

# Informatica Generale A.A. 2006/2007

11 Febbraio 2008

**Esercizio 1** Si consideri la funzione logica  $f$  che preso in input un numero binario  $x$  a quattro bit dia in output il numero binario corrispondente al doppio di  $x$ . Si riduca in forma minima ciascuna componente di output della funzione  $f$ .

**Esercizio 2** Si consideri il seguente algoritmo in pseudocodice che prende in input un array  $A$  di dimensioni  $n \times n$ .

```
input (A, n*n);
for i := 1 to n do
  B[i] := A[i, i];
return B;
```

1. Che cosa calcola l'algoritmo?
2. Si modifichi l'algoritmo in modo che dia in output un array  $B$  tale che l' $i$ -esimo elemento di  $B$  sia la somma degli elementi dell' $i$ -esima riga di  $A$ .

**Esercizio 3**

1. Si consideri il seguente algoritmo che calcola il massimo di 3 numeri interi  $a, b, c$  e mette il risultato nella variabile  $d$ .

```
MAX(a, b, c)
int d;
d := a;
if (b - d) > 0 then d := b;
if (c - d) > 0 then d := c;
```

Si traduca l'algoritmo al punto sopra in ASSEMBLER usando il numero minimo di registri e le seguenti istruzioni:

- LOAD  $R_i, X$  = carica il contenuto della cella di memoria  $X$  nel registro  $R_i$
- STORE  $R_i, X$  = memorizza nella cella di memoria  $X$  il contenuto del registro  $R_i$

- SUB  $R_i, R_j, R_k$  = sottrai al contenuto del registro  $R_i$  il contenuto del registro  $R_j$  e metti il risultato in  $R_k$
  - NEG\_JUMP  $R_i E$  = se il contenuto del registro  $R_i$  è strettamente minore di zero carica l'etichetta  $E$  nel Program Counter, altrimenti passa all'istruzione successiva. [Questa istruzione ha l'effetto di saltare all'istruzione  $E$  se  $val(R_i) < 0$ .]
  - HALT = termina l'esecuzione.
2. Si supponga che: (i) gli indirizzi di memoria delle variabili  $a, b, c, d$  siano rispettivamente 0, 1, 2, 3 e che i valori assunti siano inizialmente 4, 8, 3, 0, (ii) le istruzioni del programma ASSEMBLER richiesto al punto sopra siano caricate in memoria in successione a partire dalla locazione 4, (iii) ciascuna cella di memoria contenga un'istruzione.

INDIRIZZO	CONTENTUTI
(a) 0	4
(b) 1	8
(c) 2	3
(d) 3	0
4	Istruzione 1
5	Istruzione 2
⋮	⋮

Assumendo che inizialmente il PC valga 4, scrivere il contenuto dei registri ( $R_1, R_2, \dots, PC, IR$ ) e della memoria ad ogni passo di esecuzione fino alla fine dell'esecuzione dell'istruzione HALT.

**Esercizio 4** Si fornisca una codifica di lunghezza fissa delle istruzioni ASSEMBLER dell'esercizio precedente in codice macchina che usi il numero minimo di bit, assumendo che: (i) la memoria abbia 16 locazioni, (ii) i registri siano quanti servono per il programma richiesto sopra.