**Design Analysis and Algorithm**

**Lab Assignment No.: 06**

**Rushikesh Bhivsen Shinde**

**BBCO19134**

**Problem Statement:**

Write a program for analysis of quick sort by using deterministic and randomized variant.

**Code:**

/\* C++ program to find Approximate Median using

1/2 Approximate Algorithm \*/

#include<bits/stdc++.h>

using namespace std;

// This function returns the Approximate Median

int randApproxMedian(int arr[],int n)

{

// Declaration for the random number generator

random\_device rand\_dev;

mt19937 generator(rand\_dev());

// Random number generated will be in the range [0,n-1]

uniform\_int\_distribution<int> distribution(0, n-1);

if (n==0)

return 0;

int k = 10\*log2(n); // Taking c as 10

// A set stores unique elements in sorted order

set<int> s;

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

{

// Generating a random index

int index = distribution(generator);

//Inserting into the set

s.insert(arr[index]);

}

set<int> ::iterator itr = s.begin();

// Report the median of the set at k/2 position

// Move the itr to k/2th position

advance(itr, (s.size()/2) - 1);

// Return the median

return \*itr;

}

// Driver method to test above method

int main()

{

int arr[] = {1, 3, 2, 4, 5, 6, 8, 7};

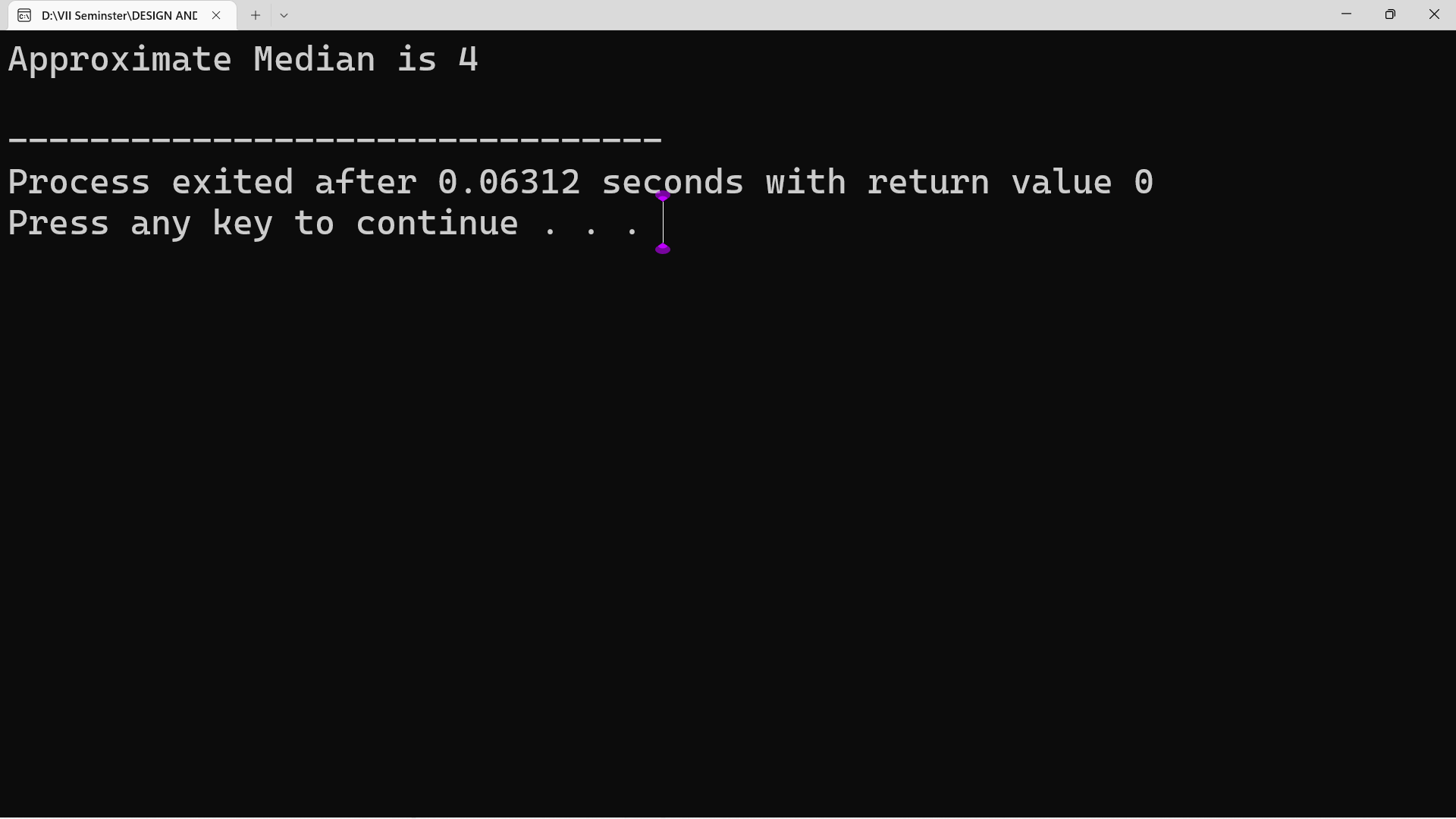
int n = sizeof(arr)/sizeof(int);

printf("Approximate Median is %d\n",randApproxMedian(arr,n));

return 0;

}

**Output:**

****