



Piscina C

Ziua 08

Staff Academy+Plus contact@academyplus.ro

Sumar: Acest document este subiectul zilei a 08 a piscinei C din cadrul Academy+Plus.

Cuprins

I	Instructiuni	2
II	Preambul	4
III	Exercitiu 00 : ft_split_whitespaces	5
IV	Exercitiu 01 : ft.h	6
V	Exercitiu 02 : ft_boolean.h	7
VI	Exercitiu 03 : ft_abs.h	9
VII	Exercitiu 04 : ft_point.h	10
VIII	Exercitiu 05 : ft_param_to_tab	11
IX	Exercice 06 : ft_show_tab	13

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 vostri 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

Iată ce spune enciclopedia colaborativă *Wikipedia* despre ornitorinc:

Ornitorincul (*Ornithorhynchus anatinus*) este o specie de mamifer mic, semi-acvatic, endemic din estul Australiei, inclusiv Tazmania. Este una dintre cele cinci specii din ordinul monotremelor, singurul ordin de mamifere care depune ouă în loc să dea naștere la pui complet formați (celelalte patru specii sunt echidne). Este singura specie care a supraviețuit din familia Ornithorhynchidae și din genul *Ornithorhynchus*, deși au fost descoperite numeroase fragmente de specii fosile din această familie și din acest gen.


Primii exploratori care au descoperit ornitorincul au fost puternic surprinși de aspectul ciudat al acestui mamifer care depune ouă, care are țepi veninoși, o falcă cu corn, ce seamănă cu ciocul de rață, o coadă care amintește de un castor, pe care o folosește drept cârmă în apă și ca rezervă de grăsime și labe de vidră; un număr mare de naturaliști europeni au crezut că este o glumă. Este unul dintre puținele mamifere veninoase: pe labele din spate, masculul are un ac ce poate elibera venin, care poate provoca dureri puternice unui om. Trăsăturile unice ale ornitorincului fac obiectul multor studii realizate pentru a înțelege mai bine evoluția speciilor animale și îl transformă într-unul dintre simbolurile Australiei: acesta a fost folosit ca mascotă pentru numeroase evenimente naționale și apare pe spatele monedei de 20 de cenți australieni.

Până la începutul secolului XX, a fost vânat pentru blană, dar în prezent este protejat. Deși programele de reproducere în captivitate au avut un succes limitat și deși ornitorincul este sensibil la efectele poluării, specia nu este considerată încă în pericol.

Acest subiect nu tratează ornitorincul.

Capitolul III

Exercitiu 00 : ft_split whitespaces

	Exercitiu: 00
ft_split whitespaces	
Director de lucru: ex00/	
Fisier(e) de iesire: ft_split whitespaces.c	
Functii autorizate: malloc	
Observatii: n/a	


Academy+ - Classics : Aceste exercitii sunt inevitabile chiar daca nu aduc nici un punct, este obligatoriu sa le validati pentru a accesa adevaratele exercitii ale zilei.

- Scrieti o functie care decupeaza un sir de caractere in cuvinte.
- Separatoarele vor fi spatiul, tabulatorul si returul de linie.
- Functia va returna un tabel cu fiecare celula continand adresa unui sir de caractere ce reprezinta un cuvint. Ultimul element al tabelului trebuie sa fie egal cu 0 pentru a marca sfarsitul tabloului.
- Nu trebuie sa existe celule cu siruri vide in tablou. Trageti concluziile ce se impun.
- Sirul care va fi transmis nu va mai putea fi modificat.
- Functia trebuie sa aiba prototipul urmator:

```
char **ft_split whitespaces(char *str);
```

Capitolul IV


Exercitiu 01 : ft.h

	Exercitiu: 01
	ft.h
Director de lucru: <i>ex01/</i>	
Fisier(e) de iesire: ft.h	
Functii autorizate: Niciuna	
Observatii: n/a	

- Creati un fisier **ft.h**
- El trebuie sa contina prototipurile functiilor create de voi in biblioteca **libft.a**.

Capitolul V

Exercitiu 02 : ft_boolean.h

	Exercitiu: 02
	ft_boolean.h
	Director de lucru: ex02/
	Fisier(e) de iesire: ft_boolean.h
	Functii autorizate: Niciuna
	Observatii: n/a

- Creati un fisier `ft_boolean.h` care va face functia main de mai jos sa fie compilabila si programul rezultat sa ruleze corect:

```
#include "ft_boolean.h"

void      ft_putstr(char *str)
{
    while (*str)
        write(1, str++, 1);
}

t_bool    ft_is_even(int nbr)
{
    return ((EVEN(nbr)) ? TRUE : FALSE);
}

int       main(int argc, char **argv)
{
    (void)argv;
    if (ft_is_even(argc - 1) == TRUE)
        ft_putstr(EVEN_MSG);
    else
        ft_putstr(ODD_MSG);
    return (SUCCESS);
}
```

- Acest program trebuie sa afiseze

```
I have a pair number of arguments.
```



- sau

```
I have an impair number of arguments.
```

- urmat de un retur de linie, dupa caz.

Capitolul VI

Exercitiu 03 : ft_abs.h


	Exercitiu: 03
	ft_abs.h
Director de lucru: ex03/	
Fisier(e) de iesire: ft_abs.h	
Functii autorizate: Niciuna	
Observatii: n/a	

- Scrieti o functie macro ABS care isi inlocuieste parametrul cu valoarea sa absoluta:

```
#define ABS(Value)
```

Capitolul VII

Exercitiu 04 : ft_point.h

	Exercitiu: 04
ft_point.h	
Director de lucru: ex04/	
Fisier(e) de iesire: ft_point.h	
Functii autorizate: Niciuna	
Observatii: n/a	

- Scrieti un fisier `ft_point.h` care va face functia main de mai jos sa fie compilabila:

```
#include "ft_point.h"


void      set_point(t_point *point)
{
    point->x = 42;
    point->y = 21;
}

int       main(void)
{
    t_point      point;

    set_point(&point);
    return (0);
}
```

Capitolul VIII

Exercitiu 05 : ft_param_to_tab

	Exercitiu: 05
ft_param_to_tab	
Director de lucru: ex05/	
Fisier(e) de iesire: ft_param_to_tab.c, ft_stock_par.h	
Functii autorizate: ft_split_whitespaces, ft_show_tab, malloc	
Observatii: n/a	

- Scrieti o functie care stocheaza parametri programului intr-un tablou de structuri si care returneaza adresa primului element al tabloului.
- Toate elementele tabloului trebuie stocate, inclusiv av[0].
- Functia trebuie sa aiba urmatorul prototip:

```
struct s_stock_par *ft_param_to_tab(int ac, char **av);
```

- Tabloul de structuri trebuie sa fie alocat si ultima celula va contine 0 in elementul sau str pentru a semnaliza sfarsitul.


- Structura este definita intr-un fisier `ft_stock_par.h` dupa cum urmeaza:

```
typedef struct s_stock_par
{
    int size_param;
    char *str;
    char *copy;
    char **tab;
} t_stock_par;
```

- `size_param` fiind lungimea parametrului;
 - `str` fiind adresa parametrului;
 - `copy` fiind copia parametrului;
 - `tab` fiind tabloul returnat de `ft_split_whitespaces`.
- Va vom testa functia voastra cu implementarile noastre pentru `ft_split_whitespaces` si `ft_show_tab` (vezi exercitiul urmator). Luati masurile necesare pentru ca acest lucru sa functioneze!

Capitolul IX

Exercice 06 : ft_show_tab

	Exercitiu: 06
	ft_show_tab
	Director de lucru: ex06/
	Fisier(e) de iesire: ft_show_tab.c, ft_stock_par.h
	Functii autorizate: ft_putchar
	Observatii: n/a

- Scrieti o functie care afiseaza continutul unui tabloul creat de functia precedenta.
- Ea trebuie sa aiba prototipul urmator:

```
void ft_show_tab(struct s_stock_par *par);
```

- Structura e definita intr-un fisier ft_stock_par.h dupa cum urmeaza:

```
typedef struct s_stock_par
{
    int size_param;
    char *str;
    char *copy;
    char **tab;
} t_stock_par;
```

- Pentru fiecare element din tablou se vor afisa (cate o valoare pe fiecare rand):
 - parametrul
 - marimea
 - fiecare cuvant (unu pe rand)

- Va vom testa functia voastra cu implementarea noastra pentru `ft_param_to_tab` (vezi exercitiul anterior). Luati masurile necesare pentru ca acest lucru sa functioneze!