

## Exercitii laborator

Se da un numar intreg  $n$ , cuprins intre 1000 si 99999. Sa se verifice daca suma cifrelor lui  $n$  este un numar prim. Programul va intoarce un string astfel:

- daca  $n$  nu este in intervalul mentionat, se va afisa un mesaj de eroare; "Numarul nu este valid"
- DA/NU daca respecta cerinta sau nu.

### 1. Partitionarea in clase de echivalenta

Intrarile programului:  $n$

Clasele intrarilor:

$N_1 = \{n \mid n \text{ intreg}, n \geq 1000, n \leq 99999\}$  ---- cazurile pentru  $n$  valid

$N_2 = \{n \mid n < 1000\}$

$N_3 = \{n > 99999\}$

Clase iesiri

$I_1 = \text{"DA"}$

$I_2 = \text{"NU"}$

$I_3 = \text{"Numarul nu este valid"}$

Clase de echivalenta finale:

$C1 = \{n \text{ in } N_1, \text{iesirea } I_1\} \text{ — } n = 5413$

$C2 = \{n \text{ in } N_1, \text{iesirea } I_2\} \text{ — } n = 8200$

$C3 = \{n \text{ din } N_2, I_3\} \text{ — } 29$

$C4 = \{n \text{ din } N_3, I_3\} \text{ — } 1234567$

Teste:

N	iesirea
5413	DA
8200	NU
29	"Numarul nu este valid"

1234567	"Numarul nu este valid"
---------	-------------------------

## 2. Analiza valorilor de frontiera

Intrari: n

Frontier

e pt n: 999, 1000, 99999, 100000

cazuri speciale: 1001 (suma cifrelor 2)

Teste:

(999, "eroare"), (1000, "NU"), (99999, "NU"), (100000, "eroare"), (1001, "da")  
7777

## 3. Graful cauza efect

Cauze:

C1 = n valid, n in intervalul 1000 - 99999

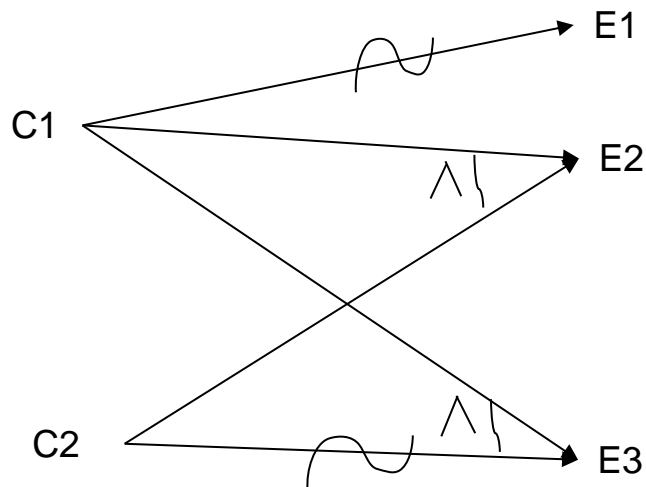
C2 = n verifica enuntul, suma cifrelor sa fie nr prim.

Efecte:

E1: mesaj eroare

E2: da

E3: nu



Definim tabelul de decizie, cu 5 linii (nr cauze + nr efecte):

Parcurgem efectele:

E1: mesaj de eroare:  $\sim C1 = 1 \Rightarrow C1 = 0$

Nu  
ne intereseaza valoarea lui C2 deci consideram ca este 0  
C1 = 0 si C2 = 0

C1	0
C2	0
E1	1
E2	0
E3	0

E2:  $C1 \wedge C2 = 1 \Rightarrow C1 = 1, C2 = 1$

C1	0	1
C2	0	1
E1	1	0
E2	0	1
E3	0	0

E3:  $C1 \wedge \sim C2 = 1 \Rightarrow C1 = 1, C2 = 0$

C1	0	1	1
C2	0	1	0
E1	1	0	0
E2	0	1	0
E3	0	0	1

**Tabelul final:**

<b>C1</b>	0	1	1
<b>C2</b>	0	1	0

<b>E1</b>	1	0	0
<b>E2</b>	0	1	0
<b>E3</b>	0	0	1
<b>Input</b>	<b>687</b>	<b>5321</b>	<b>5733</b>
<b>Output</b>	<b>Eroare</b>	<b>DA</b>	<b>NU</b>

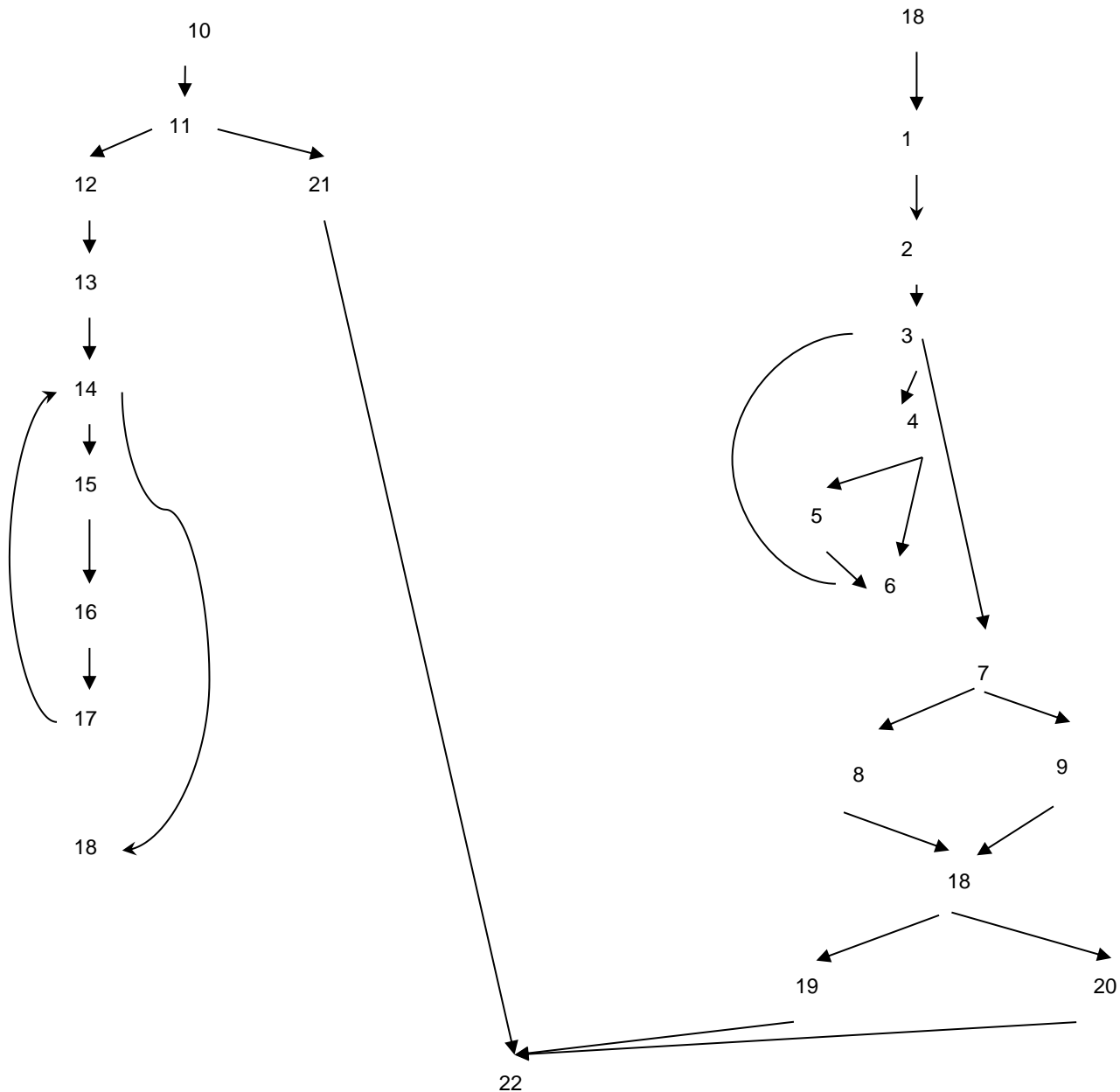
## Testare structurala

Graful orientat al programului:

```

1 static Boolean isPrim(Integer n){
2     Integer nr = 0 ;
3     for(int i = 2;i<n/2;i++){
4         if(n%i==0){
5             nr++;
6         }
7     }
8     if(nr == 0 && n >= 2)
9         return true;
10    else return false;
11    }
12
13 static String calculeaza(Integer n){
14     if(n >=1000 && n <=99999){
15         Integer nr = n, sum = 0;
16         int c;
17         while(nr>0){
18             c = nr % 10;
19             nr = nr / 10;
20             sum = sum + c;
21         }
22         if(isPrim(sum))
23             return "DA";
24         else return "NU";
25     }else{
26         return "Numarul nu este valid";
27     }
28 }

```



**if(nr == 0 && n >= 2)**

Conditii individuale: **C1** = nr == 0 si **C2** = n >= 2

Decizia **D** = (nr == 0 && n >= 2) = **C1** ^ **C2**

- Acoperire la nivel de instructiune D = true, D = false
- Acoperire la nivel de decizie D = true, D = false (acelasi lucru cu acoperirea anterioara)

D = true : C1 = true, C2 = true

D = false: C1 = true, C2 = false

- Acoperire la nivel de conditie: C1 = true/false si C2 = true/false

$C1 = \text{true}$  si  $C2 = \text{false} \Rightarrow D = \text{false}$

$C1 = \text{false}$  si  $C2 = \text{true} \Rightarrow D = \text{false}$

- Acoperire la nivel de decizie/conditie  $C1 = \text{true/false}$  si  $C2 = \text{true/false}$ ,  $D = \text{true/false}$

$D = \text{true}$ :  $C1 = \text{true}$ ,  $C2 = \text{true}$

$D = \text{false}$ :  $C1 = \text{false}$ ,  $C2 = \text{false}$

- Acoperire la nivel de conditii multiple: trebuie sa luam toate combinatiile posibile

$D = \text{true}$ :  $C1 = \text{true}$ ,  $C2 = \text{true}$

$D = \text{false}$ :  $C1 = \text{false}$ ,  $C2 = \text{false}$

$D = \text{false}$ :  $C1 = \text{true}$ ,  $C2 = \text{false}$

$D = \text{false}$ :  $C1 = \text{false}$ ,  $C2 = \text{true}$

- **Acoperire la nivel de conditie/decizie modificata:** fiecare conditie individuala sa fie atat true cat si false:  $C1 = \text{true/false}$ ,  $C2 = \text{true/false}$ ; fiecare decizie este true/false:  $D = \text{true/false}$  si **fiecare conditie individuala influenteaza independent decizia.**

$D = \text{true}$ :  $C1 = \text{true}$ ,  $C2 = \text{true} \Rightarrow 1231$

$D = \text{false}$ :  $C1 = \text{true}$ ,  $C2 = \text{false} \Rightarrow 1000$

$D = \text{false}$ :  $C1 = \text{false}$ ,  $C2 = \text{true} \Rightarrow 3450$