



Informatica A – a.a. 2012/2013 – 16/11/20102

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** (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 (2 punti) _____

Esercizio 2 (2 punti) _____

Esercizio 3 (2 punti) _____

Esercizio 4 (4 punti) _____

Esercizio 5 (4 punti) _____

Totale: (14 punti) _____

Esercizio 1 - Algebra di Boole, Aritmetica Binaria, Codifica delle Informazioni (2 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 le parentesi. Non si accetteranno soluzioni senza il procedimento.

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

(b) Si stabilisca il minimo numero di bit sufficiente a rappresentare in complemento a due i numeri $A = 48_{\text{dec}}$ e $B = -33_{\text{dec}}$, 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.

(c) Si converta il numero 67.625 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 $1/128 = 0.0078125$). Non si accetteranno soluzioni senza il procedimento.

Esercizio 2 - Architettura del calcolatore (2 punti)

Si descriva brevemente cosa si intende per memorie di tipo RAM e di tipo ROM, e si indichi a cosa servono.

Esercizio 3 - Interpretazione del codice (2 punti)

Completare il seguente codice con il proprio nome (variabile m) e la propria matricola (variabile n). Indicare cosa viene stampato a video dal programma. Spiegare "passo per passo" cosa fa ciascuna istruzione.

```
#include <stdio.h>
#include <string.h>
#define MAX 10
```

```
int main()
{
    char m[MAX] = "_____";
    int n = _____;
    int o, p, f, i;
    char j;
```

```
    printf("m: %s\n", m);
    printf("n: %d\n", n);
```

```
    p = n%10000/1000;
    printf("p: %d\n", p);
```

```
    o = p%3;
    printf("o: %d\n", o);
```

```
    switch (o) {
        case 0:
            f = 20;
            break;
        case 1:
            f = 22;
            break;
        default:
            f = 24;
            break;
    }
```

```
    printf("f: %d\n", f);
```

```
    for (i=0; i<strlen(m); i++) {
        if (m[i] >= 'a' && m[i] <= 'z') {
            m[i] = m[i] - ('a'-'A');
        }
    }
    printf("m: %s\n", m);
```

```
    for (i=0; i<strlen(m); i++)
        j = m[i] + f;
        if (j>'Z') {
            j = 'A' + (j-'Z'-1);
        }
        m[i] = j;
    }
    printf("m: %s\n", m);
}
```

Spiegazione

Spiegazione

Spiegazione

Spiegazione

Spiegazione

Spiegazione

Esercizio 4 - Programmazione C (4 punti)

Si ipotizzi di dovere implementare una versione semplificata del gioco dello Scarabeo, e di avere a disposizione le seguenti strutture dati:

```
int punt[26] = {2, 5, 1, 5, 4, ...};  
char lett[10] = {'f', 'c', 'a', 'a', 'b', 's', ...};  
int num;
```

dove

- **punt** è un array che contiene il punteggio che viene assegnato dal gioco a ciascuna lettera ('a' vale 2, 'b' vale 5, 'c' vale 1, ecc.),
- **lett** è un array che contiene le lettere che il giocatore ha in mano in questo momento, e
- **num** è il numero di lettere che il giocatore ha in mano in questo momento (e quindi anche il numero di lettere da considerare nell'array **lettere**).

Si scriva un programma che 1) chiede all'utente di inserire una parola di al massimo 10 caratteri, 2) verifica se la parola può essere scritta usando le lettere a disposizione in **lett**, e calcola e stampa a video il punteggio assegnato a quella parola.

Attenzione: se la parola usa più volte la stessa lettera, quella lettera deve essere presente un numero sufficiente di volte in **lett**.

Esercizio 5 - Programmazione C (4 punti)

Si ipotizzi di dovere implementare una versione semplificata del gioco "Campo Minato".

Nel gioco "Campo Minato" il tabellone consiste di una matrice di interi di dimensioni NxN.

Ciascuna casella contiene un numero compreso tra -1 e 8.

Il numero -1 viene utilizzato per indicare la presenza di una bomba, mentre un numero compreso tra 0 e 8 viene utilizzato per indicare il numero di bombe che si trovano nell'immediato intorno di quella casella.

1	1	2	1	1
1	-1	3	-1	2
2	3	-1	3	-1
1	-1	3	3	2
1	1	2	-1	1

Per intorno di una casella si intendono le quattro caselle presenti sopra, sotto, a sinistra e a destra, più le quattro caselle che si trovano nelle posizioni diagonali, ovvero, in alto a sinistra, in alto a destra, in basso a sinistra e in basso a destra.

Scrivere un programma che presa una matrice NxN verifica che il tabellone sia ben riempito.

Attenzione: Le caselle che si trovano sui lati e negli angoli della matrice non hanno un intorno composto di 8 caselle. L'intorno di un angolo della matrice ha solo 3 caselle, mentre l'intorno di una casella sul bordo della matrice ha solo 5 caselle.

