

Esame di Informatica Generale I

Mauro Brunato

Mercoledì 15 settembre 2004

Esercizio 1

- 1.1) Enunciare il principio della “valutazione pigra” (“lazy evaluation”) relativamente ai connettivi logici e all’operatore ternario.
- 1.2) Illustrare un esempio con un connettivo logico.

Esercizio 2

Scrivere la funzione

```
void enumera (int v[], int n, int c[], int max)
```

che riceve in ingresso un vettore *v*, contenente *n* elementi di valore compreso tra 0 e *max*, e il vettore non inizializzato *c*, di *max*+1 elementi. La funzione dovrà registrare in *c* il numero di occorrenze di ogni numero tra 0 e *max* nel vettore *v*. In altri termini, per ogni *i* tra 0 e *max*, *c*[*i*] deve contenere il numero di volte che il valore *i* compare nel vettore *v*.

Esercizio 3

Che cosa scrive il seguente programma?

```
#include <stdio.h>

void scrivi (char miao[], int n)
{
    printf ("%s\n", miao+n);
}

int main (void)
{
    int i;
    for ( i = 0; i < 5; i++ )
        scrivi ("34261", i);
    return 0;
}
```

Esistono vari modi di generalizzare la successione di Fibonacci. Uno di questi è il seguente.

La *successione di Fibonacci di ordine n* è la successione a valori naturali i cui primi n termini sono uguali a 1, e ogni termine successivo è uguale alla somma degli n immediatamente precedenti.

La successione classica di Fibonacci si ottiene ponendo $n = 2$.

Esercizio 4

4.1) Scrivere la funzione

```
void fibonacci (int n, int c, int *v)
```

che riempie il vettore v con i primi c termini della successione di Fibonacci di ordine n . Si supponga che il vettore abbia lo spazio necessario.

4.2) Scrivere un programma che chiede all'utente l'ordine della successione, il numero di termini da stampare e, dopo aver invocato la funzione di cui al punto precedente, stampi il vettore risultante. Il vettore deve essere allocato dal programma principale.