

Nome:

Cognome:

Matricola:

2 Punti

1. Si descriva la sintassi dell'istruzione `for` e se ne spieghi il significato.

Solution: L'istruzione `for` ha la sintassi:

```
for (expr1; expr2; expr3) { blocco-istruzioni; }
```

dove `expr1` `expr2` ed `expr3` sono tre espressioni del linguaggio C. Di queste, `expr1` è detta inizializzazione e viene eseguita una sola volta e all'inizio del ciclo; `expr2` viene valutata al termine di ogni ripetizione di blocco-istruzioni ed `expr3` viene valutata sempre immediatamente dopo l'esecuzione di blocco-istruzioni. Se `expr2` è vera, si rientra nel ciclo, altrimenti se ne esce. Come già accennato, ogni ripetizione del ciclo consiste nell'esecuzione di blocco-istruzioni seguita immediatamente dalla valutazione di `expr3`.

In altre parole, il ciclo `for` descritto è equivalente al seguente costrutto:

```
expr1;  
while (expr2) {  
    blocco-istruzioni;  
    expr3;  
}
```

3 Punti

2. Il seguente programma dovrebbe, nelle intenzioni dell'autore, chiedere all'utente un numero maggiore di 1, n , facendo ripetere l'input se non viene soddisfatta questa condizione; il programma dovrebbe quindi stampare se n è primo oppure, se non lo è, il numero dei suoi divisori (eccetto sé stesso e l'unità). Ad esempio, se l'utente inserisce 12, il programma dovrebbe rispondere che ha 4 divisori.

```

1 #include <stdio.h>
2 main(){
3     int i;
4     int n;
5     int divisori;
6
7     do {
8         printf("Inserire n:");
9         scanf("%d", &n);
10    } while (n <= 1);
11
12    i=2;
13    while (i<=n) {
14        if ((n%i)==0) divisori++;
15        i++;
16    }
17
18    if (divisori == 0)
19        printf("%d e' un numero primo", n);
20    else
21        printf("%d ha %d divisori", n, divisori);
22
23    system("pause");
24 }
```

Esso contiene due errori. Un errore può essere:

- una istruzione mancante, oppure
- una istruzione con errori di sintassi o di altro tipo.

È richiesto di trovare questi errori e proporre le relative modifiche per rendere il programma correttamente funzionante. Proporre le modifiche nella forma: “Sostituire la riga numero X con: ...” oppure “Inserire quanto segue dopo la riga numero X: ...”.

Solution: Esistono varie soluzioni. Di queste, una tra le più semplici è descritta di seguito.

Sostituire la linea 5 con `int divisori = 0;`

Sostituire la linea 13 con `while (i<n) {`

In alternativa alla modifica della linea 13, è corretto anche inserire, al posto della linea 17, l'istruzione `divisori--`;

5 Punti

3. Scrivere un programma C che:

- anzitutto legga da tastiera una serie di numeri in virgola mobile maggiori di zero salvandoli nel vettore `num1[10]`. L'inserimento termina quando l'utente ha inserito dieci numeri oppure non appena ne inserisce uno minore o uguale a zero (che va scartato).
- successivamente stampi a video il più grande tra i numeri effettivamente inseriti (che quindi potrebbero essere meno di dieci);
- infine calcoli e stampi a video la media aritmetica tra i numeri effettivamente inseriti.

Solution:

```
1 #include <stdio.h>
2
3 main() {
4     float num1[10];
5     int i = 0;
6     float n, max, somma, media;
7     int j;
8     do {
9         printf("Inserire l'elemento numero %d : ", i);
10        scanf("%f", &n);
11        if (n>0) {
12            num1[i] = n;
13            i++;
14        }
15    } while ((n>0) && (i<10));
16
17    max = 0;
18    for (j=0; j<i; j++) {
19        if (num1[j]>max) max=num1[j];
20    }
21    printf("Ecco il massimo: %f \n", max);
22
23    somma = 0;
24    for (j=0; j<i; j++) {
25        somma += num1[j];
26    }
27    media = somma / i;
```

```
28     printf("Ecco la media: %f \n", media);
29
30     system("pause");
31
32 }
```