

## Soluzioni della prima simulazione di prova d'esame

Mauro Brunato

16 maggio 2006

### Esercizio 1

**1.1)** Per verificare se l'insieme rappresentato dal vettore  $v$  è vuoto, controlliamo se il vettore  $v$  contiene soltanto zeri. Non appena incontriamo un 1, possiamo restituire *false* perché significa che l'insieme contiene almeno un elemento. Se non troviamo elementi non nulli, allora possiamo senz'altro restituire *true*.

```
int vuoto (int s[])
{
    int i;
    for ( i = 0; i < 100; i++ )
        if ( s[i] ) /* se l'i-esima posizione del vettore non contiene 0*/
            return 0; /* allora restituiamo falso */
    return 1; /* Se il programma esce dal ciclo, allora abbiamo */
                /* incontrato solo elementi nulli, quindi l'insieme è vuoto */
}
```

**1.2)** L'unione di due insiemi è quell'insieme che contiene un elemento se uno dei due insiemi lo contiene.

```
void unione (int s1[], int s2[], int s[])
{
    int i;
    for ( i = 0; i < 100; i++ )
        s[i] = s1[i] || s2[i]; /* l'i-esimo elemento appartiene a s se e solo se */
                                /* appartiene ad almeno uno dei due insiemi s1 o s2 */
}
```

**1.3)** Si tratta di contare quanti elementi del vettore  $s$  sono diversi da zero:

```
int cardinalita (int s[])
{
    int n = 0;
    int i;
    for ( i = 0; i < 100; i++ )
        if ( s[i] )
            n++;
    return n;
}
```

In alternativa, visto che gli elementi dell'insieme sono rappresentati da degli uno nel vettore, basta sommare il contenuto del vettore.

## Esercizio 2

Vedere gli appunti di lezione.

N.B.: Tagliar corto è una prerogativa del docente: lo studente *deve* dare una risposta un po' più esauriente.

## Esercizio 3

3.1) Il programma stamperà

16 8  
8

## Esercizio 4

Cominciamo col calcolare le rappresentazioni binarie a 8 bit dei numeri 45 e 87:

$$45_{10} = 00101101_2 \quad 87_{10} = 01010111_2.$$

A questo punto effettuiamo la somma binaria (le cifre piccole rappresentano il riporto dalla colonna precedente):

$$\begin{array}{cccccccc} ^10 & ^10 & ^11 & ^10 & ^11 & ^11 & ^10 & 1 & + \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & = \\ \hline 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & . \end{array}$$

Ritradotto in base 10, il risultato è dunque

$$10000100_2 = 2^7 + 2^2 = -128 + 4 = 132,$$

che è precisamente il risultato dell'operazione  $45 + 87$  richiesta.