



Piscina C

C 08

Sommario: Questo documento tratta il modulo C 08 della Piscina C @ 42.

Indice

I	Istruzioni	2
II	Preambolo	4
III	Esercizio 00 : ft.h	5
IV	Esercizio 01 : ft_boolean.h	6
V	Esercizio 02 : ft_abs.h	8
VI	Esercizio 03 : ft_point.h	9
VII	Esercizio 04 : ft_strs_to_tab	10
VIII	Esercizio 05 : ft_show_tab	12

Capitolo I

Istruzioni

- Fate riferimento solo a questa pagina: non fidatevi delle dicerie.
- Questo documento può subire variazioni prima della scadenza per la presentazione.
- Controllate i permessi dei vostri file e delle vostre cartelle.
- Dovete seguire le procedure di presentazione per tutti gli esercizi.
- I vostri esercizi saranno controllati e valutati dai vostri compagni di corso.
- Moulinette sarà estremamente meticolosa e severa nel valutare il vostro lavoro. Essendo il suo un processo automatico senza possibilità di ricorso, assicuratevi di essere il più precisi possibile al fine di evitare brutte sorprese.
- I vostri esercizi saranno soggetti, oltre alla valutazione tra pari, al controllo e alla valutazione da parte di un programma chiamato Moulinette.
- Moulinette non ha una mentalità aperta. Non proverà a comprendere il vostro codice se non rispetta la Norma. Moulinette utilizza un programma di nome **norminette** per controllare la validità dei vostri file. TL;DR: sarebbe scocco tentare di consegnare un esercizio che non pass il controllo di **norminette**.
- Gli esercizi sono presentati seguendo un ordine di difficoltà crescente. Ai fini della valutazione **NON** si prendono in considerazione gli esercizi se i precedenti non sono stati completati correttamente
- Usare una funzione non autorizzata viene considerato come barare. Chi bara ottiene un **-42** senza possibilità di ricorso.
- Dovrete consegnare una funzione `main()` solo se l'esercizio richiede un programma.
- Moulinette compila per mezzo di `gcc` utilizzando queste flag: `-Wall -Wextra -Werror`.
- Se il vostro programma non compila, il voto sarà 0.
- NON sarà tollerato ALCUN file aggiuntivo nelle cartelle presentate oltre a quelli specificati in questo documento.

- Dubbi o domande? Chiedi a chi si trova alla tua destra, altrimenti a chi si trova alla tua sinistra
- Your reference guide is called **Google / man / the Internet /**
- Date un'occhiata alla sezione Piscina C del forum dell'Intranet.
- Prestate attenzione agli esempi proposti, in quanto potrebbero mostrare dettagli non esplicitamente presentati nel documento...
- Per Odin, Per Thor ! Usate la testa !!!

Capitolo II

Preambolo

Here's what Wikipedia have to say about Platypus :

The platypus (*Ornithorhynchus anatinus*), also known as the duck-billed platypus, is a semiaquatic egg-laying mammal endemic to eastern Australia, including Tasmania. Together with the four species of echidna, it is one of the five extant species of monotremes, the only mammals that lay eggs instead of giving birth. The animal is the sole living representative of its family (*Ornithorhynchidae*) and genus (*Ornithorhynchus*), though a number of related species have been found in the fossil record.


The unusual appearance of this egg-laying, duck-billed, beaver-tailed, otter-footed mammal baffled European naturalists when they first encountered it, with some considering it an elaborate hoax. It is one of the few venomous mammals, the male platypus having a spur on the hind foot that delivers a venom capable of causing severe pain to humans. The unique features of the platypus make it an important subject in the study of evolutionary biology and a recognisable and iconic symbol of Australia; it has appeared as a mascot at national events and is featured on the reverse of its 20-cent coin. The platypus is the animal emblem of the state of New South Wales.

Until the early 20th century, it was hunted for its fur, but it is now protected throughout its range. Although captive breeding programs have had only limited success and the platypus is vulnerable to the effects of pollution, it is not under any immediate threat.

This subject is absolutly not talking about platypus.

Capitolo III

Esercizio 00 : ft.h


	Esercizio 00
	ft.h
	Cartella per la consegna : <i>ex00/</i>
	File da consegnare : ft.h
	Funzioni permesse : Nessuna

- Creare il file `ft.h`.
- Deve contenere i prototipi di funzione:

```
void    ft_putchar(char c);
void    ft_swap(int *a, int *b);
void    ft_putstr(char *str);
int     ft_strlen(char *str);
int     ft_strcmp(char *s1, char *s2);
```

Capitolo IV

Esercizio 01 : ft_boolean.h

	Esercizio 01
	ft_boolean.h
	Cartella per la consegna : <i>ex01/</i>
	File da consegnare : ft_boolean.h
	Funzioni permesse : Nessuna

- Creare un file **ft_boolean.h** che compili ed esegua correttamente il seguente main :

```
#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);
}
```

- Questo programma deve stampare :

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

- O

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


- seguito da una nuova linea quando appropriato.



Norminette va utilizzata con la flag `-R CheckForbiddenSourceHeader`.
Moulinette farà la stessa cosa.

Capitolo V

Esercizio 02 : ft_abs.h

	Esercizio 02
	ft_abs.h
	Cartella per la consegna : <i>ex02/</i>
	File da consegnare : ft_abs.h
	Funzioni permesse : Nessuna

- Creare il macro ABS che sostituisce al suo argomento il suo valore assoluto :


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



Norminette va utilizzata con la flag `-R CheckForbiddenSourceHeader`.
Moulinette farà la stessa cosa.

Capitolo VI

Esercizio 03 : ft_point.h

	Esercizio 03
	ft_point.h
	Cartella per la consegna : <i>ex03/</i>
	File da consegnare : ft_point.h
	Funzioni permesse : Nessuna

- Creare il file **ft_point.h** che compili con il seguente main :

```
#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);
}
```

Capitolo VII

Esercizio 04 : ft_strs_to_tab

	Esercizio 04
	ft_strs_to_tab
	Cartella per la consegna : ex04/
	File da consegnare : ft_strs_to_tab.c
	Funzioni permesse : malloc, free

- Creare una funzione che abbia come argomenti una array di stringhe e la dimensione di questo array.
- Il prototipo è il seguente :

```
struct s_stock_str *ft_strs_to_tab(int ac, char **av);
```

- Convertirà ogni elemento di av in una struttura.
- La struttura è definita nel file ft_stock_str.h provvisto da noi :


```
typedef struct s_stock_str
{
    int size;
    char *str;
    char *copy;
} t_stock_str;
```

- size è la lunghezza della stringa;
 - str è la stringa;
 - copy è una copia della stringa ;
- L'ordine deve essere lo stesso di av.

- L'array restituito deve essere allocato in memoria e l'ultimo elemento **str** impostato a 0 per indicare la fine dell'array.
- Deve restituire un puntatore a NULL se vi sono errori.
- Testeremo la vostra funzione con il nostro `ft_show_tab` (prossimo esercizio).

Capitolo VIII

Esercizio 05 : ft_show_tab

	Esercizio 05
	ft_show_tab
	Cartella per la consegna : <i>ex05/</i>
	File da consegnare : ft_show_tab.c
	Funzioni permesse : write

- Creare una funzione che stampi a video il contenuto dell'array creato dalla funzione precedente.
- Il prototipo è il seguente :

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

- La struttura sarà la stessa dello stesso esercizio definita nel file **ft_stock_str.h**.
- Per ogni elemento stamperemo :
 - la stringa seguita da `'\n'`
 - la dimensione seguita da `'\n'`
 - la copia della stringa (che può aver subito modifiche) seguita da `'\n'`
- Testeremo la vostra funzione con il nostro **ft_show_tab** (prossimo esercizio).