TW :- 7

Design, develop, code and run the program in any suitable language to implement the quicksort algorithm. Determine the basis paths and using them derive different test cases, execute these test cases and discuss the test results. discuss the test results.

PROGRAM CODE:

#include <stdio.h>

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

{

int t = \*a;

\*a = \*b;

\*b = t;

}

int partition (int arr[], int l, int h)

{

int x = arr[h];

int i = (l - 1),j;

for (j = l; j <= h- 1; j++)

{

if (arr[j] <= x)

{

i++;

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

}

}

swap (&arr[i + 1], &arr[h]);

return (i + 1);

}

void quickSortIterative (int arr[], int l, int h)

{

int stack[10],p;

int top = -1;

stack[ ++top ] = l;

stack[ ++top ] = h;

while ( top >= 0 )

{

h = stack[ top-- ];

int main ()

{

int arr[20],n,i;

clrscr();

printf("Enter the size of the array");

scanf("%d",&n);

printf("Enter %d elements",n);

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

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

quickSortIterative( arr, 0, n - 1 );

printf("Elements of the array are;");

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

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

getch();

return 0;

}

Enter the size of the array : 9

9

Enter 9 elements :

23 5 32 63 12 1 7 65 99

Elements of the array are :

1 5 7 12 23 32 63 65 99

Enter the size of the array: 4

4

Enter 4 elements :

-3 21 3 32

Elements of the array are:

-3 3 21 32