

QuickSort

Created by Gianluca DeLucia
(N86001826)
gian.delucia@studenti.unina.it

SCOPO:

Ordinare un vettore col metodo del QuickSort

SPECIFICHE:

```
void QuickSort1 (int A[], primo, ultimo);
```

DESCRIZIONE:

a) Background del problema:

Ordinare:

- 1) Metodo Ricorsivo
- 2) Metodo Iterativo

b) Algoritmo:

```
function quickSort1 (in A, primo, ultimo)
  var A : array [1...n] of integer
  var n,i,j,primo,ultimo,medio : integer
  begin
    i := primo ;
    j := ultimo;
    medio = A[(primo + ultimo)/2];

    while (i<=j)
      while (A[i] < medio) do
        i++;
      endwhile

      while (A[j] > medio) do
        j--;
      endwhile

      if (i<= j) then
        temp = A[i];
        A[i] = A[j];
        A[j] = temp;
      endif
      i++;
      j--;
    endwhile

    if (primo<j)then
      QuickSort1 (A,primo, j);
    endif
```

```

        if (i<ultimo)then
            QuickSort1 (A,i,ultimo);
        endif
    end
end QuickSort1

```

RIFERIMENTI BIBLIOGRAFICI:

<http://www.federica.unina.it/corsi/programmazione-ii-laboratorio/>
 A. Murli, G. Laccetti, Laboratorio di Programmazione I, Ed. Liguori.
 Criscuolo, F.P. Tramontano, Introduzione alla Programmazione, Algoritmi Imperativi in C++, Ed. Manna.
 G. Dromey, Algoritmi fondamentali, Ed. Jackson.
 B. Kernighan, D. Ritchie, Linguaggio C, Ed. Jackson.
 H. M. Deitel, P. J. Deitel, Corso completo di programmazione, Ed. Apogeo.

LISTA DEI PARAMETRI:

int A[100], primo, ultimo, n, i;

INDICATORI DI ERRORI:

Nessuno.

PROCEDURE AUSILIARIE:

Nessuna.

RACCOMANDAZIONI SULL'USO:

Non utilizzare numeri che abbiano più di otto cifre.

ESEMPIO D'USO:

a) Esempio di programma chiamante:

```

void QuickSort ()
{
    int A[100], primo, ultimo, n, i=0;

    cout<<"\t\t\t\tQuick Sort\n\n\n";
    void QuickSort1 (int A[], int primo, int ultimo);
    carica_vettore(A,n);
    primo=A[i]; ultimo=A[n];
    QuickSort1(A,0,n-1);
    stampa_vettore (A,n);
    system ("PAUSE");
}

void QuickSort1 (int A[], int primo, int ultimo)
{
    int i, j, n, temp=0, medio;

    i = primo ; j = ultimo;
    medio = A[(primo + ultimo)/2];

    while (i<=j){

```

```

        while (A[i] < medio)
            i++;
        while (A[j] > medio)
            j--;
        if (i <= j)
        {
            temp = A[i];
            A[i] = A[j];
            A[j] = temp;

            i++;
            j--;
        }
    }

    if (primo < j){
        QuickSort1 (A, primo, j);
    }
    if (i < ultimo){
        QuickSort1 (A, i, ultimo);
    }
}

```

b) Esempio di esecuzione:

QuickSort

Quanti elementi contiene il vettore? 4

A[0] : 9
 A[1] : 8
 A[2] : 7
 A[3] : 6

vettore ordinato:

[6][7][8][9]