***Assignment 12 || Dynamic Memory Allocation***

Arjun Patel – FRN006

Q1)Find max and min element.

#include <stdio.h>

#include <stdlib.h>

void storeArr(int \*arr, int size)

{

    for (int i = 0; i < size; i++)

    {

        printf("Enter value for index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void printArr(int arr[], int size)

{

    printf("[ ");

    for (int i = 0; i < size; i++)

    {

        printf("%d ,", arr[i]);

    }

    printf("\b ]");

}

int \*findMaxNMinEle(int arr[], int size)

{

    int \*maxMinArr = (int \*)malloc(2 \* sizeof(int));

    int max = arr[0], min = arr[0];

    for (int i = 0; i < size; i++)

    {

        if (arr[i] < min)

        {

            min = arr[i];

        }

        if (arr[i] > max)

        {

            max = arr[i];

        }

    }

    maxMinArr[0] = min;

    maxMinArr[1] = max;

    return maxMinArr;

}

int main()

{

    int n;

    printf("\nEnter the size of array\n");

    scanf("%d", &n);

    int arr[n];

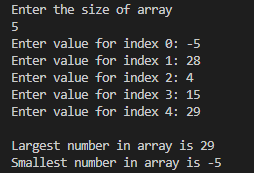
    storeArr(arr, n);

    printf("\nLargest number in array is %d\n", findMaxNMinEle(arr, n)[1]);

    printf("Smallest number in array is %d\n", findMaxNMinEle(arr, n)[0]);

    return 0;

}



Q2)Search element

#include <stdio.h>

#include <stdlib.h>

void storeArr(int \*arr, int size)

{

    for (int i = 0; i < size; i++)

    {

        printf("Enter value for index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

int searchInArray(int arr[], int size, int num)

{

    for (int i = 0; i < size; i++)

        if (arr[i] == num)

            return i;

    return -1;

}

int main()

{

    int n;

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

    scanf("%d", &n);

    int searchNum;

    int\* arr = (int\*)malloc(n \* sizeof(int));

    storeArr(arr, n);

    printf("\nEnter a number u want to search in array\n");

    scanf("%d", &searchNum);

    int res = searchInArray(arr, n, searchNum);

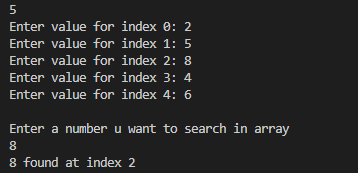
    res==-1 ? printf("Number not found in array\n") : printf("%d found at index %d",searchNum, res);

    free(arr);

    arr=NULL;

    return 0;

}



Q3) Sum of all nums in array

#include<stdio.h>

#include<stdlib.h>

int storeNSum(int arr[], int size){

    int sum = 0;

    for (int i = 0; i < size; i++)

    {

        printf("Enter the value at index %d: ", i);

        scanf("%d", &arr[i]);

        sum += arr[i];

    }

    return sum;

}

int main(){

    int n;

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

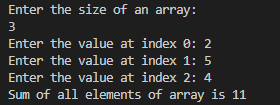
    scanf("%d", &n);

    int\* arr = (int\*)malloc(n \* sizeof(int));

    printf("Sum of all elements of array is %d", storeNSum(arr,n));

    return 0;

}



Q4) Odd even num

#include<stdio.h>

#include<stdlib.h>

void storeArr(int \*arr, int size)

{

    for (int i = 0; i < size; i++)

    {

        printf("Enter value for index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void printOddEven(int arr[], int size){

    printf("\n--------Even Nums in Array-------\n");

    for (int  i = 0; i < size; i++)

        !(arr[i]%2) && printf("%d ", arr[i]);

    printf("\n--------Odd Nums in Array-------\n");

    for (int  i = 0; i < size; i++)

        arr[i]%2 && printf("%d ", arr[i]);

}

int main(){

    int n;

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

    scanf("%d", &n);

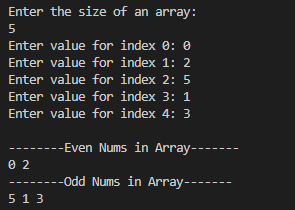
    int\* arr = (int\*)malloc(n \* sizeof(int));

    storeArr(arr, n);

    printOddEven(arr, n);

    return 0;

}



Q5)Print alternate numbers

#include<stdio.h>

void storeArr(int \*arr, int size)

{

    for (int i = 0; i < size; i++)

    {

        printf("Enter value for index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void printAlternate(int arr[], int size){

    for (int i = 0; i < size; i=i+2)

    {

        printf("%d ", arr[i]);

    }

}

int main(){

    int n;

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

    scanf("%d", &n);

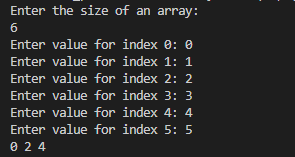
    int arr[n];

    storeArr(arr, n);

    printAlternate(arr, n);

    return 0;

}



Q6)Print Prime numbers

#include <stdio.h>

#include <stdlib.h>

void checkPrime(int arr[], int size)

{

    for (int i = 0; i < size; i++)

    {

        if(arr[i]==1 || arr[i]==0) continue;

        int isPrime = 1;

        for (int j = 2; j \* j <= arr[i]; j++)

        {

            if (arr[i] % j == 0)

            {

                isPrime = 0;

                break;

            }

        }

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

    }

}

int main()

{

    int n;

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

    scanf("%d", &n);

    int\* arr = (int\*)malloc(n \* sizeof(int));

    for (int i = 0; i < n; i++)

    {

        printf("Enter the value at index %d: ", i);

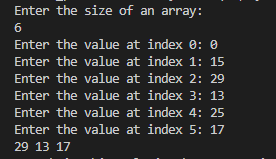
        scanf("%d", &arr[i]);

    }

    checkPrime(arr, n);

    return 0;

}



Q7) Add sum in third arr

/\*

7. Take two array and add sum in third array

Examplearr[

5]= {1,2, 3, 4,5}

brr[5]={10,20,30, 40, 50}

crr[5]={11,22,33,44,55}

\*/

#include<stdio.h>

#include<stdlib.h>

void storeArr(int arr[], int size){

    for (int i = 0; i < size; i++)

    {

        printf("Enter the value at index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void findSum(int arr[], int brr[], int size){

    int crr[size];

    for (int i = 0; i < size; i++)

    {

        crr[i] = arr[i]+ brr[i];

        printf("%d ", crr[i]);

    }

}

int main(){

    int n;

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

    scanf("%d", &n);

**int\* arr = (int\*)malloc(n \* sizeof(int));**

    int sum = 0;

    //taking arr 1 from user

    printf("------Array 1 - arr --------\n");

    storeArr(arr,n);

    int\* brr = (int\*)malloc(n \* sizeof(int));

    int\* crr = (int\*)malloc(n \* sizeof(int));

    //taking brr (2) from user

    printf("------Array 2 - brr --------\n");

    storeArr(brr,n);

    //find sum and adding in crr in respective element

    findSum(arr,brr, n);

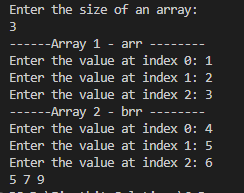
    free(arr);

    free(brr);

    free(crr);

    return 0;

}



Q8)Merge two arrays

#include<stdio.h>

#include<stdlib.h>

void storeArr(int arr[], int size){

    for (int i = 0; i < size; i++)

    {

        printf("Enter the value at index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void displayArr(int arr[], int size){

    for (int i = 0; i < size; i++)

    {

        printf("%d ", arr[i]);

    }

}

int\* mergeArr(int arr1[], int size1, int arr2[], int size2){

    int n3 = size1+size2;

**int\* arr3 = (int\*) malloc(n3 \* sizeof(int));**

    for (int i = 0; i < size1+size2 ; i++)

    {

        arr3[i] = i<size1 ? arr1[i] : arr2[i-size1];

    }

    return arr3;

}

int main(){

    int n1;

    printf("Enter the size of an array1:\n");

    scanf("%d", &n1);

    int arr1[n1];

    //taking arr 1 from user

    printf("-----Enter values for Array 1-----\n");

    storeArr(arr1, n1);

    int n2;

    printf("Enter the size of an array1:\n");

    scanf("%d", &n2);

    int arr2[n2], n3 = n1+n2;

    //taking arr 2 from user

    printf("-----Enter values for Array 2-----\n");

    storeArr(arr2,n2);

    //adding values from arr1 to arr3

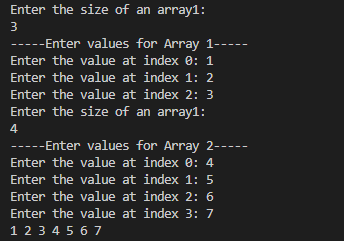
    int\* arr3 = mergeArr(arr1, n1, arr2, n2);

    //printing arr3

    displayArr(arr3, n3);

    return 0;

}



Q9)Reverse an array

    #include <stdio.h>

    #include <stdlib.h>

    void displayArr(int arr[], int size)

    {

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

            printf("%d ", arr[i]);

    }

    void storeArr(int arr[], int size)

    {

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

        {

            printf("Enter the value for index %d: ", i);

            scanf("%d", &arr[i]);

        }

    }

    void reverseArray(int arr[], int n)

    {

        for (int i = 0; i <= n / 2; i++)

        {

            int temp = arr[i];       // 1 2

            arr[i] = arr[n - i - 1]; // a[0] = a[3-0-1] = a[2] = 3 2

            arr[n - i - 1] = temp;   // a[2] = temp = 1 2

            // printf("%d ", arr[i]);

        }

    }

    int main()

    {

        int n;

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

        scanf("%d", &n);

**int\* arr = (int\*) malloc(n \* sizeof(int));**

        // taking arr from user

        printf("-----Enter values for Array 1-----\n");

        storeArr(arr, n);

        // reverse array

        reverseArray(arr, n);

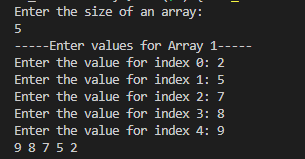
        // printing array

        displayArr(arr, n);

        free(arr);

        return 0;

    }



Q10)Sort an array

#include <stdio.h>

#include <stdlib.h>

void displayArr(int arr[], int size)

{

    for (int i = 0; i < size; i++)

        printf("%d ", arr[i]);

}

void storeArr(int arr[], int size)

{

    for (int i = 0; i < size; i++)

    {

        printf("Enter the value for index %d: ", i);

        scanf("%d", &arr[i]);

    }

}

void sortArray(int arr[], int n)

{

    for (int i = 0; i < n; i++)

    {

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

        {

            if (arr[i] > arr[j])

            {

                int temp = arr[i];

                arr[i] = arr[j];

                arr[j] = temp;

            }

        }

    }

}

int main()

{

    int n;

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

    scanf("%d", &n);

**int\* arr = (int\*) malloc(n \* sizeof(int));**

    // taking arr from user

    storeArr(arr,n);

    //sort array

    sortArray(arr, n);

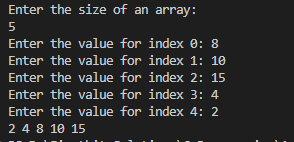
    // printing array

    displayArr(arr,n);

    free(arr);

    return 0;

}



***-----END-----***