



Politecnico di Milano

Facoltà di Ingegneria Civile, Ambientale e Territoriale

Informatica [097256] [091461] [091456]

Prof. G. Boracchi

Allievi Ingegneria Civile e Ambientale

Il prova in itinere (25 Novembre 2016)

Cognome e nome	
Matricola	
Firma	

Domanda	1	2	3	4	5	TOT
Punteggio max	2	6	6	12	6	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. Non dilungarsi però in spiegazioni prolisse.**

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. Macchina di Von Neumann (2 punti).

Si descriva brevemente la memoria centrale nell'architettura della macchina di Von Neumann ed il suo ruolo

Domanda 2. Matlab: interpretazione del codice (6 punti).

2a) Date le seguenti funzioni

```
function [a,b] = funz_b(x)
    b = floor(x(end) / 2);
    a = funza(x,b);
```

```
function k = funz_a(x, w)
k = [];
for y = x
    if(y > w(1))
        k = [k, y];
    end
end
```

si descriva brevemente il funzionamento di ogni funzione e dica che messaggi vengono visualizzati a schermo dalle seguenti istruzioni:

```
[p,w] = funz_b([10, 30, 21]);
disp(p)
disp(w)
```

2b) Disegnare quello che viene visualizzato a schermo dai seguenti comandi (si utilizzino gli elementi grafici corrispondenti).

```
x = [0 : 1 : 10];
y = 1/3 * x;

z = [10 : -2 : 0];
k = 3^(-1) * z - 1;
```

```
[~, b] = max(x);
```

```
figure(),
plot(x, y, '-')
hold on
plot(z, k, 'x')
plot(x(b), y(b), 'o');
hold off
title('il mio plot')
axis equal
```

SOLUZIONE

2a)

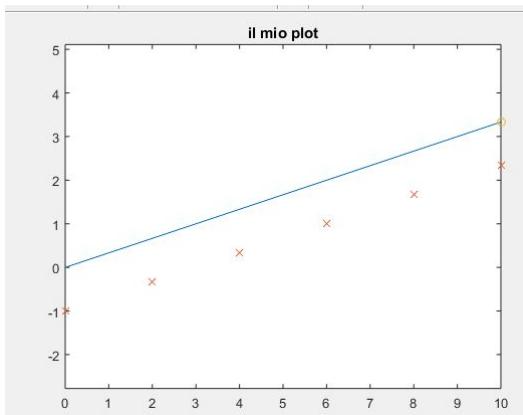
funz_b un invoca funz_a passando come primo parametro attuale x (che è il parametro formale in ingresso) e la parte intera della metà dell'ultimo elemento di x.

funz_a prende in ingresso due parametri formali (x e w) e restituisce un vettore contenente tutti gli elementi di x maggiori di w

```
>> disp(p)
30 21
```

```
>> disp(w)
10
```

2b)



Domanda 3. Matlab (6 punti).

Si scriva una funzione *divisoneSenzaDiviso* che esegue la divisione intera tra due numeri (dividendo e divisore) e restituisce la parte intera ed il resto della divisione (ad esempio $7 / 2 = 3 + 1$, $14 / 7 = 2 + 0$).

Come da nome della funzione *divisoneSenzaDiviso* non può utilizzare divisioni, moltiplicazioni, la funzione mod, funzioni di approssimazione (e.g. round), ma solamente addizioni e sottrazioni.

Si gestisca correttamente il caso in cui il resto della divisione sia zero.

hint:

per calcolare la parte intera di x / y è possibile contare quante volte è possibile sommare y ad un accumulatore fino a raggiungere x (o sottrarre y ad x fino a raggiungere zero)

SOLUZIONE

```
function [r, m] = divisioneSenzaDiviso(a, b)
```

```
    r = 0;
```

```
    tot = 0;
```

```
    while(tot <= a)
```

```
        tot = tot + b;
```

```
        r = r + 1;
```

```
    end
```

```
    r = r - 1;
```

```
    m = a - (tot - b);
```

Domanda 4. Matlab (12 punti).

Si scriva una funzione *analizzaVoti* che prende in ingresso tre vettori

- *primo*: un vettore contenente le valutazioni del primo compito (interi tra 0 e 32)
- *secondo*: un vettore contenente le valutazioni del secondo compito (interi tra 0 e 32)
- *matricole*: un vettore contenenti i numeri di matricola di tutti gli studenti valutati

Si assuma che i tre vettori abbiano la stessa lunghezza e che siano stati ordinati in modo che la posizione *i*-sima di tutti i vettori faccia riferimento allo stesso studente.

La funzione restituisce:

- un vettore contenente i voti di tutti gli studenti promossi (per passare l'esame occorre avere almeno 18 in entrambe le prove, ed in tal caso il voto è dato dalla media dei due compiti)
- i numeri di matricola di tutti gli studenti promossi
- una variabile booleana che dice se tutti gli studenti sono stati promossi
- il voto medio degli studenti promossi

in uno script separato si definiscano

- un vettore A di 16 numeri scelti a caso tra 0 e 32
- un vettore B di 16 numeri scelti a caso tra 6 e 32
- un vettore M contenente 16 matricole a partire dal 102456

si invochi la funzione *analizzaVoti* passando come argomenti i vettori sopra definiti e si stampi a schermo un messaggio analogo al seguente, sulla base dei valori restituiti dalla funzione.

"sono stati promossi tutti gli studenti" (oppure "non sono stati promossi tutti gli studenti")

"La media dei promossi è: 23.3."

"Ecco le matricole dei promossi:

102456

102466

102467

...

"

SOLUZIONE

```
function [votiPromossi, matricolePromossi, tuttiPromossi, mediaPromossi] =  
analizzaVoti(primo, secondo, matricola)
```

```
votiPromossi = [];
```

```
matricolePromossi = [];
```

```
tuttiPromossi = true;
```

```
for ii = 1 : numel(primo)
```

```
    if primo(ii) >= 18 && secondo(ii) >= 18
```

```
        votiPromossi = [votiPromossi, mean([primo(ii), secondo(ii)])];
```

```
        matricolePromossi = [matricolePromossi, matricola(ii)];
        tuttiPromossi = false;
    end
end
```

```
mediaPromossi = mean(votiPromossi);
```

```
A = floor(33* rand(1, 16));
B = 6 + floor(27 * rand(1, 16));
M = [102456 : 1 : 102471]
```

```
[mediaPromossi, matricolePromossi, flag, mediaPromossi] = analizzaVoti(A,B,M)
```

```
if flag
    disp('tutti gli studeti sono stati promossi')
else
    disp('non tutti studenti sono stati promossi')
end
```

```
fprintf('\nla media dei promossi è %2.2f', mediaPromossi);
disp('ecco le matricole dei promossi')
```

```
for ii = 1 : numel(matricolePromossi)
    disp(matricolePromossi(ii))
end
```


Domanda 5. Aritmetica Binaria (6 punti).

Eseguire le seguenti conversioni di base tra numeri positivi:

$$(27376)_8 \Rightarrow (????)_2$$

$$(110111)_2 \Rightarrow (????)_{16}$$

$$(5ABF)_{16} \Rightarrow (????)_8$$

Si definisca m come il minor numero di bit in grado di rappresentare in CP2: - 56 e - 19

Si rappresentino in CP2 - 56 e - 19 utilizzando m bit.

Si eseguano quindi le seguenti operazioni indicando l'eventuale presenza di carry e di overflow.

I. $56 - 19$

II. $+ 56 + 19$

SOLUZIONI

$$(27376)_8 \Rightarrow (????)_2$$

$$\text{Sol : } 010\ 111\ 011\ 111\ 110_2$$

$$(2)\ (7)\ (3)\ (7)\ (6)_2$$

$$(110111)_2 \Rightarrow (????)_{16}$$

$$\text{Sol : } 0011\ 0111_2$$

$$(3)\ (7)_{16}$$

$$(5ABF)_{16} \Rightarrow (????)_8$$

$$\text{Sol : } 0101\ 1010\ 1011\ 1111_2$$

$$5\ 5\ 2\ 7_8$$

Si definisca m come il minor numero di bit in grado di rappresentare in CP2: - 56, - 19

SOL:

N1 = -56 : servono $m=7$ bit $-2^m < N1 < -2^{m-1}$

N2 = -19 : servono $m=6$ bit $-2^m < N2 < -2^{m-1}$

Si rappresentino in CP2 - 56 e - 19 utilizzando m bit.

SOL:

$$N1=-56 \rightarrow 1001000\ \text{CP2}$$

$$N2=-19 \rightarrow 101101\ \text{CP2}$$

Si eseguano le seguenti operazioni indicando l'eventuale presenza di carry e di overflow.

III. 56 - 19

SOL:

$$\begin{array}{r} 56 + \quad 0\ 1\ 1\ 1\ 0\ 0\ 0\ + \\ -19 = \quad 1\ 1\ 0\ 1\ 1\ 0\ 1\ = \\ \hline 37 \quad (1)\ 0\ 1\ 0\ 0\ 1\ 0\ 1 \\ \text{CARRY} \end{array}$$

IV. + 56 + 19

SOL:

$$\begin{array}{r} 56 + \quad 0\ 1\ 1\ 1\ 0\ 0\ 0\ + \\ 19 = \quad 0\ 0\ 1\ 0\ 0\ 1\ 1\ = \\ \hline 75 \quad (0)\ 1\ 0\ 0\ 1\ 0\ 1\ 1 \\ \text{OVERFLOW} \end{array}$$

