

	Politecnico di Milano Facoltà di Ingegneria Industriale <b>INFORMATICA B</b> Appello del 16 Settembre 2011		COGNOME E NOME
	RIGA	COLONNA	MATRICOLA
			<i>Spazio riservato ai docenti</i>
			<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>

- Il presente plico contiene 3 esercizi e deve essere debitamente compilato con cognome e nome, numero di matricola, posizione durante lo scritto (comunicata dal docente).
- Il tempo a disposizione è di 1 ora e mezza.
- 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** (o ripudiate) con un tratto di **penna**.
- Ogni parte non cancellata a penna sarà considerata parte integrante della soluzione.
- **È possibile scrivere a matita** (e non occorre ricalcare al momento della consegna!).
- È **vietato** utilizzare **calcolatrici, telefoni o pc**. Chi tenti di farlo vedrà **annullata** la sua prova.
- È ammessa la consultazione di **libri e appunti**, purché con pacata discrezione e senza disturbare.
- 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.
- Per l'ammissione all'esame orale è necessario aver almeno impostato sufficientemente **entrambi** gli esercizi 1 e 2.
- L'esame orale è parte integrante dell'esame e deve essere realizzato almeno sufficientemente per il superamento dell'esame complessivo.

## Esercizio 1 (10 punti)

Si considerino le seguenti dichiarazioni:

```
typedef struct {
    float x;
    float y;
} punto;

typedef struct {
    punto a;
    punto b;
} segmento;

segmento dati[100];
segmento s;
segmento ris[100];
int num_coincidenti;
```

dove il tipo di dato `punto` rappresenta un punto nel piano cartesiano, il tipo di dato `segmento` rappresenta un segmento con estremi i punti `a` e `b`, e l'array `dati` contiene le informazioni di 100 segmenti nel piano cartesiano.

Si assuma che l'array `dati` sia già stato opportunamente riempito in una porzione di programma di cui non viene richiesto lo sviluppo. Scrivere un frammento di codice in linguaggio C che:

1. Acquisisca da tastiera e memorizzi nella variabile `s` le informazioni relative a un segmento.
2. Inserisca nell'array `ris` tutti i segmenti presenti in `dati` che sono coincidenti con quello inserito.
3. Assegni alla variabile `num_coincidenti` il numero di segmenti coincidenti trovati.

Si ricorda che due segmenti si dicono coincidenti quando entrambi i loro punti estremi, **indipendentemente dall'ordine**, hanno le stesse coordinate cartesiane.

**Note.** Riportare soltanto il frammento di programma richiesto dichiarando le eventuali variabili aggiuntive necessarie.



## Esercizio 2 (10 punti)

Sia data la seguente definizione di funzione matematica di *media mobile a finestra* di dimensione  $m$ :

$$y(x, i, m) = \frac{1}{\min(i, m)} \sum_{1 \leq k \leq \min(i, m)} x(i - k + 1)$$

*Breve spiegazione:*

La funzione calcola la media di  $m$  numeri del vettore  $x$  a partire da  $i-m+1$  fino all'indice  $i$ . Se l'indice  $i$  è minore di  $m$ , allora si prende come dimensione minima della finestra il valore minimo fra  $i$  ed  $m$ .

Ad esempio, per un generico vettore  $x$  e un valore di  $m$  pari a 3 si ha:

$$y(x, 1, 3) = x(1)$$

$$y(x, 2, 3) = (x(1) + x(2)) / 2$$

$$y(x, 3, 3) = (x(1) + x(2) + x(3)) / 3$$

$$y(x, 4, 3) = (x(2) + x(3) + x(4)) / 3$$

$$y(x, 5, 3) = (x(3) + x(4) + x(5)) / 3$$

.. e così via.

1. Si scriva una funzione MATLAB *moving\_average* che riceve in ingresso un vettore  $x$ , il valore dell'indice  $i$  e la dimensione della finestra  $m$  e restituisca il valore di  $y(x, i, m)$ . Nella soluzione non è possibile utilizzare funzioni per il calcolo della media di MATLAB.
2. Si scriva una funzione *confronta* con due parametri in ingresso:
  - un vettore  $z$ ;
  - un intero  $m$ .La funzione calcola un vettore *mov\_avg* applicando la funzione *moving\_average* con finestra  $m$  al vettore  $z$  per valori dell'indice  $i$  che variano da 1 a  $length(z)$  e restituisce gli indici di  $z$  in cui  $z$  è maggiore del vettore *mov\_avg*.
3. Si scriva un esempio di invocazione della funzione *confronta*.



### **Esercizio 3** (6 punti)

1. Si determini, giustificando la risposta, la codifica binaria del numero 1073741825 come naturale a 32 bit sapendo che tale numero si può scrivere come somma di  $2^{30}+1$ .
2. Si converta tale numero nello standard in virgola mobile IEEE-754 a precisione singola riportando i calcoli effettuati. Si ricorda che la precisione singola prevede 1 bit di segno, 8 bit di esponente e 23 bit di mantissa.
3. Tenendo presente le risposte fornite nei punti precedenti, si dica se il valore del numero codificato come naturale è lo stesso della sua conversione in virgola mobile a precisione singola. In caso di risposta negativa si dica quale dei due ha il valore maggiore. Si giustifichi adeguatamente la risposta.