

Esercizio 1

Si discuta la rappresentazione dei numeri nei calcolatori descrivendo in dettaglio il complemento A2.

Esercizio 2

Si discuta il tipo puntatore in C.

Esercizio 3

Individuare e motivare gli errori di **sintassi** che pregiudicano la compilazione del seguente programma modulare. Dopo averli corretti dire cosa stampa il programma una volta compilato ed eseguito.

file `fun.h`

```
1 void fun(char *);
```

file `fun.c`

```
1 void fun(char *str)
2 {
3     char* ptr;
4     for (ptr=str; *ptr!="\0"; ptr++) {
5         *ptr=*ptr+1
6     }
```

file `main.c`

```
1 #include <stdio.h>
2 #include <fun.h>
3
4 int main(void)
5 {
6     char str []="anno";
7     printf("Passo 0: %s\n", str);
8     fun(str);
9     printf("Passo 1: %s\n", str);
10    fun(str);
11    printf("Passo 2: %s\n", str);
12    return (0);
13 }
```

comandi shell

```
gcc -c fun.h -o fun.o
```

```
gcc -c main.c -o main.o
```

```
gcc -c fun.o main.o -o programma
```

```
./programma
```

Soluzione Esercizio 3

file `fun.c`

- A riga 4 mettere gli apici singoli al posto di quelli doppi, visto che ci riferiamo ad un carattere singolo e non ad una stringa. C'è anche una parentesi parentesi graffa di troppo: toglierla oppure aggiungere la chiusa dopo riga 5.
- A riga 5 manca il ; finale

Nome _____

Cognome _____

Matricola _____

file `main.c`

- A riga 2 per includere il file `fun.h` la include vuole i doppi apici `""` e non le parentesi `{}.`

comandi shell CORRETTI

```
gcc -c fun.c -o fun.o (fun.c e non fun.h)
```

```
gcc -c main.c -o main.o
```

```
gcc fun.o main.o -o programma (senza -c visto che stiamo linkando i file oggetto invece che compilare)
```

```
./programma
```

OUTPUT:

- Passo 0: anno
- Passo 1: boop
- Passo 2: cppq