt;=9;i++)         g&lt;&lt;v[i]&lt;&lt;" ";     g.close();     f.close(); }</pre>	
Limbajul Pascal	Limbajul C/C++					
<pre>var     f,g:text;     v:array[1..10] of integer;     i,x,y:integer; begin     assign(f,'date.in');     reset(f);     assign(g,'date.out');     rewrite(g);     while not eof(f) do         begin             read(f,x);             y:=x mod 10;             while(x&gt;9) do                 x:=x div 10;             if(x=y) then                 inc(v[x]);             end;             for i:=1 to 9 do                 write(g, v[i], ' ');             close(f);             close(g);             readln;         end.</pre>	<pre>#include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("date.in"); FILE *f=fopen("date.in" ,"r"); ofstream g("date.out"); int v[10]; int main() {     int x,i;     while(f&gt;&gt;x)     {         int y=x%10;         while(x&gt;9)             x/=10;         if(x==y)             v[x]++;     }     for(i=1;i&lt;=9;i++)         g&lt;&lt;v[i]&lt;&lt;" ";     g.close();     f.close(); }</pre>					
	<p>b) Se parcurge fișierul <b>date.in</b> și se reține într-un tablou unidimensional (vectorul de frecvențe) numărul elementelor care au prima și ultima cifră egale. Prin urmare <b>v[1]</b> reține numărul elementelor care încep și se termină cu 1, <b>v[2]</b> reține numărul elementelor care încep și se termină cu 2 și așa mai departe. Se scrie în fișierul <b>date.out</b> conținutul tabloului unidimensional. Din punct de vedere al eficienței de timp, avem o parcurgere liniară a șirului deci <b>O(n)</b>. Pentru o descriere coerentă a metodei se acordă <b>1p.</b>, iar pentru justificarea eficienței, încă <b>1p.</b> - în total <b>2p.</b></p>					