Procedura Partiziona

- Scopo

  Partiziona l'Array in base a un discriminante x

- Specifiche

  void Partiziona (int Array[], int N, int x)

- Descrizione

  a) Background del problema
  La partizione di un Array consiste nel controllare i valori contenuti nelle celle speculari e nel caso in cui siano superiori o inferiori al discriminante x verrano scambiati o meno seguendo il concetto che nella parte sinista <=x e nella destra >x.
  Indicando con Array=(a_0,¡K, a_{N-1}) l' Array

  b) Descrizione del algoritmo
  L' algoritmo adoperato copia il contenuto delle celle da una parte al altra del Array sfruttando due indici che di muovono in maniera speculare, e nel caso siano mal posizionati rispetto al discriminante ne consegue il seguente codice in Pascal-LIKE

```
while (i<j) do
     App:=Array(i)
     Array(i):=Array(j)
     Array(j):=App

     while (Array(i)<=x) do
          i:=i+1;
     end while
     while (Array(j)>x) do
          i:=i+1;
     end while
end while
```

- Riferimenti bibliografici

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

- Lista dei parametri

int Array[]	:	Array. In Output partizionato rispetto al Input
int N	:	Lunghezza del Array. Ricevuta in Input non va ad essere modificata
int i	:	Indice.
int j	:	Cella speculare a quella puntata.
int x	:	Discriminante ricevuto in Input.

- Indcatore d' errore

  Nessuno

- Procedure ausiliarie

  Nessuno

- Raccomandazioni sull'uso

  Nessuno

- Complessità Computazionale

  a) Complessità di tempo
     T(N)=O(N/2)
  b) Complessità di spazio
     S(N)=S(N)

- Esempio d'uso

Esempio di programma chiamante

```
#include <stdio.h>
#include <stdlib.h>

      void Partiziona (int Array[],int N, int x);

main ()
{
      //Dichiarazione
      int         *Array ;
      int         i,
                  N,
                  x;
      //Inizializzazione del array da invertire
      printf      ("Inserisci il numero di elementi del array: ");
      scanf ("%d",&N);

      Array = (int *) malloc (N*sizeof(int ));

      for (i=0; i<N; i++){
            printf("\nInserisci il valore della cella[%d]: ", i);
            scanf("%d",&Array[i]);
      }

      //Inizializzazione Discriminante X
      printf      ("Inserisci il numero del discriminante X: ");
      scanf ("%d",&x);

      //Chiamata delle function
      Partiziona ( Array, N, x);

      //Stampo del risultato
      printf("\n L`Array e` stato partizionato:");
      for (i=0; i<N; i++){
            printf("\nArray[%d]:\t%d", i, Array[i]);
      }
}
```

- Esempio di esecuzione

```
Inserisci il numero di elementi del array: 5

Inserisci il valore della cella[0]: 10
Inserisci il valore della cella[1]: 22
Inserisci il valore della cella[2]: 13
Inserisci il valore della cella[3]: 24
Inserisci il valore della cella[4]: 5
Inserisci il numero del discriminante X: 20

L`Array e` stato partizionato:
   10      5    13    24    22
```