

# Laboratorio di Algoritmi e Strutture Dati

Esercitazioni del 13 Novembre 2012

## Esercizio 2: Aritmetica intera di precisione estesa

Si desidera implementare un tipo di numeri interi, chiamato `INT`, che permetta di memorizzare valori interi molto grandi, senza perdita di precisione.

In particolare, un `INT` sarà rappresentato come un array di `DIGITS` `char`, dove `DIGITS` è una macro (assumiamo `DIGITS` sia 500).

Sia  $d$  un numero intero e sia  $c_1 c_2 \dots c_k$  la sua rappresentazione in cifre in base 10, dunque  $c_i \in \{0, 1, \dots, 9\}$ .

La rappresentazione di  $d$  sarà la seguente:

$$\{c_k, c_{k-1}, \dots, c_2, c_1, -1, -1, \dots, -1\}.$$

### Esempio:

Il numero 125 sarà rappresentato come

{ 5, 2, 1, -1, -1, ..., -1 }

Si richiede di implementare un programma che:

1. Introduca le seguenti due linee:

```
#define DIGITS 500
```

```
typedef char INT[DIGITS];
```

(Dove la dichiarazione `typedef` stabilisce che `INT a` è sinonimo di `char a[DIGITS]`, e dunque per dichiarare una variabile `var` di tipo `INT`, o equivalentemente, di tipo array di `DIGITS` caratteri, basta scrivere:

```
INT var;
```

```
)
```

2. Implementi la funzione:

```
void carica(INT n, int c)
```

che memorizza l'intero  $c$  nell'`INT`ero esteso  $n$ .

3. Implementi la funzione:

```
void stampa(INT n)
```

che visualizzi sullo schermo il valore intero memorizzato in  $n$ .

4. Implementi la funzione:

```
void copia(INT n1, INT n2)
```

che copi **n2** in **n1**.

5. Implementi la funzione:

```
void mult(INT n1, INT n2, INT n3)
```

che ponga in **n3** il valore risultante dalla moltiplicazione di **n1** e **n2**.

6. Utilizzi le funzioni implementate per calcolare il fattoriale di un numero specificato in input.

Per quanto riguarda l'implementazione della funzione `mult`, si consiglia di implementare l'algoritmo della moltiplicazione che avete imparato alle elementari:

**Esempio:**

```
  123 x
   45 =
----
  615
 492
----
5535
```

## Facoltativo

La rappresentazione scelta spreca in genere molto spazio. Infatti usa un intero carattere di 8 bit per memorizzare un valore scelto fra 10 (una cifra decimale per carattere). Modificare l'implementazione rappresentando un intero in base  $2^8 = 256$ .