



**Informatica A – a.a. 2012/2013 – 11/09/2013**

Cognome \_\_\_\_\_ Matricola \_ \_\_\_\_\_

Nome \_\_\_\_\_ Firma dello studente \_\_\_\_\_

**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** (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.
- **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: **2h**

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

**Esercizio 1 ( 5 punti )**

**Esercizio 2 ( 4 punti )**

**Esercizio 3 ( 5 punti )**

**Esercizio 4 ( 7 punti )**

**Esercizio 5 ( 7 punti )**

**Totale: ( completo 28 punti )**

### Esercizio 1 - Algebra di Boole, Aritmetica Binaria, Codifica delle Informazioni

(a) Si costruisca la tabella di verità della seguente espressione booleana in tre variabili, badando alla precedenza tra gli operatori logici. Eventualmente si aggiungano le parentesi (1 punto).

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

(b) Si stabilisca il minimo numero di bit sufficiente a rappresentare in complemento a due i numeri  $A = 17$  e  $B = -47$ , 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 (2 punti).

- (c) Si converta il numero 24.875 in virgola fissa e in virgola mobile ( $r = m \times 2^n$ , con  $m$  e  $n$  codificati in binario, 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  $1/128 = 0.0078125$ ) (1 punto)

## **Esercizio 2 - Domanda di teoria**

Spiegare cosa sono i puntatori e a cosa servono. Si faccia utilizzo di semplici esempi.

### Esercizio 3 - Matlab

Risolvere i seguenti problemi utilizzando meno righe di codice possibile (ottimo se solo una riga):

1. Creare una matrice di 5 colonne e 4 righe contenente tutti valori pari a 3;
2. Modificare la seconda riga in modo che contenga numeri da 1 a 5
3. Modificare le colonne dispari in modo che contengano valori tutti pari a 10
4. Modificare i valori delle celle contenenti valori dispari ponendole uguale a 0
5. Calcolare la radice quadrata della somma di tutte le celle della matrice

Dato il vettore  $T = [32 \ 9 \ -2 \ 42 \ 23 \ -11 \ -15 \ 40 \ 18 \ 16 \ 28]$  contenente le temperature misurate da un sensore, scrivere il codice matlab che:

1. calcoli la media delle temperature
2. calcola la media delle temperature superiori lo 0
3. il sensore è starato, sottrarre due gradi a tutte le temperature inferiori a 0

## Esercizio 4 - Programmazione C

Si supponga di avere a disposizione la seguente struttura dati per gestire viaggi effettuati in treno.

```
#define N 10
#define F 4

typedef struct stazione {
    int lat;
    int long;
    char nome[N]
    struct stazione* prox;
} tappa;
```

La struttura dati rappresenta una stazione del treno all'interno di un itinerario. Ogni stazione ha un nome, una posizione (lat, long) e un puntatore alla prossima stazione dell'itinerario.

Definire e implementare:

1. una funzione che permette di creare itinerari mediante l'aggiunta di un dato di tipo tappa a una lista appositamente creata.
2. una funzione che calcola la lunghezza di un itinerario. Si ipotizzi che il tragitto tra una tappa e la sua successiva venga sempre fatta lunga una linea retta.

## Esercizio 5 - Programmazione in C

Si ipotizzi di dovere immagazzinare le letture realizzate da un sensore di temperatura. Una lettura è composta da l'orario in cui è stata effettuata (HH:MM:SS), una stringa che indica il locale in cui è stata effettuata la lettura (cucina, salotto, camera1, camera2, ecc.), e il valore stesso della lettura in gradi Celsius.

**L'immagazzinamento delle letture deve avvenire mediante una lista concatenata, ordinata per orario di lettura.**

**Si definiscano le strutture dati necessarie allo sviluppo di questo programma.**

Inoltre, definire e implementare:

1. una funzione per aggiungere una lettura alla lista

ptrLista aggiungiLettura(ptrLista testa, int ora, int minuto, int secondo, char \*locale, int valore)

2. una funzione che riceve due orari e elimina dalla lista tutte le letture avvenute in quella finestra temporale. Si ipotizzi che il primo orario sia "prima" del secondo orario.

ptrList rimuoviLettture(int daOra, int daMinuto, int daSecondo, int aOra, int aMinuto, int aSecondo)