

VERIFICA DI INFORMATICA (SOLUZIONE)

Liceo Scientifico "A. Volta"
classe 5° B, 29/03/2008
prof. Magni Claudio

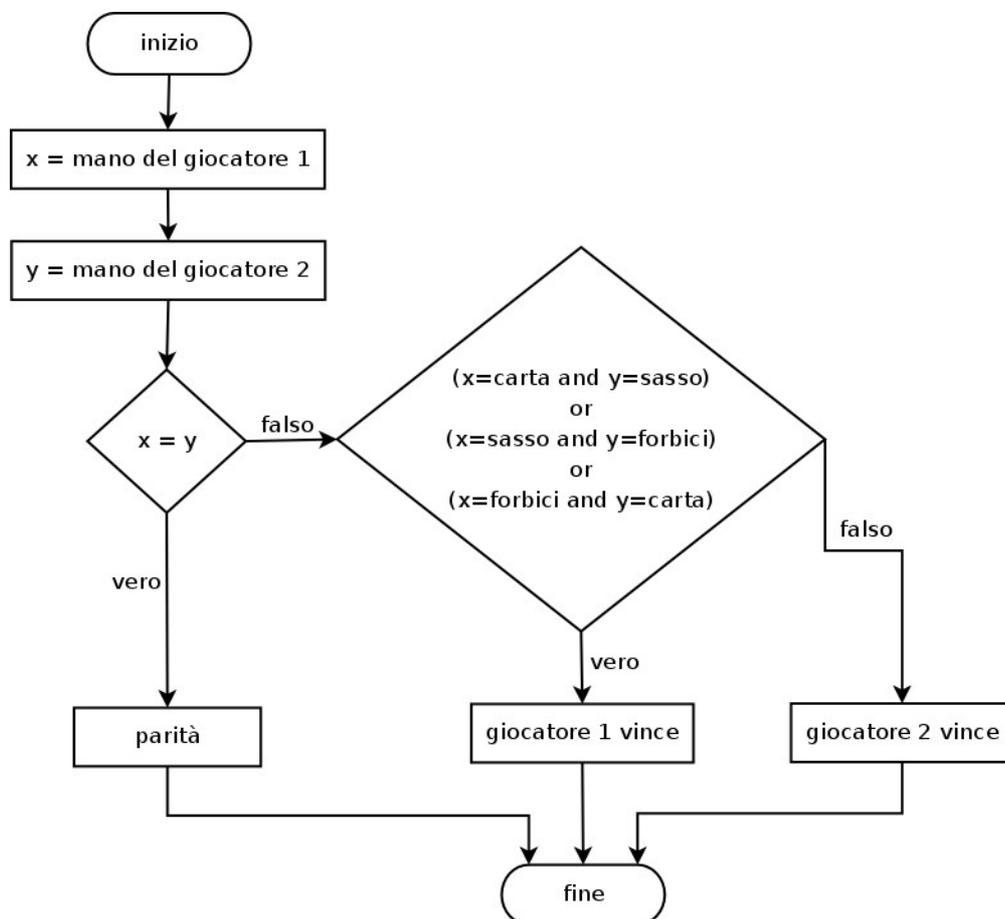
1)

154	0	↑	$154_{10} = 10011010_2$
77	1		
38	0		
19	1		
9	1		
4	0		
2	0		
1	1		
2			

2)

$$1001110_2 = 1 \cdot 2^6 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 = 64 + 8 + 4 + 2 = 78$$

3)



4)

Un ciclo di clock è composto da 3 fasi:

1. fetch
2. decode
3. execute

Nella terza fase la CPU esegue l'istruzione. Può quindi svolgere diversi compiti; uno di questi è la lettura/scrittura in memoria che abbiamo visto a lezione.

La prima fase interessa due registri della CPU: PC (Program Counter), IR (Instruction Register). Il primo registro contiene l'indirizzo della istruzione da eseguire. Il secondo contiene l'istruzione in sé (in linguaggio macchina). Le operazioni eseguite in questa fase sono:

1. viene messo sul bus indirizzi l'indirizzo contenuto nel PC
2. viene letta dalla memoria la cella indirizzata, mettendo il suo contenuto nel bus dati
3. nel IR viene copiato il contenuto del bus dati (ovvero l'istruzione)
4. il PC viene incrementato di 1, in modo da leggere l'istruzione successiva al prossimo ciclo

La seconda fase semplicemente decodifica il contenuto del IR, capisce quindi che istruzione eseguire.

5)

La formula logica della porta NAND è $\neg(A \wedge B)$.

Tabella di verità:

A	B	uscita
0	0	1
0	1	1
1	0	1
1	1	0

6)

- linguaggio formale:** definito da un insieme di simboli e da regole precise che permettono di stabilire con certezze se una sequenza di simboli appartiene o meno al linguaggio.
- linguaggio macchina:** insieme di istruzioni e dati direttamente eseguiti dalla CPU; è la rappresentazione di un programma a livello più basso.
- linguaggio di alto livello:** linguaggio formale che cerca di avvicinarsi alla lingua umana partendo dal linguaggio macchina; rispetto a quest'ultimo è più astratto e facile da usare.
- compilazione:** traduzione da un linguaggio di alto livello al linguaggio macchina.

7)

- primo quesito: #C8642D 
- verde: #00FF00 
- giallo: #FFFF00 

8)

Riporto il codice, commentando a lato ciò che viene svolto.

```
1 | int x, y;           /* dichiaro 2 variabili intere */
2 | scanf("%d", &x);   /* ricevo un numero dall'utente che metto in x */
3 | scanf("%d", &y);   /* ricevo un numero dall'utente che metto in y */
4 | if (x != y || x == y) { /* se x è diverso da y o x è uguale a y */
5 |     printf("finta. "); /* stampo su schermo "finta. " */
6 | }                  /* chiudo la parentesi dell' if */
7 | if ( (x/y) == 1 ) { /* se x diviso y è uguale a 1 */
8 |     printf("uguali"); /* stampo su schermo "uguali" */
9 | } else {           /* altrimenti */
10 |    printf("diversi"); /* stampo su schermo "diversi" */
11 | }                  /* chiudo la parentesi dell' else */
```

Si possono distinguere 2 casi:

1. Se i due numeri inseriti sono uguali l'output è: `finta. uguali`
2. Se i due numeri inseriti sono diversi l'output è: `finta. diversi`

Esiste un caso particolare in cui il programma non fornisce output ma si blocca generando un errore. È il caso in cui il secondo numero inserito (y) sia 0: infatti alla riga 7 viene fatta una divisione per zero, che è una operazione impossibile per il processore.

Notate che la condizione alla riga 4 è sempre verificata, quindi `finta.` verrà sempre stampato, dopodiché si procederà dalla riga 7.