

3 Costrutti `while` e `for`

Questa dispensa propone esercizi sulla scrittura di algoritmi, in linguaggio C, utili alla comprensione dei costrutti `while` e `for`.

I costrutti per costruire cicli in C sono il `while` (e la variante `do...while` e il `for`.

```
inizializzazione; //opzionale

while (condizione) {
    corpo;

    incremento; //opzionale
}
```

La condizione è valutata prima di ogni iterazione, inclusa la prima; quindi il corpo del ciclo (e quindi anche l'eventuale istruzione di incremento) potrebbero non eseguire mai, nel caso in cui la condizione sia falsa dall'inizio.

Invece, nella variante `do...while`, il corpo e l'eventuale istruzione di incremento sono eseguiti almeno una volta prima di valutare la condizione.

```
inizializzazione; //opzionale

do {
    corpo;

    incremento; //opzionale
} while (condizione);
```

Il `for` è equivalente, ma ha una sintassi più compatta:

```
for (inizializzazione; condizione; incremento) {
    corpo;
}
```

Si noti che le parentesi sono necessarie solo nel caso di corpo con più di un'istruzione. Inoltre, l'espressione di inizializzazione e di incremento sono opzionali. Di fatto, un ciclo `while` può essere scritto in modo equivalente con un costrutto `for` nel seguente modo:

```
inizializzazione;  
  
for (; condizione; ) {  
    corpo;  
  
    incremento;  
}
```

3.1 Esercizi

Esercizio 3.1

Scrivere un programma che richiede all'utente un intero positivo e determina se è primo o meno. Il programma deve continuare a chiedere il numero fino a che l'utente non ne inserisce uno positivo.

Esercizio 3.2

Scrivere un programma che richiede all'utente un intero positivo N e stampa a schermo i primi N numeri primi. Ad esempio: con $N = 7$ a schermo avremo 1 2 3 5 7 11 13.

Esercizio 3.3

Scrivere un programma che richiede all'utente un intero positivo e ne stampa a schermo tutti i divisori.

Esercizio 3.4

Scrivere un programma che richiede due numeri e ne stampa il massimo comune divisore.

Esercizio 3.5

Implementare il gioco della morra cinese, noto anche come "carta-forbice-sasso". Il programma si aspetta da tastiera l'input dal giocatore 1 e, dopo, l'input dal giocatore 2. L'input potrà essere uno dei caratteri 'C', 'F' o 'S'. Il programma dovrà continuare a chiedere input ai due giocatori fino a che uno dei due giocatori non inseriscono una giocata valida a testa. In questo caso calcola il vincitore (giocatore 1 o 2), stampa a video il vincitore e termina.