

➤ Procedure QuickSort_Ricorsivo -Documentazione

✓ Scopo

Ordinare un Array di integer tramite l'algoritmo di QuickSort

✓ Specifiche

void QuickSort_Ricorsivo (int Array[] , int Inizio , int Fine)

✓ Descrizione

a) Background del problema

Avendo un Array NON ordinato, ci preoccupiamo di ordinarlo tramite il metodo del QuickSort

✓ Riferimenti bibliografici

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

✓ Lista dei parametri

Array	Array : Array che verrà ordinato
Integer	Inizio : Contiene il valore del limite inferiore del nostro Array
Integer	Fine : Contiene il valore del limite superiore del nostro Array

✓ Indicatore di errore

Nessuno

✓ Procedure ausiliarie

Scambia_Array_int(Array,I,J)
Ottenuti 2 indici scambia i loro valori nel Array

✓ Raccomandazioni d'uso

Nessuno

✓ Accuratezza fornita

Dipende dalla accuratezza del sistema integer fornito

✓ Esempio d'uso

x Programma chiamante

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

```
using namespace std;
```

```
/
```

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

```
Autore : Carmine Cuofano
Matricola: N86001700
```

Programma : QuickSort
Data : 17/05/2015

-----*/

```
main()
{
    int Tasto,
        Da,
        A,
        N,
        *Array;

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

        //Title
        cout << "QuickSort \n\n";

        //Stampo il menu`
        cout << "\t1 - Genera Vettore Random\n";
        cout << "\t2 - Ordina tramite QuickSort\n";
        cout << "\nESC to exit";

        //Menu`
        Tasto = _getch();
        switch(Tasto)
        {
            case '1' ://Genera Random
                system(CLEAR);
                cout << "Definisci la grandezza del Array da
generare: ";

                cin >> N;
                //Alloco l'Array dinamicamente
                Array = new int[N];
                cout << "Minimo valore generabile: ";
                cin >> Da;
                cout << "Massimo valore generabile: ";
                cin >> A;
                Array_Random_Int (Array, N, Da, A);
                cout << "Array Generato";
                Stampa_Array_int (Array, N);
                _getch();
                break;
            case '2' :
                system(CLEAR);
                cout << "Array prima dell'ordinamento:\n";
                Stampa_Array_int (Array, N);
                QuickSort_Ricorsivo (Array, 0, N);
                cout << "\nArray dopo l'ordinamento:\n";
```

```

        Stampa_Array_int (Array, N);
        _getch();
        break;
    }
}
while(Tasto != ESC);
}

#include "Ordinamenti.cpp"

```

✓ Esempio d'esecuzione

QuickSort

- 1 - Genera Vettore Random
- 2 - Ordina tramite QuickSort

ESC to exit

```

=====
- 1
  Definisci la grandezza del Array da generare: 10
  Minimo valore generabile: 1
  Massimo valore generabile: 30
  Array Generato

```

```

=====
  Array prima dell'ordinamento:

```

7	16	17	17	1	15	28	15	13	21
---	----	----	----	---	----	----	----	----	----

```

Array dopo l'ordinamento:

```

1	7	13	15	15	16	17	17	21	28
---	---	----	----	----	----	----	----	----	----