

## ➤ Procedure SommaComplex – Documentazione

### ✓ Scopo

Sommare 2 dati di tipo Complex

### ✓ Specifiche

void SommaComplex (Complex &X, Complex &Y, Complex &Z)

### ✓ Descrizione

#### a) Background del problema

Ponendo il tipo Complex sia composto da una parte reale e una immaginaria la sua Somma sarà la somma tra la parte reale e la parte immaginaria

Dunque

$$Z=X+Y$$

$$Z.Re=X.Re+Y.Re$$

$$Z.Im=X.Im+Y.Im$$

#### b) Descrizione del algoritmo

L'algoritmo implementato calcola il risultato utilizzando la definizione su riportata di somma complessa. L'algoritmo in Pascal-like è, dunque, il seguente:

$$Z.Re := X.Re+Y.Re;$$

$$Z.Im := X.Im+Y.Im;$$

### ✓ Riferimenti bibliografici

A. Murli, G. Laccetti, et al., Laboratorio di Programmazione I Liguori 2003

### ✓ Lista dei parametri

Complex	X	:In input contiene il primo valore da sommare, invariato in output
Complex	Y	:In input contiene il secondo valore da sommare, invariato in output
Complex	Z	:In output sarà il valore della nostra somma

### ✓ Indicatore di errore

Nessuno

### ✓ Procedure ausiliarie

Nessuno

### ✓ Raccomandazioni d'uso

Nessuno

✓ Complessità Computazionale

✓ Accuratezza fornita

Dipende dalla accuratezza del sistema floating-point fornito

✓ Esempio d'uso

x Programma chiamante

```
#include "Console.h"
```

```
#include "Complex.cpp"
```

```
using namespace std;
```

```
/
```

```
*=====
```

```
Autore          : Carmine Cuofano
```

```
Matricola: N86001700
```

```
Programma       : ADTComplex
```

```
Data           : 19/04/2015
```

```
-----  
-----*/
```

```
int main()
```

```
{
```

```
    int Tasto;
```

```
    Complex X, Y, Z;
```

```
    //Title
```

```
    cout << "ADT Complex\n\n";
```

```
    //Var Attribution
```

```
    cout << "Definire X\n";
```

```
    cout << "\tReale X: ";
```

```
    cin >> X.Re;
```

```
    cout << "\tImmaginario X: ";
```

```
    cin >> X.Im;
```

```
    cout << "\nDefinire Y\n";
```

```
    cout << "\tReale Y: ";
```

```
    cin >> Y.Re;
```

```
    cout << "\tImmaginario Y: ";
```

```
    cin >> Y.Im;
```

```
    do
```

```
    {
```

```
        //Pulisco lo schermo -Win/Linux-  
        system(CLEAR);
```

```
        //Title
```

```
        cout << "ADT Complex\n\n";
```

```
        //Stampo il menu`
```

```
        cout<<"\t1 - Somma\n";
```

```

cout<<"\t2 - Sottrazione\n";
cout<<"\t3 - Moltiplicazione\n";
cout<<"\t4 - Divisione\n";
cout<<"\nESC to exit";

//Menu`
Tasto=_getch();
switch(Tasto)
{
    case '1' :
        system(CLEAR);
        SommaComplex (X, Y, Z);
        cout<<"La somma Z e`:";
        cout<<"\n\tReale: "<< Z.Re;
        cout<<"\n\tImmaginario: "<< Z.Im;
        _getch();
        break;
    case '2' :
        system(CLEAR);
        SottrazioneComplex (X, Y, Z);
        cout<<"La Sottrazione Z e`:";
        cout<<"\n\tReale: "<< Z.Re;
        cout<<"\n\tImmaginario: "<< Z.Im;
        _getch();
        break;
    case 3 :
        system(CLEAR);
        MoltiplicazioneComplex (X, Y, Z);
        cout<<"La Moltiplicazione Z e`:";
        cout<<"\n\tReale: "<< Z.Re;
        cout<<"\n\tImmaginario: "<< Z.Im;
        _getch();
        break;
    case 4 :
        system(CLEAR);
        DivisioneComplex (X, Y, Z);
        cout<<"La Divisione Z e`:";
        cout<<"\n\tReale: "<< Z.Re;
        cout<<"\n\tImmaginario: "<< Z.Im;
        _getch();
        break;
}
}
while(Tasto != ESC);
}

```

✓ Esempio d'esecuzione

ADT Complex

Definire X

Reale X: 4.05

Immaginario X: 5.04

Definire Y

Reale Y: 5.04

Immaginario Y: 4.05

=====

ADT Complex

1 - Somma

2 - Sottrazione

3 - Moltiplicazione

4 - Divisione

ESC to exit

=====

La somma Z e`:

Reale: 9.09

Immaginario: 9.09