

## WEB DEVELOPER

**Unità Formativa (UF): Fondamenti di Programmazione**

**Docente: Massimo PAPA**

**Titolo argomento: Correzione semplici esercizi sulla selezione**

# Esercizi di rinforzo selezione

---

<b>Indicazioni sulla consegna</b>	<b>3</b>
Struttura del codice sorgente	4
<b>Testo Esercizi</b>	<b>5</b>
Selezione	5
1 Esercizio	5
2 Esercizio:	5
3 Esercizio:	5
4 Esercizio:	5
5 Esercizio:	5

# Indicazioni sulla consegna

---

Implementare gli algoritmi risolutivi dei seguenti esercizi codificandoli in linguaggio c++. Partire analizzando il problema seguendo la seguente traccia:

- Quali sono gli input del problema?
- Quali sono i vincoli?
- Quali sono gli output?
- Eventualmente suddividi il problema in problemi più semplici e se lo ritieni opportuno descrivi la soluzione di ogni sottoproblema con un flow-chart
- Esegui la codifica dichiarando le variabili una per ogni riga, indicando per ognuna una riga di commento
- Scrivi la sezione dell'inizializzazione variabili
- Scrivi la sezione dell'input
- Scrivi le sezioni elaborative di ogni sottoproblema
- Scrivi la sezione dell'output
- Verifica la codifica utilizzando input di test, cercando di provare anche i casi limite.

~~Preferibilmente carica la cartella zippata del progetto DevCpp, un progetto per ogni esercizio.~~

Attenzione: non utilizzare array o matrici in nessuno dei seguenti esercizi.

## Struttura del codice sorgente

Relativamente alle indicazioni di scrittura del codice, utilizza il seguente schema generale:

```
#include <iostream>
using namespace std;
int main()
{
    // Dichiarazione variabili

    // Inizializzazioni variabili

    // Sezione di input Dati

    // Elaborazione
    // Eventuali sotto processi di Elaborazione
    // ---
    // ---

    // Sezione di output

    return 0;
}
```

# Testo Esercizi

---

## Selezione

### 1 Esercizio

Sommare i primi 10 numeri naturali e visualizzarne il risultato.

### 2 Esercizio:

Calcolare la potenza di un numero elevato a  $n$  usando solo la moltiplicazione.

### 3 Esercizio:

Dati in ingresso  $N$  numeri interi, stampare se viene immesso il numero 9.

### 4 Esercizio:

Dati in ingresso  $N$  numeri interi, stampare se viene immesso il numero 9.

### 5 Esercizio:

Calcolare e visualizzare il valore minimo di una serie di  $N$  misurazioni.

# Correzione Esercizi

---

## Selezione

### 1 Esercizio

```
#include <iostream>

using namespace std;
/*
 * Esercizio 1
 * Sommare i primi 10 numeri naturali e
 * visualizzarne il risultato.
 */

int main()
{
    // Dichiarazione delle variabili
    int i;        // variabile di controllo ciclo
    int somma;    // accumulatore

    // Inizializzazione variabili
    i = 0;
    somma = 0;

    // Loop principale di stampa
    while (i < 10){
        // Stampa di debug
        cout << "** " << i << endl;
        somma += i; // somma = somma + i
        i++;       // Numero naturale successivo
    }

    // Stampa del risultato
    cout << "Somma = " << somma << " ";

    return 0;
}
```

## 2 Esercizio

```
#include <iostream>

using namespace std;

/*
Esercizio 3
Dati in ingresso N numeri interi,
stampare se viene immesso il numero 9.
*/

int main()
{
    // Dichiarazione variabili
    int a;          // base esponente
    int n;          // esponente
    float potenza;  // valore della potenza
    int i;          // indice del ciclo iterativo

    //Inserimento dati
    cout << "Inserisci la base: ";
    cin >> a;
    cout << "Inserisci l'esponente: ";
    cin >> n;
    cout << "Calcoliamo " << a << "^" << n << endl;

    if(n>0){

        // Inizializzazione variabili
        i = 0;
        potenza = 1;

        // Loop principale di elaborazione
        do {
            potenza *= a;    // accumulo prodotto
            i++;             // incremento indice

        }while(i<n);

        // Output dati
        cout << "Il risultato è: " << potenza;

    } else { // se n fuori dominio
        cout << "ERRORE!!";
    }

    return 0;
}
```

### 3 Esercizio

```
/*
 *
 * Dati in ingresso N numeri interi, stampare se viene immesso il
 * numero 9.
 *
 */

#include <iostream>

using namespace std;

int main()
{
    // Dichiarazione variabili
    int N;    // Quantità di numeri che si inseriscono
    int num;  // Numero intero dato in input
    int i;    // Variabile indice del ciclo for

    // Sezione di input Dati
    cout << "Inserisci N: ";
    cin >> N;

    // Verifica dominio di N
    if (N > 0){

        // Ciclo per l'inserimento di N valori numerici
        for(i=0; i<N; i++){
            // Inserimento iesimo Numero
            cout << "Inserisci " << i << "^ numero : ";
            cin >> num;

            // Verifico se il numero è uguale a 9
            if (num == 9)
                cout << "Imnesso il numero 9!!\n";
        }
    } else {

        cout << "\n Il valore di N deve essere >=0\n";

    }

    return 0;
}
```



## 4 Esercizio

```
/*
 *
 * Dati in ingresso N numeri stampare se il numero è divisibile
 * per 3.
 *
 */

#include <iostream>

using namespace std;

int main()
{
    // Dichiarazione variabili
    int N;    // Quantità di numeri che si inseriscono
    int num;  // Numero intero dato in input
    int i;    // Variabile indice del ciclo for

    // Sezione di input Dati
    cout << "Inserisci N: ";
    cin >> N;

    // Verifica dominio di N
    if (N > 0){

        // Ciclo per l'inserimento di N valori numerici
        i = 0;
        while(i < N){
            // Inserimento iesimo Numero
            cout << "Inserisci " << i+1 << "^ numero : ";
            cin >> num;

            // Verifico se il numero è divisibile per 3
            if (num%3 == 0)
                cout << num << " divisibile per 3\n";

            // incremento la variabile indice
            i++;
        }
    } else {

        cout << "\n Il valore di N deve essere >=0\n";

    }

    return 0;
}
```

## 5 Esercizio

```
/*
 *
 * Calcolare e visualizzare il valore minimo
 * di una serie di N misurazioni.
 *
 */

#include <iostream>

using namespace std;

int main()
{
    // Dichiarazione variabili
    int N;          // Quantità di misurazioni che si inseriscono
    float num;      // misurazione data in input
    int i;          // Variabile indice del ciclo for
    int minimo;     // Valore minimo

    // Sezione di input Dati
    cout << "Inserisci N: ";
    cin >> N;

    // Verifica dominio di N
    if (N > 0) {

        // Inserimento fuori ciclo prima misura
        cout << "Inserisci 1^ misura : ";
        cin >> num;
        // inizializzo il valore minimo
        minimo = num;

        // Ciclo per l'inserimento di N valori numerici
        for(i=1; i<N; i++){
            // Inserimento i-esima misurazione
            cout << "Inserisci " << i+1 << "^ misura : ";
            cin >> num;

            // Verifico se il numero è uguale a 9
            if (num < minimo)
                minimo = num;
        }

        // output del valore minimo
        cout << "\n Valore minimo : " << minimo << endl;

    } else {

        cout << "\n Il valore di N deve essere >=0\n";
    }
}
```

```
}  
  
return 0;  
}
```