



Politecnico di Milano

Facoltà di Ingegneria Civile, Ambientale e Territoriale

**Informatica (ICA-LC) [091456] – Informatica B [079904]**

**Informatica B [060054]**

**Prof. P. Plebani**

**Allievi Ingegneria Civile e Ambientale**

**I prova in itinere (18 Novembre 2011)**

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

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

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 :

- $N1 = -20$  in base 10
- $N2 = 110$  in base 2 (complemento a 2)
- $N3 = 2A$  in base 16

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

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

- $N1 + N2$
- $N1 + N3$ .

**Domanda 2. Logica booleana (4 punti).**

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

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

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

Dopo aver definito il ruolo della memoria centrale secondo la macchina di von Neumann, illustrare la struttura della memoria centrale sottolineando la relazione che esiste tra tale struttura e il bus degli indirizzi e dei dati

#### Domanda 4. Tracing di programmi (4 punti).

Ipotizzando di eseguire il codice matlab di seguito riportato, indicare l'output nel momento in cui vengono eseguite le istruzioni in grassetto

```
a = 1:10
b = [10:-1:1]
c = [ b(find(b>5)); find(a<=5)]
d = [];
i=1;
while (i<=size(c,2))
    if (c(1,i)+c(2,i) > 10)
        d = [d 0]
    else
        d = [d c(1,i)+c(2,i)]
    end
    i = i+1;
end
```

```
a =
    1    2    3    4    5    6    7    8    9   10

b =
   10    9    8    7    6    5    4    3    2    1

c =
   10    9    8    7    6
    1    2    3    4    5

d = 0
d =
    0    0

d =
    0    0    0

d =
    0    0    0    0

d =
    0    0    0    0    0
```

**Domanda 5. Linguaggio Matlab (12 punti).**

Partendo da una matrice **A** di dimensione **10x10** la si valorizzi utilizzando il linguaggio Matlab con numeri reali casuali compresi tra -5 e +5.

Si scriva quindi il codice in grado di eseguire le seguenti operazioni

- generazione di un vettore colonna **S** che contiene per ogni elemento la somma degli elementi contenuti su ciascuna delle righe di **A**
- conteggio degli elementi del vettore **S** ottenuto il cui valore è maggiore di 10 o minore di -10

```
A=(rand(10)*10)-5;
```

```
S=[];
```

```
for i=1:10
```

```
    S=[S; sum(A(i,:))];
```

```
end
```

```
C=[find(S>10 | S<-10)]
```

```
sum(S(C))
```

```
v = input('inserisci valore');
```

```
risultato = scopri(A, v);
```

```
if (risultato ==-1)
```

```
    disp('errore');
```

```
else
```

```
    risultato
```

```
end
```

Definire inoltre una funzione **scopri** che riceva in ingresso una matrice **B** di dimensione 10x10 e uno scalare **C**:

- controlla che il valore di **C** sia minore o uguale a 5. In caso contrario ritorna al programma chiamante il valore -1
- cerca la posizione di tutti i valori di **B** che risultano essere minori di **C** o maggiori di  $-C$
- somma i valori nelle posizioni trovate precedentemente e restituisce il valore ottenuto al programma chiamante

Si definisca inoltre il codice Matlab che chiama la funzione **scopri** per la matrice **A** definita nella prima parte dell'esercizio e uno scalare inserito dall'utente. In caso la funzione ritorni il valore -1 il programma visualizzerà un messaggio di errore

```
function r=scopri(B,C)
    if (C>5)
        r = -1;
    else
        X=find(B<C & B>-C);
        r = sum(B(X));
    end
end
```



**Domanda 6. Domanda extra (2 punti).**

Illustrare brevemente il motivo per cui un programma può essere eseguito solo su calcolatori di una determinata famiglia (ad esempio Intel) e non è compatibile con calcolatori basati su architettura differente (ad esempio Motorola)