

Informatica - Allievi Ambientali - Prova del 6 febbraio 2023

ATTENZIONE Per superare la prova occorre ottenere almeno 18 in ognuna delle 2 parti. Inoltre:

- Utilizzare solo i fogli messi a disposizione, scrivendo chiaramente sulla metà sinistra del fronte-spazio (**come da figura**): sulla prima riga prima il cognome e poi il nome (entrambi in stampatello), sulla riga seguente numero di matricola (o identificativo di persona) e data dell'appello, sulla riga seguente parte a cui si riferisce il foglio (1 Parte o 2 Parte) e firma.
- Consegnare le due parti su fogli separati.
- Per ogni esercizio indicare il numero dello stesso separando chiaramente le soluzioni con una linea orizzontale.
- Scrivere in modo **CHIARO e COMPRENSIBILE** a penna blu o nera o matita morbida (**max HB**). Non usare penne di altri colori o matite troppo dure, di difficile lettura.
- Consegnare solo i fogli protocollo contenenti le due parti, non il testo o la "brutta".

Cognome	Nome
Matricola	Data
Parte X	Firma
Esercizio n° Y	
<small>Testo di prova per la simulazione della prova scritta. Il testo è generato automaticamente dal sistema di simulazione. Non è possibile modificare il testo.</small>	
Esercizio n° Z	
<small>Testo di prova per la simulazione della prova scritta. Il testo è generato automaticamente dal sistema di simulazione. Non è possibile modificare il testo.</small>	
...	

Compiti non soddisfacenti queste semplici regole subiranno penalizzazioni.
Durante la prova non è permesso consultare libri o appunti, nè lasciare l'aula.

Parte I - Teoria [Totale punti 30]

Esercizio 1 - Aritmetica binaria [5 punti]

Si codifichino in complemento a 2 i numeri 6 e -12, se ne esegua la somma e la verifica di correttezza della stessa, eseguendo tutti i passaggi.

Soluzione

$$\begin{array}{r} 6|0 \\ 3|1 \\ 1|1 \\ \hline +6_{10} = 0110 \end{array}$$

$$\begin{array}{r} 12|0 \\ 6|0 \\ 3|1 \\ 1|1 \\ \hline +12_{10} = 01100_2 \end{array}$$

Ne facciamo il complemento a 2 invertendo i bit e sommando 1.

$$01100 \rightarrow 10011$$

$$\begin{array}{r} 10011 + \\ 1 \\ \hline 10100 \end{array}$$

$$10100_2 = -12_{10}$$

Facciamo la somma.

$$\begin{array}{r} 10100 + \\ 00110 \\ \hline 11010 \end{array}$$

Verifica, sottraendo 1 e invertendo i bit trovo il numero positivo corrispondente.

```
11010 -  
1  
-----  
11001 -> 00110
```

$$00110_2 = 1 \cdot 2^1 + 1 \cdot 2^2 = 6$$

Quindi il risultato è $11010_2 = -6_{10}$

Esercizio 2 - Macchina di Von Neumann [5 punti]

Si spieghino brevemente i ruoli del registro contatore di programma e del registro istruzione corrente. Quanto potrebbero esser lunghi, in bit, se abbiamo un bus dati lungo 16 bit e un bus indirizzi lungo 8 bit?

Soluzione

il registro contatore di programma contiene un indirizzo, quindi potrà essere lungo come il bus indirizzi (8 bit in questo caso), mentre il registro istruzione corrente contiene un'istruzione, quindi un dato di memoria e potrà avere un numero di bit pari a quelli del bus dati (16 bit in questo caso).

Esercizio 3 - SQL

Si progetti in SQL lo schema di una base di dati per la gestione di dati relativi a popolazioni di animali protetti presenti sul territorio. Per ogni animale sono memorizzati il nome, il nome dell'area in cui è stato rilevato (che assumiamo unico nel sistema), la data del rilevamento, il numero di telefono del responsabile del rilevamento, e la quantità di esemplari rilevati. Si consideri che possono esserci diversi rilevatori nella stessa area e nello stesso giorno. Per ogni area sono memorizzate le coordinate geografiche dei due punti estremi del rettangolo che la include (in alto a sinistra e in basso a destra), il nome e la provincia di appartenenza. [5 punti]

Si progettino due query: la prima per ottenere la lista degli animali rilevati a gennaio 2023 in provincia di Lecco. La seconda per ottenere le aree di rilevamento in cui sono stati rilevati galli cedroni della provincia in cui non siano stati rilevati più di 10 galli cedroni in totale. [5 punti]

Soluzione

```
CREATE TABLE Rilevamento  
(NomeAnimale CHAR(20) NOT NULL,  
NomeLuogo CHAR(20) NOT NULL,  
NumeroEsemplari INTEGER,  
Telefono CHAR (13) NOT NULL,  
Data DATE NOT NULL,  
PRIMARY KEY (NomeAnimale, NomeLuogo, Data, Telefono)  
);  
  
CREATE UNIQUE INDEX Rilevamento_key ON Rilevamento(NomeAnimale, NomeLuogo, Data);  
  
CREATE TABLE Area  
(NomeArea CHAR(20) NOT NULL PRIMARY KEY,  
Latitudine1 CHAR(8),  
Longitudine1 CHAR (8),  
Latitudine2 CHAR(8),  
Longitudine2 CHAR (8),  
Provincia CHAR(6)  
);  
  
CREATE UNIQUE INDEX Area_key ON Rilevazione(NomeArea);  
  
SELECT NomeAnimale  
FROM Rilevamento JOIN Area ON (Area.NomeArea = Rilevamento.NomeLuogo)  
WHERE (Data > 31-12-2022) AND  
(Data < 01-02-2023) AND
```

```
(Provincia = 'LC');
```

```
SELECT NomeLuogo  
FROM Rilevamento  
WHERE (NomeAnimale = 'Gallo Cedrone')  
GROUP BY Provincia  
HAVING SUM(NumeroEsemplari) < 10  
);
```

Esercizio 4 - Sistema Operativo

Si descriva brevemente il ruolo del gestore dei processi in un sistema operativo. Come gestisce il fatto che l'esecuzione di un programma sia giunta all'istruzione in cui ci si aspetta un input dell'utente? **[5 punti]**

Soluzione

Vd libro di testo. Il gestore dei processi gestisce l'esecuzione dei processi assegnando ad ognuno la possibilità di essere eseguito sulla CPU per un quanto di tempo e allo stesso tempo gestendo una o più code di processi in attesa del proprio turno. Inoltre gestisce interruzioni interne ed esterne. In particolare, la necessità di accettare un input dall'utente è un'interruzione esterna. Il processo in esecuzione viene messo in stato di *attesa* e una volta che arrivi l'input dalla tastiera (gestito dal gestore dei processi con una funzione di gestione dell'interrupt, viene rimesso in coda in stato di *pronto*.

Esercizio 5 - Strumenti di produttività

Progettare un foglio elettronico che permetta di calcolare la formula $dx/dt = k1 * x(t)^2 + k2 * x(t) + k3$ con passo $P=0.5$ da 0 a 10. **[5 punti]**

Soluzione

Si dovevano indicare le caselle e i relativi contenuti. Una possibile organizzazione è la seguente. Nella prima riga, A0 e B0 vanno inserite le intestazioni delle colonne che conterranno gli istanti di tempo (t) e i valori della variabile x, rispettivamente. In D0 si può inserire $P =$ e in E0 il valore del passo 0.5. In D1 si può inserire $x(0) =$ ed in E1 il valore di x al tempo 0. In G0 possiamo scrivere $k1 =$ e in H0 il valore di $k1$. le altre costanti potrebbero andare sulle righe successive. Nella colonna A va inserito il tempo da 0 a 10 con passo 0.5 ad esempio scrivendo in A1 0, in A2 A1+E0 e trascinando il valore di questa casella fino ad ottenere il valore 10. Nella casella B1 va inserito il riferimento al valore iniziale di x che qui supponiamo sia memorizzato in E1: $=E1$. Nella casella B2 va inserito il valore aggiornato di x, cioè $=E0((H0*B1*B1)+H1*B1) + B1$.*

Parte II - Linguaggio C [Totale punti 30]

Esercizio 1 - Sviluppo

[6 punti] Si definiscano le opportune strutture dati per rappresentare un array di voli aerei. Per ogni volo è necessario memorizzare il numero (intero), l'aeroporto di partenza e di arrivo (stringhe), la durata del volo (ore e minuti) e il tipo di volo (domestico, europeo, intercontinentale).

1. [6 punti] Si scriva una funzione `inserisci` che riceva come parametri l'array di voli, la sua dimensione e un volo `v`. La funzione deve cercare se nell'array è già presente un volo con lo stesso numero di `v`. In caso fosse presente, è necessario sostituirlo con `v` qualora la durata del volo `v` sia inferiore alla durata del volo trovato. La funzione deve restituire 1 se il volo è stato trovato, 0 in caso contrario.
2. [6 punti] Si scriva una funzione `coincidenza` che riceva con parametri l'array di voli, la sua dimensione e i nomi di due aeroporti `aero1` e `aero2`. La funzione deve verificare se nell'array è presente una soluzione di viaggio per andare da `aero1` a `aero2` con due voli di cui al massimo uno intercontinentale. La funzione deve inoltre restituire la durata totale dei due voli. (Ad esempio, se `aero1` è "Milano Linate LIN" e `aero2` è "Los Angeles LAX", si ha una coincidenza se sono presenti due voli `v1` e `v2`, tali che ad esempio `v1` è europeo e va da "Milano Linate LIN" a "Parigi CDG" e `v2` va da "Parigi CDG" a "Los Angeles LAX".)

Soluzioni

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

#define MAX 100

typedef enum {DOMESTICO, EUROPEO, INTERCONTINENTALE} tipo_volo_t;

typedef struct
{
    int ore;
    int minuti;
} durata_t;

typedef struct
{
    int numero;
    char aeroporto_partenza[MAX];
    char aeroporto_arrivo[MAX];
    durata_t durata;
    tipo_volo_t tipo_volo;
} volo_t;

int inserisci(volo_t Voli[], int dim, volo_t v)
{
    int i, trovato;

    trovato = 0;
    for (i = 0; i < dim && trovato == 0; i++)
        if (Voli[i].numero == v.numero)
        {
            trovato = 1;
            if (Voli[i].durata.minuti + Voli[i].durata.ore * 60 >
                v.durata.minuti + v.durata.ore * 60)
                Voli[i] = v;
        }

    return trovato;
}

durata_t coincidenza (volo_t Voli[], int dim, char aero1[], char aero2[])
```

```

{
    int i, j;
    int trovato;
    durata_t totale = {0, 0};    //Per convenzione restituisco {0, 0} se non trovo la coincidenza

    trovato = 0;
    for (i = 0; i < dim && trovato == 0; i++)
    {
        if (strcmp(Voli[i].aeroporto_partenza, aero1) == 0)
        {
            for (j = 0; j < dim && trovato == 0; j++)
            {
                if (strcmp(Voli[i].aeroporto_arrivo, Voli[j].aeroporto_partenza) == 0 &&
                    strcmp(Voli[j].aeroporto_arrivo, aero2) == 0 &&
                    (Voli[i].tipo_volo != INTERCONTINENTALE || Voli[i].tipo_volo != INTERCONTINENTALE))
                {
                    trovato = 1;
                    totale.minuti = Voli[i].durata.minuti + Voli[j].durata.minuti;
                    totale.ore = Voli[i].durata.ore + Voli[j].durata.ore + totale.minuti / 60;
                    totale.minuti = totale.minuti % 60;
                }
            }
        }
    }
    return totale;
}

```

Esercizio 2 - Rilevazione di errori

[6 punti] Sia dato il seguente programma il cui scopo è verificare se una matrice di interi $N \times M$ inserita dall'utente contiene *al massimo tre zeri in ogni colonna*. In caso affermativo, il programma deve stampare la media delle componenti della matrice. Si identifichino gli errori presenti nel programma, motivandone brevemente la scelta, e li si corregga. Sono presenti 6 linee errate o mancanti.

1. #include <stdio.h>	28. }
2.	29.
3. #define N 3	30. int main()
4. #define M 5	31. {
5.	32. int Matrice[N][M];
6. int zeri_colonne(int Mat[][], float *p_media)	33. int i, j, res;
7. {	34. float media;
8. int somma = 0;	35.
9. int conta_zeri, flag, i, j;	36. printf("Inserisci la matrice: ");
10.	37. for (i = 0; i < N; i++)
11. flag = 1;	38. for (j = 0; j < M; j++)
12. for (j = 0; j < M && flag == 1; j++)	39. scanf("%d", &Matrice[i][j]);
13. {	40.
14. for (i = 0; i < N; i++)	41. res = zeri_colonne(Matrice, media);
15. {	42. if (res == 1)
16. if (Mat[i][j] == 0)	43. {
17. conta_zeri++;	44. printf("La matrice contiene al massimo
18.	tre zeri per colonna!\n");
19. somma = Mat[i][j];	45. printf("La media e' %f\n", media);
20. }	46. }
21.	47. else
22. if (conta_zeri >= 3)	48. printf("La matrice NON contiene al massimo
23. flag = 0;	tre zeri per colonna!\n");
24. }	49.
25.	50. return 0;
26. *p_media = somma / (N * M);	51. }
27. return flag;	

Soluzioni

1. Linea 41: `res = zeri_colonne(Matrice, media);` → `res = zeri_colonne(Matrice, &media);`
2. Linea 6: `int zeri_colonne(int Mat[][], float *p_media)` → `int zeri_colonne(int Mat[N][M], float *p_media)` oppure `int zeri_colonne(int Mat[][M], float *p_media)`
3. Linea 13-14: Aggiungere `conta_zeri = 0;`
4. Linea 19: `somma = Mat[i][j];` → `somma += Mat[i][j];`
5. Linea 26: `*p_media = somma / (N * M);` → `*p_media = (float)somma / (N * M);` oppure, equivalentemente, Linea 8: `int somma = 0; → float somma = 0;`
6. Linea 22: `if (conta_zeri >= 3) → if (conta_zeri > 3)`

Esercizio 3 - Comprensione del codice

[6 punti] Si dica cosa viene stampato dal seguente programma nei seguenti casi:

1. è inserita la matricola 645432;
2. è inserita la propria matricola.

Si dica poi cosa viene salvato nell'array `Array` per una generica matricola.

```
#include <stdio.h>

#define M 7
#define N 4

void mah(int matricola, int Array[M])
{
    int i;
    Array[M-1] = matricola % 2;
    for (i = M-2; i >= 1; i--)
    {
        matricola /= 10;
        Array[i] = matricola % 2;
    }

    Array[i] = matricola % 10;
}

void boh(int Array[M])
{
    int i, j;
    for (i = 0; i < N; i++)
    {
        for (j = 0; j < N; j++)
            printf("%d", Array[i+j]);
        printf("\n");
    }
}

int main()
{
    int matricola, Array[M];

    printf("Inserisci la matricola: ");
    scanf("%d", &matricola);

    mah(matricola, Array);
    boh(Array);
    return 0;
}
```

Soluzioni

1. Inserendo la matricola 645432 viene stampato:

```
6001
0010
0101
1010
```

2. Inserendo la matricola 788955 viene stampato:

```
7100
1001
0011
0111
```

Nella prima componente dell'array `Array` viene salvata la prima cifra della matricola. Nelle componenti seguenti viene salvato uno 0 o un 1 per ogni cifra della matricola. In particolare, se la cifra è pari viene salvato 0, mentre se è dispari viene salvato 1.