



# Piscina C

Ziua 04

Staff 42 [piscina@academyplus.ro](mailto:piscina@academyplus.ro)

*Sumar: Acest document este subiectul zilei a 04 a piscinei C din cadrul Academy+Plus.*

# Cuprins

|      |                                       |    |
|------|---------------------------------------|----|
| I    | Instructiuni                          | 2  |
| II   | Preambul                              | 4  |
| III  | Exercitiu 00 : ft_iterative_factorial | 5  |
| IV   | Exercitiu 01 : ft_recursive_factorial | 6  |
| V    | Exercitiu 02 : ft_iterative_power     | 7  |
| VI   | Exercitiu 03 : ft_recursive_power     | 8  |
| VII  | Exercitiu 04 : ft_fibonacci           | 9  |
| VIII | Exercitiu 05 : ft_sqrt                | 10 |
| IX   | Exercitiu 06 : ft_is_prime            | 11 |
| X    | Exercitiu 07 : ft_find_next_prime     | 12 |
| XI   | Exercitiu 08 : Cele opt regine        | 13 |
| XII  | Exercitiu 09 : Cele opt regine 2      | 14 |

# Capitolul I

## Instructiuni

- Utilizati doar aceste pagini ca referinta, nu plecati urechea la zgomotul de pe coridor.
- Subiectul se poate schimba cu cel mult o ora inainte de incepere.
- Fiti atenti la drepturile pe care le aveti asupra fisierelor si directoarelor.
- Trebuie sa urmati procedurile de parcurgere pentru toate exercitiile voastre.
- Exerciitiile voastre vor fi corectate de colegii de piscina.
- Pe langa colegi, veti fi corectati de un program numit Moulinette.
- Aplicatia Moulinette este foarte stricta in notare. Ea este total automatizata. Este imposibil sa comentati in legatura cu nota primita. Fiti foarte rigurosi pentru a evita surprizele.
- Moulinette nu e foarte desteapta. Ea nu poate intelege codul care nu respecta Standardele de scriere a codului (Norme).
- Utilizarea unei functii interzise este un caz de inselaciune (trisare). Toate aceste cazuri sunt sanctionate cu nota -42.
- Daca `ft_putchar()` este o functie valida, veti compila fisierul `ft_putchar.c`.
- Nu trebuie sa creati o functie `main()` decat atunci cand vi se cere sa scrieti un program.
- Exerciitiile sunt strict ordonate de la cele simple spre cele complexe. In nici un caz nu vom lua in considerare un exercitiu complex rezolvat, daca unul anterior, mai simplu, nu a fost rezolvat perfect.
- Aplicatia Moulinette se compileaza cu flag-urile: `-Wall -Wextra -Werror`.
- Daca programul vostru nu se compileaza, veti primi nota 0.

- Nu lasati in directorul de lucru niciun fisier, altul decat cele specificate de enuntul exercitiului.
- Aveti intrebari? Intrebat-l pe vecinul din dreapta. Daca nu, incercati la cel din stanga.
- Manualele voastre de referinta sunt Google / man / Internet / ....
- Puteti folosi forumul de pe Intranet pentru discutii legate de Piscina!
- Cititi cu atentie exemplele. Va pot oferi informatii suplimentare pentru elementele neclare din enunt...
- Reflectati la asta. Aveti mare grija!

# Capitolul II


## Preambul

Vedeti mai jos primele 2399 zecimale ale numarului pi:

3,141592653589793238462643383279502884197169399375105820974944592307816406286208  
99862803482534211706798214808651328230664709384460955058223172535940812848111745  
02841027019385211055596446229489549303819644288109756659334461284756482337867831  
65271201909145648566923460348610454326648213393607260249141273724587006606315588  
17488152092096282925409171536436789259036001133053054882046652138414695194151160  
94330572703657595919530921861173819326117931051185480744623799627495673518857527  
24891227938183011949129833673362440656643086021394946395224737190702179860943702  
77053921717629317675238467481846766940513200056812714526356082778577134275778960  
91736371787214684409012249534301465495853710507922796892589235420199561121290219  
60864034418159813629774771309960518707211349999998372978049951059731732816096318  
59502445945534690830264252230825334468503526193118817101000313783875288658753320  
83814206171776691473035982534904287554687311595628638823537875937519577818577805  
32171226806613001927876611195909216420198938095257201065485863278865936153381827  
96823030195203530185296899577362259941389124972177528347913151557485724245415069  
59508295331168617278558890750983817546374649393192550604009277016711390098488240  
12858361603563707660104710181942955596198946767837449448255379774726847104047534  
64620804668425906949129331367702898915210475216205696602405803815019351125338243  
00355876402474964732639141992726042699227967823547816360093417216412199245863150  
30286182974555706749838505494588586926995690927210797509302955321165344987202755  
96023648066549911988183479775356636980742654252786255181841757467289097777279380  
00816470600161452491921732172147723501414419735685481613611573525521334757418494  
68438523323907394143334547762416862518983569485562099219222184272550254256887671  
79049460165346680498862723279178608578438382796797668145410095388378636095068006  
42251252051173929848960841284886269456042419652850222106611863067442786220391949  
45047123713786960956364371917287467764657573962413890865832645995813390478027590  
09946576407895126946839835259570982582262052248940772671947826848260147699090264  
01363944374553050682034962524517493996514314298091906592509372216964615157098583  
87410597885959772975498930161753928468138268683868942774155991855925245953959431  
04997252468084598727364469584865383673622262609912460805124388439045124413654976  
27807977156914359977001296160894416948685558484063534220722258284886481584560285

# Capitolul III

## Exercitiu 00 : ft\_iterative\_factorial

|   |               |
|---|---------------|
|  | Exercitiu: 00 |
| ft_iterative_factorial  |               |
| Director de lucru: ex00/  |               |
| Fisier(e) de iesire: ft_iterative_factorial.c                                     |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |


- Scrieti o functie iterativa care returneaza un numar. Acest numar este rezultatul operatorului factorial aplicat asupra parametrului functiei.
- In caz de eroare functia va returna 0.
- Ea trebuie sa aiba prototipul urmator:

```
int ft_iterative_factorial(int nb);
```

- Functia trebuie sa dea rezultatul in mai putin de doua secunde.

# Capitolul IV

## Exercitiu 01 : ft\_recursive\_factorial


|   |                        |
|---|------------------------|
|  | Exercitiu: 01          |
|   | ft_recursive_factorial |
| Director de lucru: <i>ex01/</i>   |                        |
| Fisier(e) de iesire: <b>ft_recursive_factorial.c</b>                              |                        |
| Functii autorizate: Niciuna   |                        |
| Observatii: n/a   |                        |

- Scrieti o functie recursiva care returneaza factorialul numarului transmis ca parametru.
- Functia va trebui sa se comporte ca functia anterioara.
- Ea trebuie sa aiba prototipul urmator:

```
int ft_recursive_factorial(int nb);
```

# Capitolul V

## Exercitiu 02 : ft\_iterative\_power

|   |               |
|---|---------------|
|  | Exercitiu: 02 |
| ft_iterative_power  |               |
| Director de lucru: ex02/  |               |
| Fisier(e) de iesire: ft_iterative_power.c   |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |

- Scrieti o functie iterativa care returneaza puterea unui numar. O putere mai mica decat 0 (zero) va returna 0. Cazurile de depasire a memoriei alocate nu trebuie gestionate.
- Ea trebuie sa aiba prototipul urmator:


```
int ft_iterative_power(int nb, int power);
```

- Functia trebuie sa dea rezultatul in mai putin de doua secunde.



# Capitolul VI

## Exercitiu 03 : ft\_recursive\_power


|   |               |
|---|---------------|
|  | Exercitiu: 03 |
| ft_recursive_power  |               |
| Director de lucru: ex03/  |               |
| Fisier(e) de iesire: ft_recursive_power.c   |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |

- Scrieti o functie recursiva care returneaza puterea unui numar.
- Functia va trebui sa se comporte ca functia anterioara.
- Ea trebuie sa aiba prototipul urmator:

```
int ft_recursive_power(int nb, int power);
```

# Capitolul VII

## Exercitiu 04 : ft\_fibonacci

|   |               |
|---|---------------|
|  | Exercitiu: 04 |
|   | ft_fibonacci  |
| Director de lucru: <i>ex04/</i>   |               |
| Fisier(e) de iesire: <b>ft_fibonacci.c</b>  |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |


- Scrieti o functie `ft_fibonacci` care returneaza al `n`-lea element al seriei Fibonacci, primul fiind cel corespunzator indicelui 0. Consideram ca seria Fibonacci incepe astfel cu 0, 1, 1, 2.
- Ea trebuie sa aiba prototipul urmator:

```
int ft_fibonacci(int index);
```

- Evident, `ft_fibonacci` trebuie sa fie recursiva.
- Daca `index` este mai mic ca 0, functia returneaza -1.

# Capitolul VIII

## Exercitiu 05 : ft\_sqrt

|   |               |
|---|---------------|
|  | Exercitiu: 05 |
| ft_sqrt   |               |
| Director de lucru: ex05/  |               |
| Fisier(e) de iesire: ft_sqrt.c  |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |


- Scrieti o functie care returneaza radacina patrata intreaga daca aceasta exista si 0 daca radacina patrata este diferita de un intreg.
- Ea trebuie sa aiba urmatorul prototip:

```
int ft_sqrt(int nb);
```

- Functia trebuie sa dea rezultatul in mai putin de doua secunde.

# Capitolul IX

## Exercitiu 06 : ft\_is\_prime

|   |               |
|---|---------------|
|  | Exercitiu: 06 |
|   | ft_is_prime   |
| Director de lucru: ex06/  |               |
| Fisier(e) de iesire: ft_is_prime.c  |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |

- Scrieti o functie care returneaza 1 daca numarul este prim si 0 in caz contrar.
- Ea trebuie sa aiba prototipul urmator:

```
int ft_is_prime(int nb);
```


- Functia trebuie sa dea rezultatul in mai putin de doua secunde.



0 si 1 nu sunt numere prime.

# Capitolul X

## Exercitiu 07 : ft\_find\_next\_prime

|   |               |
|---|---------------|
|  | Exercitiu: 07 |
| ft_find_next_prime  |               |
| Director de lucru: ex07/  |               |
| Fisier(e) de iesire: ft_find_next_prime.c   |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |


- Scrieti o functie care returneaza numarul prim imediat superior sau egal cu parametrul functiei.
- Ea trebuie sa aiba urmatorul prototip:

```
int ft_find_next_prime(int nb);
```

- Functia trebuie sa dea rezultatul in mai putin de doua secunde.

# Capitolul XI

## Exercitiu 08 : Cele opt regine

|   |               |
|---|---------------|
|  | Exercitiu: 08 |
| Les huit dames 1  |               |
| Director de lucru: <i>ex08/</i>   |               |
| Fisier(e) de iesire: <code>ft_eight_queens_puzzle.c</code>                        |               |
| Functii autorizate: Niciuna   |               |
| Observatii: n/a   |               |


- Scopul acestui joc este de a plasa opt regine pe o tablă de sah, astfel incat sa nu se poata captura dintr-o singura mutare una pe alta.
- Reîmprospătați-va cunostintele cu privire la regulile de sah.
- Bineinteles, se va folosi recursivitatea pentru a rezolva aceasta problema.
- Scrieti o functie care returneaza numarul de posibilitati de plasare a celor opt regine pe tabla de sah fara ca ele sa se poata captura.
- Ea trebuie sa aiba urmatorul prototip:

```
int ft_eight_queens_puzzle(void);
```

- Functia trebuie sa dea rezultatul in mai putin de doua secunde.

# Capitolul XII

## Exercitiu 09 : Cele opt regine 2

|   |               |
|---|---------------|
|  | Exercitiu: 09 |
| Les huit dames 2  |               |
| Director de lucru: <i>ex09/</i>   |               |
| Fisier(e) de iesire: <code>ft_eight_queens_puzzle_2.c</code>                      |               |
| Functii autorizate: <code>ft_putchar</code>                                       |               |
| Observatii: n/a   |               |

- Scrieti o functie care afiseaza toate posibilitatile de a plasa cele opt regine pe tabla de sah astfel incat ele sa nu se poata captura reciproc.
- Se va utiliza recursivitatea si in acest caz.
- Ea trebuiesc aiba prototipul urmator:

```
void ft_eight_queens_puzzle_2(void);
```

- Afisarea se va face in felul urmator:

```
$> ./a.out
15863724
16837425
17468253
...
```

- Sirul se citeste de la stanga la dreapta. Prima cifra corespunde pozitiei primei regine din prima coloana (1(indicele incepe cu 1). Cifra a n-a corespunde pozitiei a n-a din coloana a n-a.
- Există o linie nouă după ultima solutie.
- Functia trebuie sa dea rezultatul in mai putin de doua secunde.