



Piscina C

C 03

Sommario: Questo documento tratta il modulo C 03 della Piscina C @ 42.

Indice

I	Istruzioni	2
II	Preambolo	4
III	Esercizio 00 : ft_strcmp	5
IV	Esercizio 01 : ft_strncmp	6
V	Esercizio 02 : ft_strcat	7
VI	Esercizio 03 : ft_strncat	8
VII	Esercizio 04 : ft_strstr	9
VIII	Esercizio 05 : ft_strlcat	10
IX	Consegna e valutazione tra pari	11

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 !!!



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

Capitolo II

Preambolo

The first known mention of the game of RPS was in the book *Wuzazu* written by the Chinese Ming-dynasty writer Xie Zhaozhi who wrote that the game dated back to the time of the Chinese Han dynasty (206 BC – 220 AD). In the book, the game was called *shoushiling*. Li Rihua's book *Note of Liuyanzhai* also mentions this game, calling it *shoushiling*, *huozhitou*, or *huoquan*.

Throughout Japanese history there are frequent references to “*sansukumi-ken*”, meaning “*ken*” fist games with a “*san*” three-way “*sukumi*” deadlock. This is in the sense that A beats B, B beats C, and C beats A. The games originated in China before being imported to Japan and subsequently becoming popular.


By the early 20th century, rock–paper–scissors had spread beyond Asia, especially through increased Japanese contact with the west. Its English-language name is therefore taken from a translation of the names of the three Japanese hand-gestures for rock, paper and scissors: elsewhere in Asia the open-palm gesture represents “cloth” rather than “paper”. The shape of the scissors is also adopted from the Japanese style.

In 1927 *La Vie au patronage*, a children's magazine in France, described it in detail, referring to it as a “*jeu japonais*” (“Japanese game”). Its French name, “*Chi-fou-mi*”, is based on the Old Japanese words for “one, two, three” (“*hi, fu, mi*”)

A New York Times article of 1932 on the Tokyo rush hour describes the rules of the game for the benefit of American readers, suggesting it was not at that time widely known in the U.S. The 1933 edition of the *Compton's Pictured Encyclopedia* described it as a common method of settling disputes between children in its article on Japan; the name was given as “*John Kem Po*” and the article pointedly asserted, “This is such a good way of deciding an argument that American boys and girls might like to practice it too.”

Capitolo III

Esercizio 00 : ft_strcmp


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

- Riproduci il comportamento della funzione **strcmp** (man strcmp).
- Il prototipo è il seguente :

```
int      ft_strcmp(char *s1, char *s2);
```

Capitolo IV

Esercizio 01 : ft_strncmp


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

- Riproduci il comportamento della funzione **strncmp** (man **strncmp**).
- Il prototipo è il seguente :

```
int      ft_strncmp(char *s1, char *s2, unsigned int n);
```

Capitolo V

Esercizio 02 : ft_strcat


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

- Riproduci il comportamento della funzione `strcat` (man `strcat`).
- Il prototipo è il seguente :

```
char *ft_strcat(char *dest, char *src);
```


Capitolo VI

Esercizio 03 : ft_strncat

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

- Riproduci il comportamento della funzione `strncat` (`man strncat`).
- Il prototipo è il seguente :

```
char *ft_strncat(char *dest, char *src, unsigned int nb);
```

Capitolo VII

Esercizio 04 : ft_strstr


	Esercizio 04
	ft_strstr
	Cartella per la consegna : <i>ex04/</i>
	File da consegnare : ft_strstr.c
	Funzioni permesse : Nessuna

- Riproduci il comportamento della funzione **strstr** (man strstr).
- Il prototipo è il seguente :

```
char      *ft_strstr(char *str, char *to_find);
```

Capitolo VIII

Esercizio 05 : ft_strlcat

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

- Riproduci il comportamento della funzione `strlcat` (`man strlcat`).
- Il prototipo è il seguente :

```
unsigned int ft_strlcat(char *dest, char *src, unsigned int size);
```

Capitolo IX

Consegna e valutazione tra pari

Consegna gli esercizi nella tuo repository `Git` come al solito. Durante la difesa verrà considerato unicamente ciò che si trova all'interno della repository. Assicurati di controllare che i nomi dei tuoi file siano corretti.



Devi consegnare solo i file richiesti da questo documento.