Assignment - 15 A Job Ready Bootcamp in C++, DSA and IOT

Array and Functions in C Language

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

#include <stdio.h>

int i, n;

int sg(int a[n])

{

int g = -9999;

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

{

if (g < a[i])

{

g = a[i];

}

}

return g;

}

int main()

{

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

scanf("%d", &n);

int a[n];

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

{

printf("Enter a number:\n");

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

}

printf("%d is the greatest number in the array\n", sg(a));

return 0;

}

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

#include <stdio.h>

int i, n, s = 99999;

int ss(int arr[])

{

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

{

if (s > arr[i])

s = arr[i];

}

return s;

}

int main()

{

printf("Enter the size of the array you want:\n");

scanf("%d", &n);

int arr[n];

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

{

printf("Enter a number:\n");

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

}

printf("%d is the smallest number in the array.\n", ss(arr));

return 0;

}

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

#include <stdio.h>

int i, j, n;

int sort(int arr[])

{

int t;

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

{

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

{

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

{

t = arr[i];

arr[i] = arr[j];

arr[j] = t;

}

}

}

return arr[n];

}

int main()

{

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

scanf("%d", &n);

int arr[n];

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

{

printf("Enter a number:\n");

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

}

sort(arr);

printf("The sorted arry:");

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

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

printf("\n");

return 0;

}

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] )

#include <stdio.h>

int i, r, n;

int shift(int arr[n])

{

printf("Enter the digit you want to shift:\n'+' for right rotation '-'left rotation \n");

scanf("%d", &r);

if (r > 0) // for right shifting

{

while (r != 0)

{

int t = arr[n - 1];

for (i = n - 1; i > 0; i--)

{

arr[i] = arr[i - 1];

}

arr[0] = t;

r--;

}

}

else // for left shifting

{

while (r != 0)

{

int t = arr[0];

for (i = 0; i < n - 1; i++)

{

arr[i] = arr[i + 1];

}

arr[n - 1] = t;

r++;

}

}

return arr[n];

}

int main()

{

printf("Enter the size of the arrary:\n");

scanf("%d", &n);

int arr[n];

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

{

printf("Enter a number:\n");

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

}

shift(arr);

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

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

}

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.

#include <stdio.h>

#include <stdlib.h>

int i, n;

int dpl(int arr[n])

{

for (i = 0; i < n - 1; i++)

{

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

return arr[i];

}

printf("\nThere is no duplicate values in the array\n");

exit;

}

int main()

{

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

scanf("%d", &n);

int arr[n];

printf("Enter the elemets:\n");

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

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

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

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

printf("\n%d is the first duplicate adjacent value", dpl(arr));

printf("\n");

return 0;

}

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

#include <stdio.h>

int i, j, n;

void rev(int arr[n])

{

int brr[n];

j = n - 1;

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

{

brr[j] = arr[i];

j--;

}

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

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

}

int main()

{

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

scanf("%d", &n);

int arr[n];

printf("Enter the elemets:\n");

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

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

rev(arr);

printf("\n");

return 0;

}

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

#include <stdio.h>

int e, i, n, count;

int ndpl(int arr[n])

{

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

{

count = 0;

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

{

if (arr[i] == arr[e])

count++;

}

printf("There is %d number of %d in the array\n", count, arr[i]);

}

}

int main()

{

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

scanf("%d", &n);

printf("Enter the elements:\n");

int arr[n];

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

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

ndpl(arr);

printf("\n");

return 0;

}

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

#include <stdio.h>

int i, j, n, count;

int unq(int arr[n])

{

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

{

count = 0;

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

{

if (i != j && arr[i] == arr[j])

count++;

}

if (count == 0)

printf("%d is a unique element in the array\n", arr[i]);

}

}

int main()

{

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

scanf("%d", &n);

int arr[n];

printf("Enter the elements:\n");

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

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

unq(arr);

printf("\n");

return 0;

}

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

#include <stdio.h>

void mergeArrays(int n, int a[n], int b[n], int result[2 \* n])

{

int i = 0, j = 0, k = 0;

while (i < n && j < n)

{

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

{

result[k] = a[i];

k++;

i++;

}

else

{

result[k] = b[j];

k++;

j++;

}

}

while (i < n)

{

result[k++] = a[i++];

}

while (j < n)

{

result[k++] = b[j++];

}

}

void sortArray(int l, int x[l])

{

int i, j, temp;

for (i = 0; i < l - 1; i++)

{

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

{

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

{

temp = x[i];

x[i] = x[j];

x[j] = temp;

}

}

}

}

int main()

{

int n, i;

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

scanf("%d", &n);

int a[n], b[n], result[2 \* n];

printf("Enter the elements of the first array: \n");

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

{

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

}

printf("Enter the elements of the second array: \n");

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

{

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

}

mergeArrays(n, a, b, result);

sortArray(2\*n, result);

printf("The merged array in descending order is: \n");

for (i = 0; i < 2 \* n; i++)

{

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

}

printf("\n");

return 0;

}

10. Write a function in C to count the frequency of each element of an array.

#include <stdio.h>

int sg(int n, int arr[n])

{

int i, g = -9999;

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

{

if (g < arr[i])

{

g = arr[i];

}

}

return g;

}

int main()

{

int i, n, m;

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

scanf("%d", &n);

int arr[n];

printf("Enter %d numbers:\n", n);

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

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

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

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

printf("\n%d is the greatest number in the array\n", m = sg(n, arr));

int frr[m + 1];

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

frr[i] = 0;

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

frr[arr[i]]++;

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

{

if (frr[i])

printf("Number of %d is %d\n", i, frr[i]);

}

printf("\n");

return 0;

}