

Gli studenti dell'a.a. 2006-2007 (6 CFU) devono svolgere 7 degli 8 esercizi marcati "6 CFU" o "3/6 CFU".

Gli studenti degli anni precedenti non passati al nuovissimo ordinamento (3 CFU) devono svolgere i 6 esercizi marcati "3 CFU" e "3/6 CFU".

Gli studenti degli anni precedenti passati al nuovissimo ordinamento (scritto integrativo) devono svolgere i tre esercizi marcati "6 CFU".

Viene espressamente richiesto lo svolgimento di tutti i passaggi e non solo il risultato.

Esercizio 1 (3/6 CFU)

E' data la seguente implementazione **ricorsiva** della funzione fattoriale:

```
unsigned long fattoriale (int n) {  
  
    if (n = 0) return (1);  
    else return (n*fattoriale(n-1));  
  
}
```

Si descriva graficamente e analiticamente l'esecuzione ricorsiva (indicando tutti i passi e gli annodamenti) della funzione sullo stack delle chiamate supponendo **n = 4**.

Descrizioni esclusivamente qualitative non saranno considerate.

Esercizio 2 (3/6 CFU)

Dato un vettore di interi, di 13 componenti così definito:

```
int m[13];
```

scrivere un programma o una funzione in linguaggio C che, ricevuto in ingresso un intero **n**, implementi un algoritmo di ricerca dell'elemento **n** all'interno del vettore **m**, restituendo un valore vero qualora l'elemento sia presente nel vettore. Si supponga che il vettore sia già popolato da numeri interi.

Esercizio 3 (3 CFU)

Data una base B, si considerino i numeri X e Y rappresentati rispettivamente in base B dalle seguenti cifre:

$$X = C_4C_3C_2C_1C_0,$$

$$Y = P_3P_2P_1P_0,$$

dove il valore delle singole cifre è:

$$C_4 = C_3 = C_2 = C_1 = B-1 \quad C_0 = 0$$

$$P_3 = P_2 = P_1 = P_0 = B-1$$

Calcolare il risultato della seguente formula:

$$\left(\frac{X}{B}\right) - Y \cdot B^2 - 1 = \dots$$

Esercizio 4 (3/6 CFU)

Dati i seguenti valori interi su 16 bit $m = (51654)_8$, $n = (17055)_8$,

valutare l'espressione $(m - (n * 3)) / 8$,

operando in aritmetica binaria complemento a due ed utilizzando solo l'operatore di addizione e gli operatori di spostamento.

Esercizio 5 (3/6 CFU)

Applicando i teoremi dell'algebra di Boole, verificare se la seguente espressione è vera o falsa (questa proprietà viene, a volte, chiamata *teorema del consenso*).

$$A \text{ and } B \text{ or } B \text{ and } C \text{ or not } A \text{ and } C = A \text{ and } B \text{ or not } A \text{ and } C$$

Soluzioni svolte esclusivamente con tabelle di verità non saranno accettate.

Esercizio 6 (3/6 CFU)

Dire quali valori stampa la seguente funzione C,

a) quando i valori di s e t siano rispettivamente 5 e 2;

b) quando i valori di s e t siano rispettivamente 4 e 0;

```
#include <stdio.h>

void puntatori(int s, int t)
{
    int x,y;
    int *u,*v;
    x = (s + t);
    y = (s - t)-1;
    u = &x;
    v = &y;
    if ((*u / *v) == (x % y)) {
        v = u;
        u = &y;
    } else {
        *u = (*u) - (*v);
        *v = (*v) + (*u);
    }
    printf("%d %d %d %d\n",x,y,*u,*v);
}
```

Esercizio 7 (6 CFU)

Date le seguenti dichiarazioni:

```
struct elem { char c;
struct elem *prox; };
typedef struct elem ElementoLista;
```

scrivere la funzione **ricorsiva**:

```
int ContaLeMauscole(ElementoLista *L)
```

che restituisce il numero (le occorrenze) di lettere 'a' presenti nella lista passata come parametro

Esercizio 8 (6 CFU)

Creare un programma che data una stringa memorizzata in un array esegua una scansione di tale array e rovesci il testo di una parola sì e una parola no alternativamente.

Esempio: “oggi sono al compito, domani andrò al mare” → “iggo sono la compito, inamod andrò la mare”

Esercizio 9 (6 CFU)

Creare un programma che gestisca una lista ordinata nel seguente modo: nel caso in cui l'utente inserisca il carattere:

- O : vengono immessi nuovi elementi nella lista fino a che non si inserisce lo 0 (zero)
- V : viene visualizzata la lista corrente
- F : fine del programma