

	Politecnico di Milano Scuola di Ingegneria Industriale e dell'Informazione INFORMATICA B Seconda prova in itinere - 2 Febbraio 2017		COGNOME E NOME				
	RIGA	COLONNA	MATRICOLA				
			Spazio riservato ai docenti <table border="1" data-bbox="1262 528 1508 584"> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </table>				

- Il presente plico contiene 3 esercizi e **deve essere debitamente compilato con cognome e nome, numero di matricola.**
- Il tempo a disposizione è di 1 ora e 30 minuti.
- 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.
- È possibile scrivere a matita (e non occorre ricalcare al momento della consegna!).
- **È vietato utilizzare telefoni o pc.** Chi tenti di farlo vedrà annullata la sua prova.
- 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.**

Esercizio 1 (6 punti)

Scrivere un programma in C che legga una stringa `str` da tastiera (priva di spazi e lunga al più 500 caratteri). La stringa `str` contiene una sequenza ordinata (ordine alfabetico) di caratteri non necessariamente tutti distinti (può cioè contenere delle ripetizioni). Per ogni carattere `c` contenuto nella stringa `str`, **a partire dall'ultimo fino ad arrivare al primo**, il programma dovrà stampare (senza lasciare spazi) il carattere `c`, seguito dal numero di volte che questo compare in `str`.

Esempio

Se la stringa `str` letta è "aadddffffzzzz", il programma dovrà stampare z4f3d3a2

Soluzione

```
#include <stdio.h>
#include <string.h>

int main()
{
    char str[501],c;
    int freq,i;

    scanf("%s",str);

    c=str[strlen(str)-1];
    freq=1;
    for (i=strlen(str)-2; i>=0; i--)
    {
        if (str[i]!=c)
        {
            printf("%c%d",c,freq);
            freq=1;
            c=str[i];
        }
        else
            freq++;
    }

    printf("%c%d\n",c,freq);

    return 0;
}
```

Esercizio 2 (6 punti)

Il servizio online Spotify fornisce musica in streaming sotto-forma di alcuni canali a tema. Ad ogni canale è associato un codice identificativo, un titolo ed una lista di canzoni. A ciascuna canzone è associato un codice identificativo, un titolo, e una durata in secondi. L'insieme di tutti i canali è contenuto in una variabile `canali` già inizializzata. Si assuma di avere poi un array `utenti` che contiene i dati degli utenti attualmente collegati, ovvero nome dell'utente ed il canale che sta ascoltando in quel particolare momento. Date le seguenti dichiarazioni:

```
#define NUM_CANALI 20
#define NUM_UTENTI 100
#define MAX_NUM_CANZONI 50

typedef char stringa[40];

typedef struct {
    int identificativo;
    stringa titolo;
    canzone canzoni[MAX_NUM_CANZONI];
    int numCanzoni;
} canale;

typedef struct {
    stringa nomeUtente;
    int canale;
} utente;

utente utenti[NUM_UTENTI];
canale canali[NUM_CANALI];
```

- definire un tipo di dato `canzone` in grado di contenere tutte le informazioni associate a ciascuna canzone (codice identificativo, titolo e durata);
- scrivere un frammento di codice che stampi a video, per ogni canale, il nome del canale ed il numero di utenti che attualmente stanno ascoltando quel canale.

Nota. Si assuma che `utenti` e `canali` siano già state inizializzate e contengano già i dati di `NUM_UTENTI` utenti e di `NUM_CANALI` canali (non è quindi necessario caricare in alcun modo i dati degli utenti e dei canali).

Soluzione

A.

```
typedef struct {
    stringa titolo;
    int identificativo;
    int durata;
} canzone;
```

B.

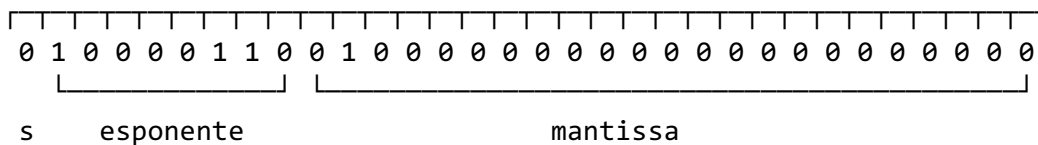
```
int i, j;
for (i = 0; i < NUM_CANALI; i++)
{
    int sumUtenti = 0;
    for (j = 0; j < NUM_UTENTI; j++)
    {
        if (utenti[j].canale == canali[i].identificativo)
            sumUtenti++;
    }
    printf("Il canale '%s' ha %d utenti al momento.\n", canali[i].titolo, sumUtenti);
}
```

Esercizio 3 (5 punti)

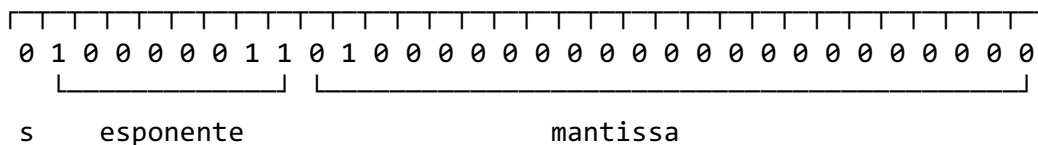
Parte A

Siano dati i seguenti numeri codificati in virgola mobile a singola precisione su 32 bit (IEEE-754):

- Codifica di A:



- Codifica di B:



Quanto vale A/B (in base 10)? Giustificare la risposta.

Parte B

Codificare in CPL2 -16 e 3, utilizzando il minor numero di bit che consenta di codificare entrambi i numeri. Effettuare la somma in CPL2 di -16 e 3 codificati al punto precedente. Si verifica un overflow? Giustificare la risposta.

Soluzione

Parte A

A e B hanno la stessa mantissa M (1.25).

L'esponente di A è $134 - 127 = 7$.

L'esponente di B è $131 - 127 = 4$.

Quindi, $A = M * 2^7$ e $B = M * 2^4$

Da cui risulta $A/B = 2^{7-4} = 2^3 = 8$

Parte B

Occorrono 5 bit per codificare i numeri fra -16 e +15

$-16_{10} = 10000_{CPL2}$

$3_{10} = 00011_{CPL2}$

10000 (-16₁₀)

00011 (+3₁₀)

10011 (-13₁₀)

Non si verifica overflow perché gli operandi hanno segno discorde.