//PROBLEM 1(WEEK 1) **(LINEAR SEARCH)**

#include<stdio.h>

#define MAX 100

void linear\_search(int A[],int n ,int key)

{

int comp = 0;

int flag = 0;

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

{

comp++;

if(A[i] == key)

{

flag = 1;

printf("Element found at index %d",i+1);

break;

}

}

if(flag == 0)

{

printf("Key not found");

}

printf("\nTotal comparisons: %d\n",comp);

}

int main()

{

int key, A[MAX], n, t;

printf("Enter number of test cases: ");

scanf("%d",&t);

while(t--)

{

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

scanf("%d",&n);

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

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

{

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

}

printf("Enter the key: ");

scanf("%d",&key);

linear\_search(A, n, key);

}

}

***OUTPUT:***

Enter number of test cases: 2

Enter the size of the array: 5

Enter the elements of array:1 3 5 7 9

Enter the key: 6

Key not found

Total comparisons: 5

Enter the size of the array: 4

Enter the elements of array:3 7 8 9

Enter the key: 8

Element found at index 3

Total comparisons: 3

//PROBLEM 2(WEEK 1) **(BINARY SEARCH)**

#include<stdio.h>

#define MAX 100

void binary\_search(int A[], int lb, int ub, int key)

{

int comp = 0;

int flag = 0;

while(lb<ub)

{

int mid = (lb+ub)/2;

if(A[mid] == key)

{

comp++;

printf("Element is found at position: %d",mid+1);

printf("\nTotal comparisons: %d",comp);

flag = 1;

break;

}

if(A[mid]>key)

{

comp++;

ub = mid;

}

if(A[mid]<key)

{

comp++;

lb = mid+1;

}

}

if(flag==0)

{

printf("Element not found");

printf("\nTotal comparisons: %d\n",comp);

}

}

int main()

{

int key, A[MAX], n, t;

printf("Enter number of test cases: ");

scanf("%d",&t);

while(t--)

{

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

scanf("%d",&n);

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

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

{

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

}

printf("Enter the key: ");

scanf("%d",&key);

binary\_search(A, 0, n, key);

}

}

***OUTPUT:***

Enter number of test cases: 3

Enter the size of the array: 3

Enter the elements of array: 6 7 8

Enter the key:5

Element not found

Total comparisons: 2

Enter the size of the array: 4

Enter the elements of array: 1 3 4 5

Enter the key:5

Element is found at position: 4

Total comparisons: 2

Enter the size of the array: 5

Enter the elements of array: 4 6 8 9 11

Enter the key:4

Element is found at position: 1

Total comparisons: 3

//PROBLEM 3(WEEK 1) **(JUMP SEARCH)**

#include<stdio.h>

#include<math.h>

#define MAX 100

void jump\_search(int A[],int n, int key)

{

int flag = 0;

int comp = 0;

int start = 0;

int end = pow(2,start);

while(A[end]<key && end<n)

{

int k = 1;

start = end-1;

end = start+pow(2,k);

k++;

}

for(int i = start; i <= end; i++)

{

comp++;

if(A[i] == key)

{

flag = 1;

printf("Present\t%d\n",comp);

break;

}

}

if(flag == 0)

{

printf("Element not found\t %d\n",comp);

}

}

int main()

{

int key, A[MAX], n, t;

printf("Enter the number of test cases: ");

scanf("%d",&t);

while(t--)

{

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

scanf("%d",&n);

printf("Enter the elements in the array: ");

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

{

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

}

printf("Enter the key: ");

scanf("%d",&key);

jump\_search(A, n, key);

}

return 0;

}

***OUTPUT:***

Enter the number of test cases: 2

Enter the size of the array: 5

Enter the elements in the array: 45 46 88 90 95

Enter the key: 95

Present 3

Enter the size of the array: 5

Enter the elements in the array: 4 6 8 23 55

Enter the key: 6

Present 2