

Informatica - Allievi Ambientali - Prova del 17 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 frontespizio (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

.....

Esercizio n° Z

.....

...

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 - Funzioni logiche [5 punti]

Si consideri la funzione logica $f = (A \text{ AND } B) \text{ OR } (\text{NOT } (C \text{ OR } B))$. Si fornisca la tabella di verità, eseguendo tutti i passaggi.

Soluzione

A	B	C	A AND B	(C OR B)	NOT (C OR B)	f
0	0	0	0	0	1	1
0	0	1	0	1	0	0
0	1	0	0	1	0	0
0	1	1	0	1	0	0
1	0	0	0	0	0	0
1	0	1	0	1	0	0
1	1	0	1	1	0	0
1	1	1	1	1	0	1

Esercizio 2 - Macchina di Von Neumann [5 punti]

Si spieghino brevemente i ruoli dell'unità di controllo e dell'unità aritmetico-logica nell'architettura di Von Neumann.

Soluzione

(Vd. libro di testo). L'unità di controllo interpreta l'istruzione presente nel registro istruzione corrente ed indirizza i dati all'unità aritmetico-logica che si occupa di effettuare le operazioni aritmetiche e valutare eventuali condizioni sui dati presenti nei registri *A* (*accumulatore*) e *B*, depositando il risultato dell'operazione nel registro accumulatore.

Esercizio 3 - SQL

Si progetti in SQL lo schema di una base di dati per la gestione di dati relativi a qualità dell'aria nel territorio. Per ogni stazione di rilevamento sono memorizzati il nome, il nome del comune in cui è posizionata, ed il numero di telefono dell'ente responsabile. I dati vengono rilevati due volte al giorno. Per ogni rilevamento vengono memorizzati la stazione di rilevamento la data, il fatto che si il rilevamento della mattina o della sera, e i dati di temperatura, SO_2 , CO e O_3 . [5 punti]

Si progettino due query: la prima per ottenere le rilevazioni fatte nel comune di Milano nel mese di gennaio 2023 in cui la temperatura era superiore a 10 gradi, la seconda per ottenere i valori dei parametri misurati, la data e il comune relativo per tutte le rilevazioni per cui il valore di temperatura massimo nel comune relativo non sia superiore a 10 gradi nel mese di gennaio 2023. [5 punti]

Soluzione

```
CREATE TABLE Rilevamento
(NomeStazione CHAR(20) NOT NULL,
Quando CHAR (1) NOT NULL,
Data DATE NOT NULL,
CO DECIMAL(5,1),
SO2 DECIMAL(3,1),
O3 DECIMAL(4,1),
PRIMARY KEY (Stazione, Data, Quando)
);

CREATE UNIQUE INDEX Rilevamento_key ON Rilevamento(Stazione, Data, Quando);

CREATE TABLE Stazione
(Nome CHAR(20) NOT NULL PRIMARY KEY,
Comune CHAR(20),
Tel CHAR (13)
);

CREATE UNIQUE INDEX Area_key ON Rilevazione(Nome);

SELECT *
FROM Rilevamento JOIN Stazione ON (Stazione.Nome = Rilevamento.NomeStazione)
WHERE (Data > 31-12-2022) AND
(Data < 01-02-2023) AND
(Comune = 'Milano') AND
(Temperatura > 10);

SELECT Comune, Data, Temperatura, CO, SO2, O3
FROM Rilevamento JOIN Stazione ON (Stazione.Nome = Rilevamento.NomeStazione)
WHERE (Data > 31-12-2022) AND
(Data < 01-02-2023)
GROUP BY (Comune)
HAVING MAX(Temperatura) <= 10;
```

Esercizio 4 - Reti di calcolatori

Si descriva brevemente il ruolo del modem nei sistemi di trasmissione dell'informazione, specificando la distinzione tra le diverse modalità di codifica dei bit. **[5 punti]**

Soluzione

Vd libro di testo. Il modem si occupa di modulare il segnale digitale trasformandolo in segnale analogico e viceversa. Il segnale analogico viene trasmesso su un canale alterando una frequenza che viene denominata “portante”. Le due modalità principali con cui questo è fatto sono la modulazione di ampiezza, in cui si varia l'ampiezza del segnale in corrispondenza dei due valori 0 e 1, e la modulazione di frequenza, in cui si varia la frequenza.

Esercizio 5 - Strumenti di produttività

Progettare un foglio elettronico che permetta di memorizzare i dati menzionati nell'esercizio 3. Inoltre si indichi come possono essere calcolati e posizionati nel foglio elettronico i valori medi e massimi settimanali per ogni parametro chimico.

Soluzione

Si dovevano indicare le caselle e i relativi contenuti. Una possibile organizzazione è riportata nel file allegato.

Parte II - Linguaggio C [Totale punti 30]

Esercizio 1 - Sviluppo

[6 punti] Si definiscano le opportune strutture dati per rappresentare un array di eventi sismici. Per ogni evento sismico, è necessario memorizzare la data e l'ora (ore, minuti e secondi) in cui si è verificato, la magnitudo (espressa come numero decimale da 0 a 10), la profondità (in km) e le coordinate dell'epicentro (latitudine e longitudine).

1. [6 punti] Si scriva una funzione `massimo` che riceva come parametri l'array degli eventi sismici, la sua dimensione e due anni `y1` e `y2`. La funzione deve restituire la magnitudo del sisma di magnitudo massima verificatosi tra gli anni `y1` e `y2`. Inoltre, la funzione deve restituire (ad esempio mediante parametro passato per indirizzo) le coordinate di tale sisma. Nel caso ve ne fossero più di uno a magnitudo massima, è irrilevante di quale sisma sono restituire le coordinate. Nel caso non si trovassero sismi tra gli anni `y1` e `y2` la funzione deve restituire `-1`.
2. [6 punti] Si scriva una funzione `percentuali_mese` che riceva come parametri l'array degli eventi sismici, la sua dimensione e un valore decimale `soglia`. La funzione deve calcolare la percentuale di eventi sismici con magnitudo almeno pari a `soglia` verificatisi per ogni mese dell'anno. La funzione deve visualizzare a console un output della forma (ad esempio con `soglia = 4.5`):

```
La percentuale per mese di sismi con magnitudo almeno 4.5 e':
Gennaio: 12.7
Febbraio: 0
Marzo: 5
...
Dicembre: 1.5
```

Infine la funzione deve restituire il numero totale di sismi con magnitudo almeno pari a `soglia`.

Soluzioni

```
#include <stdio.h>

#define MESI 12

typedef struct
{
    float latitudine;
    float longitudine;
} coordinate_t;

typedef struct
{
    int giorno;
    int mese;
    int anno;
} data_t;

typedef struct
{
    int ore;
    int minuti;
    int secondi;
} ora_t;

typedef struct
{
    coordinate_t coordinate;
    data_t data;
    ora_t ora;
    float profondita;
    float magnitudo;
} sisma_t;
```

```

float massimo (sisma_t Sismi[], int dim, int y1, int y2, coordinate_t *p_coord)
{
    int i, imax=-1;
    for (i = 0; i < dim; i++)
    {
        if (Sismi[i].data.anno >= y1 && Sismi[i].data.anno <= y2)
        if (imax == -1 || Sismi[i].magnitudo > Sismi[imax].magnitudo)
            imax = i;
    }

    if (imax == -1)
        return -1;

    *p_coord = Sismi[imax].coordinate;

    return Sismi[imax].magnitudo;
}

```

```

int percentuali_mese (sisma_t Sismi[], int dim, float soglia)
{
    int Cont[MESI] = {0};
    int i, tot;

    char Mesi[MESI][20] = {"Gennaio", "Febbraio", "Marzo", "Aprile",
        "Maggio", "Giugno", "Luglio", "Agosto",
        "Settembre", "Ottobre", "Novembre", "Dicembre"};

    tot = 0;
    for (i = 0; i < dim; i++)
        if (Sismi[i].magnitudo >= soglia)
        {
            Cont[Sismi[i].data.mese]++;
            tot++;
        }

    printf("La percentuale per mese di sismi con magnitudo almeno %f e':\n", soglia);
    for (i = 0; i < MESI; i++)
        printf("%s: %f\n", Mesi[i], (float)Cont[i] / tot);

    return tot;
}

```

Esercizio 2 - Rilevazione di errori

[6 punti] Sia dato il seguente programma il cui scopo è modificare una stringa inserita dall'utente eliminando tutti i caratteri che compaiono in più occorrenze consecutive. Ad esempio, se l'utente inserisce la stringa "Nel mezzo del cammmmin di nosssstra vita", essa verrà modificata in "Nel mezo del camin di nostra vita". Il programma stampa la stringa modificata e il numero di caratteri rimossi.

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>	10. k = 0;
2.	11.
3. #define MAX 1001	12. for (i = 0; stop == 0; i++)
4.	13. {
5. void cancella (char *s, int P[], int pdim)	14. while (k < pdim && j != P[k])
6. {	15. {
7. int i, j, k, stop;	16. j++;
8.	17. k++;
9. j = 0;	18. }

```

19.
20.     s[i] = s[j];
21.     j++;
22.
23.     if (s[i] == '\0')
24.         stop = 1;
25.     }
26. }
27.
28. int modifica (char s)
29. {
30.     int i, IndiciDaCancellare[MAX], dim;
31.     char c;
32.
33.     c = s[0];
34.     dim = 0;
35.     for (i = 1; s[i] != '\0'; i++)
36.     {
37.         if (s[i] = c)
38.         {
39.             IndiciDaCancellare[dim] = i;
40.             dim++;
41.         }
42.         else
43.             c = s[i];
44.     }
45.
46.     cancella(s, IndiciDaCancellare, dim);
47.
48.     return dim;
49. }
50.
51. int main()
52. {
53.     char s[MAX];
54.
55.     printf("Inserisci una stringa: ");
56.     gets(s);
57.
58.     l = modifica(s);
59.
60.     printf("La stringa modificata e': %s\n", s);
61.     printf("Il numero di caratteri
rimossi e' %d\n", &l);
62.
63.     return 0;
64. }

```

Soluzioni

- Linea 11: Aggiungere `stop = 0;`
- Linea 14: `while (k < pdim && j != P[k])` → `while (k < pdim && j == P[k])`
- Linea 28: `int modifica (char s)` → `int modifica (char s[])`
- Linea 37: `if (s[i] = c)` → `if (s[i] == c)`
- Linea 61: `printf("Il numero di caratteri rimossi e' %d\n", &l);` → `printf("Il numero di caratteri rimossi e' %d\n", l);`
- Linea 54: Aggiungere `int l;`

Esercizio 3 - Comprensione del codice

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

- è inserita la matricola 123456;
- è inserita la propria matricola.

Si dica poi cosa viene restituito dalla funzione `f2` per una generica matricola. Si spieghi infine perchè l'ultima `printf` stampa sempre lo stesso valore indipendentemente dalla matricola inserita.

```

#include <stdio.h>

#define MAX 6

int f1(int A[])
{
    int i=0, j=MAX-1, x=0;

    while (i < j)
    {
        x += (A[i] - A[j]);
        i++;
        j--;
        printf("%d -- %d -- %d\n", i, j, x);
    }
    return x;
}

int f2(int* V, int i)
{
    if (i < MAX)
        return 1 - V[i] % 2 + f2(V, i+1);
    else
        return 0;
}

void f3(int* V, int i)
{
    if (i < MAX)
    {
        *V = V[0] / 2 * 2 + 1;
        f3(&V[i], i+1);
    }
}

```

```

int main()
{
    int V[MAX], i;

    printf("Inserisci le cifre della matricola una per volta: ");
    for (i = 0; i < MAX; i++)
        scanf("%d", &V[i]);

    printf("%d\n", f1(V));
    printf("%d\n", f2(V, 0));
    f3(V, 0);
    printf("%d\n", f2(V, 0));
    return 0;
}

```

Soluzioni

1. Inserendo la matricola 123456 viene stampato:

```

1  --  4  -- -5
2  --  3  -- -8
3  --  2  -- -9
-9
3
0

```

2. Inserendo la matricola 788955 viene stampato:

```

1  --  4  --  2
2  --  3  --  5
3  --  2  --  4
4
2
0

```

La funzione **f2** restituisce il numero di cifre pari contenute nella matricola. L'ultima **printf** stampa sempre il valore 0 in quanto a valle della funzione **f3** l'array **V** contiene solo cifre dispari e l'esecuzione della funzione **f2** su questo array restituisce il numero di cifre dispari, cioè 0. Infatti l'operazione $V[0] / 2 * 2 + 1$ genera un numero dispari indipendentemente dal valore di $V[0]$, essendo effettuata una divisione tra interi.