EVALUARE

1. **Cum se numește șirul: 1, 4, 9, 16, 25, 36,...?**

   a) Șirul numerelor impare.

   b) Șirul numerelor triunghiulare.

   c) Șirul pătratelor perfecte nenule.

2. **Ce proprietate are șirul lui Fibonacci?**

   a) Orice element este suma pătratelor perfecte anterioare.

   b) Fiecare element este suma celor două elemente precedente.

   c) Fiecare element este divizibil cu 3.

3. **Ce reprezintă șirul 1, 3, 6, 10, 15, 21, 28,...?**

   a) Șirul numerelor triunghiulare.

   b) Șirul numerelor naturale.

   c) Șirul numerelor pare.

4. **Cum poate fi definit un șir de valori?**

   a) O serie de numere generate aleator.

   b) O serie de numere care respectă o regulă sau o proprietate.

   c) O serie de numere fără o ordine specifică.

5. Ce este un tablou unidimensional?

a) O mulțime de valori de tipuri diferite, organizate în funcție de indice

b) O mulțime de valori de același tip, organizate în funcție de poziția lor în cadrul tabloului

c) O mulțime de numere întregi fără un indice atribuit

6. Ce reprezintă dimensiunea unui tablou unidimensional?

a) Numărul maxim de valori pe care le poate conține

b) Numărul de caractere din fiecare element al tabloului

c) Numărul minim de valori din tablou

7. În sintaxa C++, cum declarăm un tablou unidimensional de numere întregi cu 25 de elemente?

a) int V[25];

b) int V(25);

c) int V=25;

LIVEWORKSHEETS

8. Ce se întâmplă dacă declarăm un tablou cu mai multe elemente decât este necesar?

a) Se va afișa o eroare

b) Se va aloca memorie în plus, dar nu va fi utilizată

c) Compilatorul va elimina automat elementele suplimentare

9. Care este sintaxa pentru inițializarea unui tablou unidimensional în C++ cu valorile {3, 2, 4}?

a) int v[3]={3,2,4};

b) int v[3](3,2,4);

c) int v[3]={3;2;4};

10. Care este sintaxa pentru a accesa elementul cu indicele 5 al tabloului numit v?

a) v.5

b) v[5]

c) v{5}

11. Cum se afișează elementele unui tablou de la stânga la dreapta?

a) for (i=0; i<n-1; i++) cout<<v[i];

b) for (i=0; i=n; i++) cout<<v[i];

c) for (i=n-1; i>=0; i--) cout<<v[i];

12. Un program care prelucrează valorile unui șir de 20 de numere reale citește aceste numere într-o variabilă *z* . Care este declararea corectă a acestei variabile?

a) float z

b) int z[20]

c) float z[20]

13. Ce afișează secvența următoare:

```
int a[] = {14, 3, 7, 0, -4, 3, 10, 15, 7};
int s = 0;
for (int i = 0; i < 9; i++)
        if (i % 2 == 0) s=s+ a[i];
cout << s;
```

a) 0; b) 34; c) 21; d) 50

LIVEWORKSHEETS

14. Pentru n=5 şi valorile următoare: 5, 3, 4, 6,7 spuneţi ce vor afişa programele următoare:

1	int v[100], n, i, p=1;
	cout<<"n="; cin>>n;
	for (i=0; i<n; i++) {
	cin>>v[i];
	if(v[i]%3==0) p=p*v[i]; }
	cout<<p;
		a) 3
		b) 9
		c) 18
		d) 12

2	int v[100], n, i, s=0;
	cout<<"n="; cin>>n;
	for (i=0; i<n; i++) {
	cin>>v[i];
	s=s+v[i]; }
	cout<<s;
		a) 25
		b) 8
		c) 12
		d) 18

15. În secvenţa alăturată se citeşte un vector cu n elemente.

```
int v[100], i, n;
cout<<"Lungimea vectorului n=";
cin>>n;
for (i=0; i<=n-1; i++)
{
  cout<<"v["<<i<<"]=";
  cin>>v[i];
}
```

Realizaţi corespondenţa pentru secvenţele de program din stânga cu afirmaţiile din dreapta.

eNr=0;
for (i=0; i<=n-1; i++)
if (v[i]%2==0) eNr=1;
 	
if (eNr==1) cout<<"Da. Vectorul contine numere pare";
else cout<<"Nu. Vectorul NU contine numere pare ";
	Afişează doar elementele unui şir care sunt divizibile cu 2 şi 3.

for(i=0;i<=n-1;i++){
if(v[i]%2==0 && v[i]%3==0)
cout<<v[i]<<" "; }
	Afişează poziţia elementelor care sunt divizibile cu 2.

for(i=0;i<=n-1;i++) {
if(v[i]%2==0)
cout<<i<<" "; }
	Afişează un nou vector în care au fost înlocuite elementele care aveau valoarea 3 cu valoarea 0.

for(i=0;i<=n-1;i++) {
if(v[i]%2==0)
cout<<v[i]<<" "; }
	Verifică dacă într-un vector de n numere întregi pozitive, există numere pare şi afişează un mesaj.

for(i=0;i<=n-1;i++){
if(v[i]%2==0 || v[i]%3==0)
cout<<v[i]<<" "; }
	Afişează doar elementele din vector care sunt divizibile cu 2.

for (i=0; i<=n-1; i++)
{
if(v[i]==3) v[i]=0;
}
for (i=0; i<=n-1; i++)
cout<<v[i]<<" ";
	Afişează doar elementele unui şir care sunt divizibile cu 2 sau 3.

LIVEWORKSHEETS