

## VARIANTA 3 - Rezolvare

### Rezolvare - Subiectul 1

1.	<p>Valoarea variabilei <math>x</math> trebuie să satisfacă condiția <math>100 \leq  x  \leq 999</math>.</p> <p>Pentru precizarea răspunsului d) se acordă <b>4p</b>.</p>
2.	<p>În desenul alăturat este prezentat mecanismul prin care se execută apelul <math>f(12345)</math> al subprogramului recursiv <math>f</math>. Se vor afișa, în ordine: <math>5432\#\star\star\star\star</math>, răspuns pentru care se acordă <b>4p</b>.</p> <pre> graph TD     f12345["f(12345)"] --&gt; f1234["f(1234)"]     f1234 --&gt; f123["f(123)"]     f123 --&gt; f12["f(12)"]     f12 --&gt; f1["f(1)"]     f1 --&gt; ret1["#"]     f12345 --&gt; ret5["5"]     f1234 --&gt; ret4["4"]     f123 --&gt; ret3["3"]     f12 --&gt; ret2["2"]     f1 --&gt; ret_star1["★"]     f12345 --&gt; ret_star2["★"]     f1234 --&gt; ret_star3["★"]     f123 --&gt; ret_star4["★"]     f12 --&gt; ret_star5["★"]     f1 --&gt; ret_star6["★"] </pre>
3.	Răspuns corect <b>a</b> . pentru care se acordă <b>4p</b> .
4.	Pentru răspunsul <b>d</b> ) se acordă <b>4p</b> .
5.	<p>Cele mai lungi lanțuri elementare din arbore cu o extremitate în nodul 8 sunt: <math>[2, 1, 4, 6, 7, 8]</math> și <math>[3, 1, 4, 6, 7, 8]</math>.</p> <pre> graph TD     1((1)) --- 2((2))     1 --- 3((3))     1 --- 4((4))     4 --- 5((5))     4 --- 6((6))     6 --- 7((7))     7 --- 8((8)) </pre> <p>Răspuns corect <b>a</b>. pentru care se acordă <b>4p</b>.</p>

## Rezolvare - Subiectul 2

1.	a)	<p>Șirul de date de intrare (fără valoarea 0 utilizată pentru încheierea operației de citire) se împarte în secvențe formate din numere cu proprietatea că prima cifră a numărului curent este egală cu cifra unităților numărului precedent, iar cifra unităților numărului curent este egală cu prima cifră a numărului următor. Astfel șirul de date de intrare se împarte în trei secvențe de acest tip cu lungimile 3,1,4:</p> <p>12 23 34 592 13 39 91 1 0</p> <p>Pentru precizarea valorii 314 se acordă <b>6p.</b></p>		
	b)	<p>Un răspuns posibil poate fi setul de date de intrare: 1 2 2 3 3 3 4 4 4 4 0</p> <p>Pentru orice set de date de intrare corect se acordă <b>6p.</b></p>		
	c)	<p>Înlocuirea structurii <b>cât timp</b> <math>t &gt; 0</math> <b>execută</b> ... presupune utilizarea unei structuri repetitive <b>repetă</b> - <b>cât timp</b> sau <b>repetă</b> - <b>până când</b>:</p> <table border="1"><tr><td><pre>repetă   t ← [t/10] cât timp t &gt; 9</pre></td><td><pre>repetă   t ← [t/10] până când t ≤ 9</pre></td></tr></table> <p>Pentru alegerea unei structuri repetitive se acordă 2p., pentru testarea condiției de continuare sau oprire se acordă 2p, iar pentru scrierea integrală a restului algoritmului se mai acordă 2p. - în total <b>6p.</b></p>	<pre>repetă   t ← [t/10] cât timp t &gt; 9</pre>	<pre>repetă   t ← [t/10] până când t ≤ 9</pre>
<pre>repetă   t ← [t/10] cât timp t &gt; 9</pre>	<pre>repetă   t ← [t/10] până când t ≤ 9</pre>			
	d)	<p>Se testează capacitatea de a reprezenta pe hârtie algoritmul dat cu ajutorul unui limbaj de programare studiat. Deși scrierea programelor pe hârtie este o activitate improprie la informatică, prin această cerință se valorifică experiențele anterioare de implementare și testare a programelor pe calculator, privind structura programelor, declararea variabilelor, sintaxa instrucțiunilor programului, regulile de scriere a expresiilor, etc.</p> <p>Pentru instrucțiunile corecte de declarare a variabilelor, de citire a datelor, de afișare a rezultatului se acordă câte 1p, pentru instrucțiunea de decizie decizie se acordă 1p., pentru instrucțiunea repetitivă se acordă 3p., iar pentru cele 6 atribuiri se acordă 2p. Pentru structura corectă a programului se mai acordă 1p., în total <b>10p.</b></p>		
2.		<p>Numerele cerute sunt de forma: <math>32x \times 3</math> cu <math>x \in \{1,2,3,4\}</math>. Sunt generate <math>4 \times 4 = 16</math> numere cu proprietatea din enunț.</p> <p>Pentru răspunsul corect se acordă <b>6p.</b></p>		
3.		<p>Șirului de caractere S i se atribuie valoarea <math>3a+b</math>. Din acesta se șterge caracterul de pe prima poziție și noul șir va fi <math>a+b=9</math>. Instrucțiunea următoare șterge caracterul al doilea și al treilea. Rezultatul afișat este <math>a=9</math>.</p> <p>Pentru răspunsul corect se acordă <b>6p.</b></p>		

## Rezolvare - Subiectul 3

1. Prelucrarea oferită de subprogram determină fiecare apariție a valorii  $x$  în tablou. Dublarea unui element cu valoarea  $x$  se realizează prin deplasarea cu o poziție la dreapta a tuturor elementelor situate la dreapta celui ce urmează a fi dublat, începând de la sfârșitul șirului. După realizarea deplasării, se mărește valoarea lui  $n$  cu 1.

### Limbajul Pascal

```
Type vector=array[1..100]of integer;
procedure sub(x:integer;var n:byte;var
v:vector);
var i,j:byte;
begin
    i:=1;
    while i<=n do begin
        if v[i]=x then begin
            for j:=n downto 1 do
                v[i+1]:=v[j];
            n:=n+1;
            i:=i+1;
        end;
        i:=i+1;
    end;
end;
```

### Limbajul C/C++

```
void sub(int x,int &n,int v[])
{
    int i=1,j;
    while(i<=n)
    {
        if(v[i]==x) {
            for(j=n;j>=1;j--)
                v[j+1]=v[j];
            n++;
            i++;
        }
        i++;
    }
}
```

**Pentru** antet corect se acordă 3p. (Pentru respectarea structurii antetului (**procedure/void**) se acordă 1p., pentru declararea corectă a parametrilor de intrare  $x$ , pentru declararea corectă a parametrilor  $n$  și  $v$  încă 1p., pentru declararea variabilelor locale încă 1p).

Pentru determinarea fiecărei apariții ale numărului  $x$  în tabloul 1p. Pentru dublarea primei apariții în tabloului și incrementarea valorii parametrului  $n$  se acordă câte 1p. Pentru determinarea tuturor aparițiilor lui  $x$  în tablou se acordă 2p. Pentru mărirea corespunzătoare a valorii parametrului  $n$  se acordă 1p. Pentru corectitudinea globală a programului 1p.

În total **10p.**

2. Pot fi concepute mai multe metode de obținere a matricei cerute. Metoda clasică ce completează coloanele matricei, cu liniile, respectiv coloanele, numerotate de la 1 la  $n$ , respectiv de la 1 la  $m$ , în maniera descrisă, este:

```
pentru i<=1,n execută
    pentru j<=1,m execută
        dacă i=1 atunci A[i,j]←j
        altfel dacă j=1 atunci A[i,j]←i
        altfel A[i,j]←A[i,j-1]+A[i-1][j]
```

Pentru declararea corectă a matricei se acordă 1p., pentru citirea lui  $n$  și  $m$  se mai acordă 1p., pentru completarea corectă a primei linii se acordă 1p., pentru completarea corectă a primei coloane se acordă 1p., pentru completarea corectă a celorlalte elemente se acordă 3p., iar pentru afișarea matricei conform cerinței se acordă 2p. În plus, pentru declararea variabilelor simple, structura și corectitudinea sintactică a programului se mai acordă 1p. - în total **10p.**

3.	<p><b>b. Algoritmul de rezolvare propus</b> presupune determinarea poziției pp a primului număr format doar din cifre pare, respectiv ui a ultimului număr format doar din cifre impare, din fișier pe măsura citirii datelor și utilizează doar variabile simple fără folosirea unor structuri suplimentare de date.</p> <p>Eficiența algoritmului, ca timp de executare, constă în faptul că parcurgem o singură dată șirul cu un număr impresionant de numere. Deoarece numărul de valori din fișier (1000000) este dimensiunea care contează în problemă, spunem că am obținut un algoritm liniar. Ca spațiu de memorie, soluția propusă este eficientă, deoarece utilizează doar variabile simple.</p> <p>Lungimea subșirului căutat va fi: <b>ui-pp+1</b> sau <b>0</b> dacă <b>pp=0</b> sau <b>ui=0</b>.</p> <p>Pentru o descriere coerentă a metodei se acordă 1p., iar pentru justificarea eficienței, încă 1p. - în total <b>2p</b>.</p> <p><b>a.</b></p> <table border="1" data-bbox="321 615 1383 1329"> <thead> <tr> <th data-bbox="321 615 857 684">Limbaajul Pascal</th><th data-bbox="857 615 1383 684">Limbaajul C/C++</th></tr> </thead> <tbody> <tr> <td data-bbox="321 684 857 1329"> <pre> var f:text;     nr, i,pp, ui,p, imp:longint; begin     assign(f,'bac.in'); reset(f);     while not(eof(f)) do begin         read(f,nr);         inc(i);         p:=0;         imp:=0;         repeat             if nr mod 2=0 then inc(p)             else inc(imp);             nr:=nr div 10         until nr=0;         if (imp=0) and (pp=0)             then pp=i             else if (p==0) ui:=i;         end;         if pp*up=0 then writeln ('0')         else writeln(ui-pp+1);         close(f);     end.</pre> </td><td data-bbox="857 684 1383 1329"> <pre> #include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("bac.in"); int main() {     int pp=0, ui=0, i=0, nr, p, imp;     while(f&gt;&gt;nr)     {         p=imp=0;         i++;         do         {             if(nr%2==0) p++;             else imp++;             nr/=10;         }while(nr&gt;0);         if(imp==0 &amp;&amp; pp=0)pp=i;         else if (p==0) ui=i;     }     if(pp*ui) cout&lt;&lt;"Nu exista";     else cout&lt;&lt;ui-pp+1;     f.close(); }</pre> </td></tr> </tbody> </table> <p>Se acordă 1p. pentru operațiile cu fișiere (declarare, nume corect și deschidere pentru citire), încă 1p. pentru citirea tuturor numerelor din fișier, 1p. pentru un algoritm principal corect, încă 1p. pentru determinarea valorii cerute, 1p. pentru afișarea ei, 1p. pentru corectitudinea formală (declararea variabilelor, structura programului, sintaxa instrucțiunilor, etc.).</p> <p>Se acordă 1p. pentru alegerea unui algoritm eficient ca timp de executare (<math>O(n)</math>) și 1p. pentru utilizarea eficientă a memoriei. În total <b>8p</b>.</p>	Limbaajul Pascal	Limbaajul C/C++	<pre> var f:text;     nr, i,pp, ui,p, imp:longint; begin     assign(f,'bac.in'); reset(f);     while not(eof(f)) do begin         read(f,nr);         inc(i);         p:=0;         imp:=0;         repeat             if nr mod 2=0 then inc(p)             else inc(imp);             nr:=nr div 10         until nr=0;         if (imp=0) and (pp=0)             then pp=i             else if (p==0) ui:=i;         end;         if pp*up=0 then writeln ('0')         else writeln(ui-pp+1);         close(f);     end.</pre>	<pre> #include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("bac.in"); int main() {     int pp=0, ui=0, i=0, nr, p, imp;     while(f&gt;&gt;nr)     {         p=imp=0;         i++;         do         {             if(nr%2==0) p++;             else imp++;             nr/=10;         }while(nr&gt;0);         if(imp==0 &amp;&amp; pp=0)pp=i;         else if (p==0) ui=i;     }     if(pp*ui) cout&lt;&lt;"Nu exista";     else cout&lt;&lt;ui-pp+1;     f.close(); }</pre>
Limbaajul Pascal	Limbaajul C/C++				
<pre> var f:text;     nr, i,pp, ui,p, imp:longint; begin     assign(f,'bac.in'); reset(f);     while not(eof(f)) do begin         read(f,nr);         inc(i);         p:=0;         imp:=0;         repeat             if nr mod 2=0 then inc(p)             else inc(imp);             nr:=nr div 10         until nr=0;         if (imp=0) and (pp=0)             then pp=i             else if (p==0) ui:=i;         end;         if pp*up=0 then writeln ('0')         else writeln(ui-pp+1);         close(f);     end.</pre>	<pre> #include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std; ifstream f("bac.in"); int main() {     int pp=0, ui=0, i=0, nr, p, imp;     while(f&gt;&gt;nr)     {         p=imp=0;         i++;         do         {             if(nr%2==0) p++;             else imp++;             nr/=10;         }while(nr&gt;0);         if(imp==0 &amp;&amp; pp=0)pp=i;         else if (p==0) ui=i;     }     if(pp*ui) cout&lt;&lt;"Nu exista";     else cout&lt;&lt;ui-pp+1;     f.close(); }</pre>				