

Esercitazione di Laboratorio: Funzioni Ricorsive

Esercizio 1

Nel file `fattoriale.c` implementare la definizione della funzione:

```
unsigned long long Fattoriale(int n);
```

La funzione accetta come parametro un numero intero n e ritorna il suo fattoriale $n!$, calcolato ricorsivamente. Se n è minore di zero la funzione deve ritornare 0.

Esercizio 2

Nel file `minimo.c` implementare la definizione della funzione:

```
int Minimo(const int *v, int v_size);
```

La funzione accetta come parametri un vettore di interi v e la sua dimensione v_size in numero di elementi. Il valore di ritorno deve essere il minimo tra gli elementi dell'array, calcolato ricorsivamente. Il vettore non deve essere modificato.

Si assuma che v sia sempre diverso da NULL e che v_size sia sempre maggiore o uguale ad 1.

Esercizio 3

Nel file `fibonacci.c` implementare la definizione della funzione:

```
int Fibonacci(int n);
```

La funzione accetta come parametro un numero intero positivo n e ritorna l' n -esimo numero della successione di Fibonacci calcolato ricorsivamente.

Una delle possibili formulazioni della successione è la seguente:

$$\begin{aligned}F_0 &= 0, \\F_1 &= 1, \\F_n &= F_{n-1} + F_{n-2}\end{aligned}$$

Nel caso che n sia minore di zero la funzione deve ritornare -1.

Esercizio 4

Nel file `prodotto.c` implementare la definizione della funzione:

```
int Prodotto(int a, int b);
```

La funzione accetta come parametri due numeri interi positivi e ritorna il loro prodotto calcolato ricorsivamente, senza utilizzare l'operatore `*`.

Se uno dei due numeri è negativo la funzione deve ritornare `-1`.

Esercizio 5

Nel file `prodotto_negativi.c` implementare la definizione della funzione:

```
int ProdottoNegativi(int a, int b);
```

La funzione accetta come parametri due numeri interi (anche negativi) e ritorna il loro prodotto calcolato ricorsivamente, senza utilizzare l'operatore `*`.

Esercizio 6

Nel file `divisione.c` implementare la definizione della funzione:

```
int Divisione(int a, int b);
```

La funzione accetta come parametri due numeri interi positivi e ritorna il loro quoziente (a/b), calcolato ricorsivamente senza utilizzare l'operatore `/`.

Se uno dei due valori è negativo o se la divisione è impossibile la funzione deve ritornare `-1`.

Esercizio 7

Nel file `divisione_negativi.c` implementare la definizione della funzione:

```
int DivisioneNegativi(int a, int b);
```

La funzione accetta come parametri due numeri interi e ritorna il loro quoziente (a/b) calcolato ricorsivamente senza utilizzare l'operatore `/`.

Se la divisione è impossibile, la funzione deve ritornare `INT_MAX` definito in `limits.h`.