

Analiza unității de învățare: Subprograme recursive

1. Încadrarea unității în programa școlară:

Se studiază în clasa a X -a, la specializarea Matematică-Informatică, intensiv Informatică, la disciplina Informatică

2. Competențe generale care corespund temei alese:

Identificarea datelor care intervin într-o problemă și a relațiilor dintre acestea
Implementarea algoritmilor într-un limbaj de programare

3. Matricea de asociere dintre conținuturi și competențe specifice *

	1.1	1.2	1.3	2.1	2.2	2.3	2.4	3.1	3.2	3.3	3.4	4.1	4.2	5.1	5.2
Definiția procesului recursiv	X		O	X			X	X	X	O	X		O	O	O
Reguli pentru construirea unui subprogram recursiv	O		X	X	X	X	X	X	O	X	X	O	O	O	O
Variabilele locale si subprogramele recursive	X	X	X	X	O		X	X							
Implementarea recursiva a algoritmilor elementari: determinarea valorii minime (maxime), c.m.m.d.c, prelucrarea cifrelor unui numar, testarea unui numar prim, determinarea divizorilor unui numar, conversia intre baze de numeratie	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O
Implementarea recursiva a algoritmilor pentru prelucrarea tablourilor de memorie	X	X	X	X	X	X	X	X	X	X	X	X	X	O	O
Recursivitatea in cascada	X	X	X	X	X	X	X	X	O	X	X	X	X		
Recursivitatea directa si indirecta	X	X	X	X	X	X	X	X	O	X	X	X	X		
Avantajele si dezavantajele recursivitatii		O	X	X	X	X	X	X	X	X	O		X	X	O
Evaluarea			O	X	X	X	X	X	X	O	X	X	X		

Notă. În celule se pune X -atunci când există o corelație puternică între conținuturi și competența specifică; se pune O - atunci când există o corelație mai slabă între conținuturi și competența specifică; se lasă celula goală atunci când nu există nici o corelație între conținuturi și competența specifică.

*** Competentele specifice:**

- 1.1. Evidențierea necesității structurării datelor
- 1.2. Prelucrarea datelor structurate
- 1.3. Alegerea structurii de date adecvate rezolvării unei probleme
- 2.1. Utilizarea corectă a subprogramelelor predefinite și a celor definite de utilizator
- 2.2. Construirea unor subprograme pentru rezolvarea subproblemelor unei probleme
- 2.3. Aplicarea mecanismului recursivității prin crearea unor subprograme recursive (definite de utilizator)
- 2.4. Compararea dintre implementarea recursivă și cea iterativă a aceluiași algoritm
- 3.1. Prelucrarea datelor structurate
- 3.2. Recunoașterea situațiilor în care este necesară utilizarea unor subprograme
- 3.3. Analiza problemei în scopul identificării subproblemelor acesteia
- 3.4. Descrierea metodei de rezolvare a unei probleme în termeni recursivi
- 4.1. Elaborarea unui algoritm de rezolvare a unor probleme din aria curriculară a specializării
- 4.2. Alegerea unui algoritm eficient de rezolvare a unei probleme
- 5.1. Identificarea aplicațiilor informaticii în viața socială
- 5.2. Elaborarea și implementarea unor algoritmi de rezolvare a unor probleme cotidiene

4. Activitati de invatare

- Explicarea conceptelor referitoare la subprogramele recursive
- Descompunerea rezolvării unei probleme în subprobleme
- Identificarea unor situații în care alegerea unui subprogram recursiv prezintă avantaje în raport cu altul iterativ
- Exersarea definirii și apelării unor subprograme recursive simple
- Proiectarea/modelarea unor algoritmi și implementarea acestora cu ajutorul subprogramelelor recursive
- Prezentarea tehnicilor de utilizare a tablourilor în cadrul subprogramelelor recursive
- Exersarea creării și aplicării subprogramelelor recursive pentru rezolvarea unor probleme întâlnite de elevi în studiul altor discipline școlare
- Evidențierea greselilor tipice în elaborarea algoritmilor recursivi

5. Tipuri de exercitii si probleme propuse

Tipuri de probleme:

- - Executarea pas cu pas a unui program ce folosește subprograme recursive
- - Exemplificarea modului în care este folosită stiva sistemului la apelarea unui subprogram recursiv
- - Exemplificarea modului în care pot fi transmiși parametrii între subprogramele recursive
- - Exemplificarea modului în care pot fi implementate subprogramele recursive

- - Elaborarea de subprograme recursive simple
- - Exemplificarea modului în care poate fi implementat un subprogram recursiv cu număr variabil de parametrii

Probleme propuse:

1. Executați programul instrucțiune cu instrucțiune, folosind tasta F7.
2. Scrieti o functie recursiva care calculeaza suma cifrelor unui numar natural n.
3. Să se scrie o **funcție recursivă** care returnează factorialul unui număr dat ca parametru.
4. Să se scrie o **funcție recursivă** cu trei parametri **n**, **k**, **c** și întoarce prin parametrul **c** numărul de cifre ale lui **n** care sunt mai mari sau egale decât **k**.