Assignment – 15

A Job Ready Bootcamp in C++, DSA and IOT

Array and Functions in C Language

Name: Tushar Maliye Date:21/12/2022

User id: [tusharmaliye59@gmail.com](mailto:tusharmaliye59@gmail.com)

1. Write a function to find the greatest number from the given array of any size. (TSRS)

Program:

#include<stdio.h>

void get\_value(int , int);

int largest\_value(int, int );

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

int greater\_value(int a[],int n)

{

int max=-1;

for(int i=1;i<=n;i++)

{

if(max<a[i])

{

max=a[i];

}

}

return max;

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

printf("The greater value in the array is %d",greater\_value(a,n));

return 0;

}

OUTPUT:

enter the size of array: 6

Enter the 6 values of array: 1 6 7 9 4 3

The greater value in the array is 9

--------------------------------

Process exited after 11.31 seconds with return value 0

Press any key to continue . . .

1. Write a function to sort an array of any size. (TSRS)

Program:

#include<stdio.h>

void get\_value(int , int);

void sorting(int, int );

void print\_array(int ,int );

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

void sorting(int a[],int n)

{

for(int i=1; i<=n; i++)

{

for(int j=i+1;j<=n;j++)

{

if(a[i]>a[j])

{

int temp= a[j];

a[j]=a[i];

a[i]=temp;

}

}

}

}

void print\_array(int a[],int n)

{

for(int i=1;i<=n;i++)

{

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

}

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

sorting(a,n);

print\_array(a,n);

return 0;

}

Output:

enter the size of array: 5

Enter the 5 values of array: 1 9 5 7 3

1 3 5 7 9

--------------------------------

Process exited after 10.08 seconds with return value 0

Press any key to continue . . .

4. Write a function to rotate an array by n position in d direction. The d is an indicative

value for left or right. (For example, if array of size 5 is [32, 29, 40, 12, 70]; n is 2 and

d is left, then the resulting array after left rotation 2 times is [40, 12, 70, 32, 29] )

PROGRAM:

#include<stdio.h>

#include<cstdlib>

void get\_value(int , int);

void left\_shift(int ,int ,int );

void right\_shift(int , int , int );

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

void left\_shift(int a[],int n,int m )

{

for(int i=(m+1);i<=n;i++)

{

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

}

for(int i=1;i<=m;i++)

{

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

}

}

void right\_shift(int a[], int n, int m )

{

for(int i=(n-m+1);i<=n;i++)

{

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

}

for(int i=1;i<=(n-m);i++)

{

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

}

}

int main()

{

int n,m,d;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

while(1)

{

printf(" \n 1. left\n 2. Right\nselect the direction : ");

scanf("%d",&d);

switch (d)

{

case 1: printf("\nEnter the number of times left shifting: ");

scanf("%d",&m);

left\_shift(a,n,m);

break;

case 2: printf("\nEnter the number of times right shifting: ");

scanf("%d",&m);

right\_shift(a,n,m);

break;

default: exit(0);

}

printf("\n-----------------------");

}

return 0;

}

OUTPUT:

enter the size of array: 6

Enter the 6 values of array: 2 6 4 3 7 1

1. left

2. Right

select the direction : 1

Enter the number of times left shifting: 2

4 3 7 1 2 6

-----------------------

1. left

2. Right

select the direction : 2

Enter the number of times right shifting: 2

7 1 2 6 4 3

-----------------------

1. left

2. Right

select the direction : 0

--------------------------------

Process exited after 37.43 seconds with return value 0

Press any key to continue . . .

5. Write a function to find the first occurrence of adjacent duplicate values in the array.

Function has to return the value of the element.

Program:

#include<stdio.h>

#include<cstdlib>

void get\_value(int , int);

int check\_occurrence(int ,int );

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

int check\_occurrence(int a[],int n)

{

int flag;

for(int i=1;i<=n;i++)

{

if(a[i]==a[i+1])

{

return a[i];

}

}

return -1;

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

int flag=check\_occurrence(a,n);

if(flag==(-1))

printf("no occurrence is present");

else

printf("occcurence is present of %d",flag);

return 0;

}

OUTPUT:

enter the size of array: 6

Enter the 6 values of array: 1 5 9 9 8 7

occcurence is present of 9

--------------------------------

Process exited after 14.08 seconds with return value 0

Press any key to continue . . .

6. Write a function in C to read n number of values in an array and display it in reverse

order.

Program:

#include<stdio.h>

void get\_value(int,int);

void print\_reverse\_value(int,int);

void get\_value(int a[],int n)

{

printf(" Enter the %d values of the array: \n ",n);

for(int i=1;i<=n;i++)

{

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

}

}

void print\_reverse\_value(int a[],int n)

{

printf("\n ",n);

for(int i=n;i>0;i--)

{

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

}

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

print\_reverse\_value(a,n);

printf("\n");

return 0;

}

Output:

Enter the size of array: 6

Enter the 6 values of the array:

1 9 7 6 4 3

3 4 6 7 9 1

--------------------------------

Process exited after 10.46 seconds with return value 0

Press any key to continue . . .

7. Write a function in C to count a total number of duplicate elements in an array.

PROGRAM:

#include<stdio.h>

void get\_value(int,int);

void repeated\_value(int,int);

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

void repeated\_value(int a[],int n)

{

int i,j,count=0;

for(i=1;i<=n;i++)

{

for(j=i+1;j<=n;j++)

{

if(a[i]==a[j])

{

count++;

}

}

}

printf("total number of repeated values are : %d",count);

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

repeated\_value(a,n);

printf("\n");

return 0;

}

Output:

Enter the size of array: 6

Enter the 6 values of array: 1 2 3 1 2 3

total number of repeated values are : 3

--------------------------------

Process exited after 9.858 seconds with return value 0

Press any key to continue . . .

8. Write a function in C to print all unique elements in an array.

Program:

#include<stdio.h>

void get\_value(int,int);

void unique\_value(int,int);

void print\_value(int , int);

void get\_value(int a[], int n)

{

printf(" Enter the %d values of array: ",n);

for(int i=1;i<=n;i++)

{

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

}

}

void print\_value(int a[], int n)

{

for(int i=1;i<=n;i++)

{

if(a[i]!=1)

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

}

}

void unique\_value(int a[],int n)

{

int i,j,count=0;

for(i=1;i<=n;i++)

{

for(j=i+1;j<=n;j++)

{

if(a[i]==a[j])

{

a[i]=1;

a[j]=1;

}

}

}

printf("unique values are: ");

print\_value(a,n);

}

int main()

{

int n;

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

scanf("%d",&n);

int a[n];

get\_value(a,n);

unique\_value(a,n);

printf("\n");

return 0;

}

Output:

Enter the size of array: 6

Enter the 6 values of array: 1 2 5 4 2 1

unique values are: 5 4

--------------------------------

Process exited after 14.87 seconds with return value 0

Press any key to continue . . .

9. Write a function in C to merge two arrays of the same size sorted in descending

order.