

**Se consideră secvențele de cod de mai jos.****I. Metoda isPrime verifică dacă un număr natural n dat este prim.**

	/** * @param n natural number * @return true, false * @exception InvalidValueException if n < 0 */
	<b>private boolean</b> isPrime( <b>int</b> n) <b>throws</b> InvalidValueException{
	<b>boolean</b> result = <b>true</b> ;
	<b>if</b> (n<0)
	<b>throw new</b> InvalidValueException("negative value was provided!");
	<b>else if</b> ((n == 0)    (n == 1))
	result = <b>false</b> ;
	<b>else</b> {
	<b>int</b> d = 2;
	<b>while</b> (d<=Math.ceil(n/2)){
	<b>if</b> (n%d == 0)
	result = <b>false</b> ;
	d++;
	}
	}
	<b>return</b> result;
	}

**II. Metoda computeMaxCounter determină frecvența de apariție a valorii maxime dintr-o listă de elemente date.**

	/** * @param list - lista de valori intregi * @return frecventa de aparitie a valorii maxime */
	<b>public int</b> computeMaxCounter(List<Integer> list){
	<b>int</b> countMax, index, posMax;
	index=0;countMax=0;posMax=0;
	<b>while</b> (index<list.size()){
	<b>if</b> (list.get(index) > list.get(posMax)){
	posMax=index;
	countMax=1;
	}
	<b>else</b>
	<b>if</b> (list.get(posMax)== list.get(index))
	countMax++;
	index++;
	}
	<b>if</b> (list.size()==0)
	<b>return</b> -1;
	<b>else return</b> countMax;
	}

**I. Se cere:**

- a. elaborarea CFG;
- b. identificarea drumurilor independente;
- c. calculul CC folosind cele 3 formule.
- d. proiectarea de cazuri de testare pentru acoperirea tuturor:
  - drumurilor;
  - instrucțiunilor;
  - deciziilor, condițiilor, deciziilor și condițiilor, condițiilor multiple;
  - buclelor.

## CFG. Construire

- pași de elaborare a unui CFG:
  1. se numerează unic fiecare element de structură secvențială (calcul) și condițională (decizie);
  2. se începe pornind de la vârful de intrare, care are (de obicei) numărul 1;
  3. se adaugă celelalte vârfuri corespunzătoare structurilor numerotate și se unesc prin arce, evidențiind transmiterea controlului în cadrul programului;
  4. la final, toate ieșirile posibile din program se unesc în vârful de ieșire;

## Complexitatea ciclomatică. Definiție

- **complexitatea ciclomatică** (*engl. McCabe's cyclomatic complexity, CC*):
  - *metrică software* aplicată pentru măsurarea cantitativă a complexității logice a unui program;
  - *permite determinarea numărului de drumuri independente din mulțimea de bază a unui CFG*;
- modalități de calcul a CC la nivelul CFG:
  - $CC = \text{numărul de regiuni din CFG}$ ;
  - $CC = E - N + 2$ , unde  $E$  - #arce,  $N$  - #vârfuri ;
  - $CC = P + 1$ , unde  $P$  - #vârfuri condiție.
- **regiune**:
  - zonă a CFG marginită parțial sau în totalitate de arce și vârfuri;