



Politecnico di Milano

Facoltà di Ingegneria Civile, Ambientale e Territoriale

Informatica (ICA-LC) [083668] – Informatica B [079904]

Informatica B [060054]

Prof. P. Plebani

Allievi Ingegneria Civile e Ambientale

I prova in itinere (18 Novembre 2010)

| | |
|-----------------------|--|
| <i>Cognome e nome</i> | |
| <i>Matricola</i> | |
| <i>Firma</i> | |

| Domanda | 1 | 2 | 3 | 4 | 5 | 6 | |
|---------------|---|---|---|---|----|---|-----|
| Punteggio max | 4 | 4 | 6 | 4 | 12 | 2 | TOT |
| Punteggio | | | | | | | 32 |

La **durata** della **prova** è di **2 ore**. Non è consentito consultare libri o appunti, non è consentito l'uso di calcolatrici.

Scrivere solo sui fogli distribuiti utilizzando il retro delle pagine in caso di necessità e cancellando le parti di brutta con un tratto di penna. Non separare questi fogli.

Per tutti gli esercizi non è sufficiente fornire il risultato, ma è **necessario mostrare il procedimento seguito**.

Gli allievi sono invitati a privilegiare **chiarezza, proprietà di linguaggio e sinteticità** nelle risposte agli esercizi, con l'obiettivo di **dimostrare la loro conoscenza degli argomenti**.

Domanda 1. Rappresentazione di valori numerici (4 punti).

1. Dati i due numeri :

- $N1 = -31$ in base 10
- $N2 = 31$ in base 16

indicare quanti bit occorrono per rappresentarli in binario complemento a 2, e effettuare la conversione

per N1 sono sufficienti 6 bit visto che 2^{5+1} permette di rappresentare numeri da -32 a + 31.

Per $N2 = 31_{16} = 49_{10}$ servono 7 bit visto che 2^{6+1} rappresenta da -64 a + 63

Per le operazioni richieste converto tutto con 7 bit

$N1 = 0011111 \rightarrow 1100000 + 1 = 1100001$

$N2 = 0110001$

2. Eseguire in complemento a 2 (mostrando i passaggi, indicando esplicitamente se si verifica overflow e motivando la risposta) le operazioni:

- $N1 + N2$
- $N2 - N1$.

$N1 + N2 = 1100001 + 0110001 = (1)0010010 = 18_{10}$ non c'è overflow (i segni sono discordi)

$N2 - N1 = 0110001 + (-1100001) = 0110001 + 0011111 = 1010000$ c'è overflow il segno è discorde

Domanda 2. Logica booleana (4 punti).

Verificare l'uguaglianza (o la disuguaglianza) delle due seguenti espressioni

- $\text{not}(\text{not } A \text{ and not } C) \text{ and not } B$
- $A \text{ or } C \text{ and not } B$

| A | B | C | Not A | Not B | Not C | notA and not C | Not(not A and not C) | Totale |
|---|---|---|-------|-------|-------|----------------------|----------------------------|--------|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |

| A | B | C | Not B | C and not B | Totale |
|---|---|---|-------|----------------|--------|
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |

Non sono equivalenti

Domanda 3. Architettura di un calcolatore (6 punti).

Illustrare la struttura della CPU e descrivere in dettaglio (evidenziando il ruolo dei componenti della CPU interessati) il ciclo fetch-decode-execute

Domanda 4. Tracing di programmi (4 punti).

Stampare a schermo l'output del seguente programma Matlab

```
a = ones(3);
b = [1 : 3 : 9];
c = [4 ; 1 ];
d = [ 4 : -2 : 2]';
e = [5 ; 1].^2;

k = [c , d , e];
b = [b ; k];

z = size(b,2);

b = b .* (b > z);

for ii = [2,3,1]
    for jj = 3: -1 :1
        if(ii == jj)
            w = z - jj +1;
            disp(['elemento alla posizione ' , num2str(ii) , ' , ' , num2str(w) , ' = '
num2str(b(ii , w))])
        end
    end
end
```

| ii | w | b(ii,w) |
|----|---|---------|
| 2 | 2 | 4 |
| 3 | 1 | 0 |
| 1 | 3 | 7 |
| | | |
| | | |
| | | |

Domanda 5. Linguaggio Matlab (12 punti).

Scrivere un programma per gestire una squadra di calcio.

I giocatori sono individuati dal loro numero di maglia, dal 1-11. Le informazioni da registrare sono, per ogni giocatore:

- il ruolo: un carattere 'a' –attaccante ; 'c' – centrocampista ; 'd' – difensore; 'p' portiere
- il numero di gol fatti (o subito nel caso del portiere)

1. Si scriva un programma per inserire i dati (i.e. ruolo e numero di gol) per tutti i giocatori dall'1-11. I dati saranno salvati in due vettori: **goal** e **ruoli**

2. Si calcoli la media di goal fatti nella squadra (ad esclusione del portiere)

3. Si calcoli il numero massimo di goal fatti da un giocatore nella squadra. Si stampino quindi le informazioni di tutti i giocatori che hanno fatto quel numero di goal (potrebbero esserci più giocatori che hanno fatto il massimo numero di goal).

4. Si aggiungano le informazioni riguardanti il valore di ingaggio espresse in migliaia di euro. Si inseriscano tali informazioni in un vettore denominato **valore**. Si calcoli quindi il miglior rapporto (numero di goal fatti)/(valore del giocatore) e si stampino le informazioni di tutti i giocatori che hanno tale rapporto.

5. Si supponga che in un file "valori.txt" siano memorizzati i valori (in migliaia di euro) dei giocatori, nel formato "num giocatore, valore". Es:

1, 330

2, 450

3, 2

Si indichi dove e come modificare il codice al punto 4 in modo che il vettore **valore** sia riempito con i dati provenienti dal file.

N.B.: Si supponga che il portiere abbia la maglia nr 1. E si assuma che l'inserimento dei dati sia sempre corretto, e che non occorra nessuna forma di controllo sull'input.

Non si possono utilizzare funzioni mean, sum, max.

Esempio di output

il numero di goal medio della squadra è 22

il numero di goal massimi fatti da un giocatore è 22

il giocatore con maglia 2 e ruolo a ha fatto 22 goal

il giocatore con maglia 3 e ruolo d ha fatto 22 goal

il miglior rapporto goal/costo della squadra è 22

il giocatore con maglia 3 e ruolo d ha fatto 22 goal. Vale 2 mila euro. Rapporto goal/costo è 11

```

N = 3;
goal = zeros(1 , N);
valore = zeros(1 , N);
ruolo = [];

for ii = 1 : N
    ruolo(ii) = input(['si inserisca il ruolo del giocatore nr ', num2str(ii) , ' ']);
    goal(ii) = input(['si inserisca il numero di goal fatti/subiti dal giocatore nr ',
num2str(ii) , ' ']);
    valore(ii) = input(['si inserisca il valore in migliaia di euro del giocatore nr ',
num2str(ii) , ' ']);
end

m = goal(1);
s = 0;

for ii = 2 : N
    if (goal(ii) > m)
        m = goal(ii);
    end
    s = s + goal(ii);
end

ngoalMedio = s/(N - 1);

disp(['il numero di goal medio della squadra è ' , num2str(ngoalMedio)]);
disp(['il numero di goal massimi fatti da un giocatore è ' , num2str(m)]);

maxGoalIndexes = find(goal == m);

for ii = maxGoalIndexes
    disp(['il giocatore con maglia ' , num2str(ii) , ' e ruolo ' , ruolo(ii) , ' ha fatto '
, num2str(goal(ii)) , ' goals']);
end

rapporto = goal./valore;

r = 0;

for ii = 2 : N
    if (rapporto(ii) > r)
        r = rapporto(ii);
    end
end

disp(['il miglior rapporto goal/costo della squadra è ' , num2str(r)]);

maxRapporto = find(rapporto== m);

for ii = maxRapporto
    disp(['il giocatore con maglia ' , num2str(ii) , ' e ruolo ' , ruolo(ii) , ' ha fatto '
, num2str(goal(ii)) , ' goals. Vale ' , num2str(valore(ii)) , 'mila euro rapporto ' ,
num2str(rapporto(ii))]);
end

```


Domanda 6. Domanda extra (2 punti).

Si consideri un elaboratore a 32 bit e uno a 64 bit. Per ognuno calcolare quanti GigaByte al massimo è possibile indirizzare?