31.Quick sort

#include <stdio.h>

void swap(int \*a, int \*b)

{

int temp = \*a;

\*a = \*b;

\*b = temp;

}

int partition(int array[], int low, int high)

{

int pivot = array[high];

int i = (low - 1);

for (int j = low; j <= high - 1; j++)

{

if (array[j] < pivot)

{

i++;

swap(&array[i], &array[j]);

}

}

swap(&array[i + 1], &array[high]);

return (i + 1);

}

void quickSort(int array[], int low, int high)

{

if (low < high)

{

int pi = partition(array, low, high);

quickSort(array, low, pi - 1);

quickSort(array, pi + 1, high);

}

}

int main()

{

int n;

printf("Enter the number of elements: ");

scanf("%d", &n);

int array[n];

printf("Enter the elements:\n");

for (int i = 0; i < n; i++)

{

scanf("%d", &array[i]);

}

quickSort(array, 0, n - 1);

printf("Sorted array:\n");

for (int i = 0; i < n; i++)

{

printf("%d ", array[i]);

}

printf("\n");

return 0;

}

OUTPUT

