**8. Write an algorithm and program to implement interpolation search.**

#include<iostream>

using namespace std;

int interpolationSearch(int array[], int start, int end, int key) {

int dist, valRange, indexRange, estimate;

float fraction;

while(start <= end && key >= array[start] && key <= array[end]) {

dist = key - array[start];

valRange = array[end] - array[start]; //range of value

fraction = dist / valRange;

indexRange = end - start;

estimate = start + (fraction \* indexRange); //estimated position of the key

if(array[estimate] == key)

return estimate;

if(array[estimate] < key)

start = estimate +1;

else

end = estimate - 1;

}

return -1;

}

int main() {

int n, searchKey, loc;

cout << "Enter number of items: ";

cin >> n;

int arr[n]; //create an array of size n

cout << "Enter items: " << endl;

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

cin >> arr[i];

}

cout << "Enter search key to search in the list: ";

cin >> searchKey;

if((loc = interpolationSearch(arr, 0, n-1, searchKey)) >= 0)

cout << "Item found at location: " << loc << endl;

else

cout << "Item is not found in the list." << endl;

}

**OUTPUT**

