#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void merge(int arr[], int left, int mid, int right) {

int i, j, k;

int n1 = mid - left + 1;

int n2 = right - mid;

int L[n1], R[n2];

for (i = 0; i < n1; i++)

L[i] = arr[left + i];

for (j = 0; j < n2; j++)

R[j] = arr[mid + 1 + j];

i = 0;

j = 0;

k = left;

while (i < n1 && j < n2) {

if (L[i] <= R[j]) {

arr[k] = L[i];

i++;

} else {

arr[k] = R[j];

j++;

}

k++;

}

while (i < n1) {

arr[k] = L[i];

i++;

k++;

}

while (j < n2) {

arr[k] = R[j];

j++;

k++;

}

}

void mergeSort(int arr[], int left, int right) {

if (left < right) {

int mid = left + (right - left) / 2;

mergeSort(arr, left, mid);

mergeSort(arr, mid + 1, right);

merge(arr, left, mid, right);

}

}

void generateArray(int arr[], int n) {

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

arr[i] = rand() % 10000;

}

void printArray(int arr[], int n) {

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

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

printf("\n");

}

int main() {

int N;

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

scanf("%d",&N);

int arr[N];

clock\_t start, end;

start = clock();

mergeSort(arr, 0, N - 1);

end = clock();

double time\_taken = ((double)(end - start)) / CLK\_TCK;

printf("Time taken to sort %d elements: %f seconds\n", N, time\_taken);

return 0;

}



#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int partition(int arr[], int low, int high) {

int pivot = arr[low];

int i = low, j = high;

while (i < j) {

while (i <= high && arr[i] <= pivot)

i++;

while (j >= low && arr[j] > pivot)

j--;

if (i < j) {

int temp = arr[i];

arr[i] = arr[j];

arr[j] = temp;

}

}

int temp = arr[low];

arr[low] = arr[j];

arr[j] = temp;

return j;

}

void quickSort(int arr[], int low, int high) {

if (low < high) {

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

quickSort(arr, low, pi - 1);

quickSort(arr, pi + 1, high);

}

}

void generateArray(int arr[], int n) {

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

arr[i] = rand() % 10000;

}

void print(int arr[], int n) {

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

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

if (i < n - 1) {

printf(", ");

}

}

printf("\n");

}

int main() {

int N;

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

scanf("%d", &N);

int arr[N];

generateArray(arr,N);

clock\_t start = clock();

quickSort(arr, 0, N - 1);

clock\_t end = clock();

double time\_taken = ((double)(end - start)) / CLOCKS\_PER\_SEC;

printf("Time taken to sort %d elements: %f seconds\n", N, time\_taken);

return 0;

}

