

## Calcolatrice aritmetica

Esercizio: implementazione di una proposta risolutiva **modulare**

Prevedere un menù di scelta con la possibilità delle seguenti opzioni

- Somma **s** di due valori letti da tastiera
- Prodotto **p** di due valori letti da tastiera
- Differenza **d** di due valori letti da tastiera
- Rapporto **q** di due valori letti da tastiera con controllo di esistenza di un risultato nel campo dei numeri reali



Implementazione con uso del costrutto **switch**

### Sintassi

```
switch (espressione) {  
    case costante : istruzioni  
    .....  
    default: istruzione  
}
```

In realtà **switch**<sup>1</sup> non è altro che una **sequenza di if**, dove la condizione da valutare per verificare se è vera è una COSTANTE.

Permette la verifica del valore di una **espressione**<sup>2</sup>, distinguendo più casi corrispondenti ad altrettanti valori COSTANTI di tale espressione.

Implicitamente, le condizioni che vengono verificate sono date dal confronto che viene fatto per verificare l'**uguaglianza** tra l'espressione indicata e ciascuna delle varie costanti corrispondenti ai casi.

*esempio:*

```
int op;  
.....  
switch (op) {  
  
    case 1:  
        istruzione1;  
        break;  
    case 2:  
        istruzione2;  
        break;  
    ...  
    default:  
        istruzione_default;  
}
```

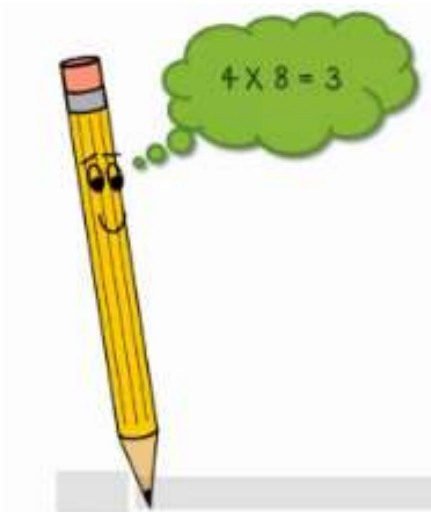
**Attenzione:** l'enunciato **switch** si può usare solo in circostanze limitate. I valori con i quali si può confrontare il selettore devono essere costanti di tipo intero, **char** o enumerativo:

**non si può usare** lo **switch** ad esempio con un selettore di tipo **double** o **String**.

<sup>1</sup>Da risorsa [online](#) (pg.6)

<sup>2</sup>In un linguaggio di programmazione un' **espressione** è un costrutto che combina uno o più valori (ad esempio costanti o variabili), utilizzando operatori ed operandi. L'espressione è (o si può dire ha) il suo valore valutato.

**Proporre un metodo per ogni operazione:**



```
public static double setProdotto (double a, double b) {  
    return a*b;  
}
```



```
public static double setSomma (double a, double b) {  
    return a+b;  
}
```

```
public static double setSottrai (double a, double b) {  
    return a-b;  
}
```

**Possibile segmento di programma – inserito nel flusso principale, all'interno di una classe:**

```
int op;  
double risultato = 0;
```

progettando un metodo  
che ritorna un valore  
recuperato alla chiamata

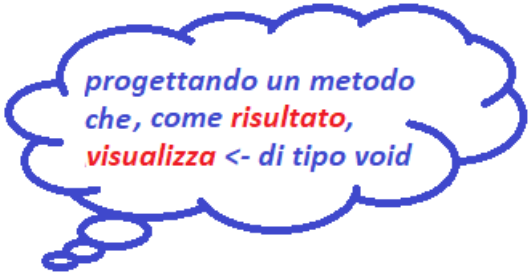
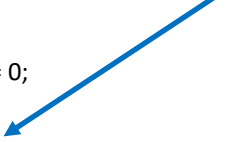
```
switch (op) {  
    case 1: risultato = setDiv(numeroA,numeroB); // per la divisione si prevede controllo:  
        ..... // progetto di un metodo apposito setDiv  
        break;  
    case 2: risultato = setProdotto(numeroA,numeroB);  
        .....  
        break;  
    case 3: risultato = setSomma (numeroA,numeroB);  
        .....  
        break;  
    case 4: risultato = setSottrai (numeroA,numeroB);  
        ....  
        break;  
    default: System.out.println("Opzione non valida");  
}  
System.out.println("RISULTATO = " + risultato);
```

**Possibile codice del metodo *setDiv*, prevedendo controllo:**

```
public static double setDiv (double a, double b) {  
    double num = 0;  
    if (b != 0)  
        num = a/b;  
    else if (a != 0) {  
        System.out.println("\n Non esiste soluzione reale e finita\n\n");  
        System.exit(-1);  
    }  
    else {  
        System.out.println("\n La soluzione è indeterminata\n\n");  
        System.exit(-2);  
    }  
    return num;  
}
```

**Possibile segmento di programma alternativo – inserito nel flusso principale, all'interno di una classe:**

```
int op;  
double risultato = 0;  
switch (op) {  
    case 1: div (numeroA,numeroB);           // per la divisione si prevede controllo:  
        // anche senza break;                // progetto di un metodo apposito div che non torna un valore  
        // e, dopo la visualizzazione del risultato esce dal programma  
    case 2: risultato = setProdotto(numeroA,numeroB);  
        .....  
        break;  
    case 3: risultato = setSomma (numeroA,numeroB);  
        ....  
        break;  
    case 4: risultato = setSottrai (numeroA,numeroB);  
        ....  
        break;  
    default: System.out.println("Opzione non valida");  
}  
System.out.println("RISULTATO = " + risultato);
```



**Possibile codice del metodo *div*, prevedendo controllo:**

```
public static void div (double a, double b) {  
    double num = 0;  
    if (b != 0) {  
        System.out.println("\nIl risultato è: " + a/b);  
        System.exit(0);  
    }  
    else if (a != 0) {  
        System.out.println("\n Non esiste soluzione reale e finita\n\n");  
        System.exit(-1);  
    }  
    else {  
        System.out.println("\n La soluzione è indeterminata\n\n");  
        System.exit(-2);  
    }  
} // fine metodo div
```