



Politecnico di Milano

## Dipartimento di Elettronica, Informazione e Bioingegneria

**Informatica A - a.a 2017/2018 - 30 Agosto 2018**

Cognome: \_\_\_\_\_ Matricola: \_\_\_\_\_  
Nome: \_\_\_\_\_ Firma: \_\_\_\_\_

### Istruzioni

- Non separate questi fogli. Scrivete la soluzione **solo sui fogli distribuiti**, utilizzando il retro delle pagine in caso di necessità. **Cancellate le parti di brutta** con un tratto di **penna**.
- Ogni parte non cancellata a penna sarà considerata parte integrante della soluzione.
- **NON è possibile scrivere a matita.**
- È **vietato** utilizzare **calcolatrici, telefoni o pc**. Chi tenti di farlo vedrà **annullata** la sua prova.
- **Non è ammessa la consultazione di libri e appunti.**
- Qualsiasi **tentativo** di comunicare con altri studenti comporta **l'espulsione** dall'aula.
- È possibile ritirarsi senza penalità.
- Non è possibile lasciare l'aula conservando il tema della prova in corso.
- Tempo a disposizione: **2h30m**

### Valore indicativo degli esercizi, voti parziali e voto finale:

Esercizio 1	3 punti	_____
Esercizio 2	3 punti	_____
Esercizio 3	5 punti	_____
Esercizio 4	6 punti	_____
Esercizio 5	11 punti	_____
<b>Totale(28)</b>		_____

**Esercizio 1 - Algebra di Boole, Aritmetica Binaria, Codifica delle Informazioni (3 punti)**

- (a) Si costruisca la tabella di verità della seguente espressione booleana in tre variabili, badando alla precedenza tra gli operatori logici. Eventualmente si aggiungano parentesi. Non si accetteranno soluzioni senza il procedimento. (1 punto)

(A or B and !C) and A or !C

**Risposta:**

A	B	C	OUT
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

OUT = A + !C

- (b) Si stabilisca il minimo numero di bit sufficiente a rappresentare in complemento a due i numeri A = -33dec e B = 95, li si converta, se ne calcolino la somma (A+B) e la differenza (A-B) in complemento a due e si indichi se si genera riporto sulla colonna dei bit più significativi e se si verifica overflow. Non si accetteranno soluzioni senza il procedimento. (1 punto)

**Risposta:**

95d = 01011111b - 7bit + 1 di segno = 8bit

33d = 0100001b - 6 bit + 1 segno

cp2 = 1011111b - cp2 8bit = 11011111

A + B = -33 + 95 = 62 = 00111110b con riporto perduto

**NOME e COGNOME:** \_\_\_\_\_

A-B = -128 = 10000000b c'è riporto perduto;  
8 bit va da -128 a 127

- (c) Si converta il numero 25.3125 in virgola fissa e in virgola mobile con codifica IEEE 754, sapendo che  $1/2 = 0.5$ ,  $1/4 = 0.25$ ,  $1/8 = 0.125$ ,  $1/16 = 0.0625$ ,  $1/32 = 0.03125$ ,  $1/64 = 0.015625$ , e  $1/128 = 0.0078125$ . Non si accetteranno soluzioni senza il procedimento. (1 punto)

**Risposta:**

5bit + segno

25d = 011001

$0.3125 = 1/4 + 1/16 = 0.0101b$

25.3125 = 011001.0101 j- Virgola fissa

segno = 0

mantissa = 1.10010101 non normalizzata

exp = 4 + 127 = 131d = 10000011b

IEEE754

segno (1bit) = 0

esponente (8bit) = 10000011

mantissa (23bit) = 100 1010 1000 0000 0000 0000

**Esercizio 2 - Domanda di teoria (3 punti)**

- (a) Che differenza c'è tra una struttura dati statica ed una struttura dati dinamica? Perché vengono usate? Quali sono i vantaggi/svantaggi?

**Risposta:**

- (b) Fare un esempio di implementazione di struttura dati statica e dinamica per la memorizzazione di numeri interi, riportando per ognuna la funzione di inserimento di un nuovo dato.

**Risposta:**

**Esercizio 3 - Comprensione del Codice (5 punti)**

Scrivere cosa stampa a video il seguente codice:

```
#include <stdio.h>
float doSomething(int val, int d);

int main(){
    int a;
    int b[10];
    int *c;

    a = 2;
    c = b;

    for (c; c<(&b[0])+10; c++)
    {
        *c = (3*(c-b));
    }

    for (int i=0; i<10; i++)
    {
        printf("%d \t %.1f\n",b[i],doSomething(b[i], a));
    }
    return 0;
}

float doSomething(int val, int d){
    if (val%9==0)
        return (float)(val/d);
    else
        return (val/(float)d);
}
```

printf:

_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

**Risposta:**

0 0.0  
3 1.5  
6 3.0

**NOME e COGNOME:** \_\_\_\_\_

9 4.0  
12 6.0  
15 7.5  
18 9.0  
21 10.5  
24 12.0  
27 13.0

**Esercizio 4 - Matlab (6 punti)**

Scrivere il codice Matlab che restituisca i valori richiesti.  
Attenersi al numero massimo di righe di codice indicato.

- (a) Creare la matrice A di 5 righe e 6 colonne contenente numeri interi casuali da 10 a 30 (max 1 riga, 1 punto)

**Risposta:**

```
A = randi([10 30], 5,6)
```

- (b) Aggiungere una riga dopo la prima, contenente numeri i ordine da 1 a 6 (max 1 riga) (1 punto)

**Risposta:**

```
A = [A(1,:); [1:6]; A(2:end,:)]
```

oppure

```
A = vertcat(A(1,:), [1:6], A(2:end,:))
```

- (c) Eliminare le colonne con media inferiore alla media totale della matrice (max 1 riga) (1 punto).

**Risposta:**

```
A(:,mean(A,1)<mean(mean(A)))=[]
```

- (d) Calcolare il vettore Z contenente le radici quadrate delle medie delle righe (max 1 riga) (1 punto).

**Risposta:**

```
Z = sqrt(mean(A,2))
```

- (e) Scrivere una funzione che, data in ingresso la matrice A, dia in uscita due vettori contenenti la posizione X e Y dei valori pari della matrice A. (max 12 righe) (2 punti)



**Risposta:**

```
function [X,Y] = trova(A)
X = []
Y = []
for i=1:1:size(A,1)
    for j=1:1:size(A,2)
        if (mod(A(i,j),2)==0)
            X = [X, i];
            Y = [Y, j];
        end
    end
end
end
end
```

### Esercizio 5 - Programmazione C (10 punti)

Si ipotizzi di volere scrivere un programma C per la gestione di un negozio online di alimentari etnici. Il programma utilizza due strutture dati dinamiche.

Nella prima struttura viene immagazzinata mediante una lista l'anagrafica della clientela. Ciascun cliente è dotato di un nome, un cognome, un codice fiscale (alfanumerico), e una data di nascita. All'interno di questa lista i clienti sono ordinati alfabeticamente per cognome e poi per nome. Se due persone si chiamano allo stesso modo sono ordinati per codice fiscale (univoco).

Nella seconda struttura viene immagazzinato mediante una coda lo storico degli acquisti fatti attraverso il sistema. Ciascun acquisto è caratterizzato da un codice univoco, una data di acquisto, il codice fiscale del cliente che ha effettuato l'acquisto, da una descrizione dei contenuti dell'ordine da un totale in euro.

Si definiscano le strutture dati necessarie ad implementare le due liste del programma. (1 punto)

Si implementi una funzione che dati due clienti li confronta e dice quale dei due deve venire prima nell'anagrafica. La funzione restituisce -1 se il primo cliente deve venire prima del secondo, 0 se i due clienti sono identici (e quindi hanno lo stesso codice fiscale), oppure 1 se il primo cliente deve venire dopo il secondo. (1 punto)

Si implementi una funzione di inserimento ordinato di un cliente all'interno dell'anagrafica. La funzione deve restituire 1 se l'inserimento è andato a buon fine, oppure 0 se c'è stato un problema (ad es. il cliente esisteva già). (2 punti)

Si implementi le funzioni di push e pop per la coda (first in - first out) contenente lo storico degli acquisti fatti. (2 punti)

Implementare una funzione che chiede all'utente (mediante printf e scanf) di inserire il nome e il cognome di un cliente. Se ci sono più clienti con quel nome e cognome la funzione deve stampare a video un elenco dei codici fiscali di questi clienti e chiedere all'utente (sempre mediante printf e scanf) di sceglierne uno. A quel punto la funzione deve procedere a stampare a video (i) il numero di acquisti fatti da quel cliente nell'anno solare 2018 e (ii) il totale in euro speso dal cliente nell'anno solare 2018. Deve inoltre stampare a video, per ciascun acquisto fatto nel 2018, la data di acquisto, la descrizione dell'acquisto e il totale dell'acquisto. (4 punti)

**Risposta:**

**NOME e COGNOME:** \_\_\_\_\_

**NOME e COGNOME:** \_\_\_\_\_