DAA Lab

Name：Prashant Regmi

Roll No: 21053305

Section: 4

Subject: DAA Lab

Lab Day: 1

QUESTIONS

Aim of the Experiment 1: To store random numbers into an array of N integers and then find out the smallest and largest number stored in it. N is the user input.

```c
#include <stdio.h>
int main(){
   int max;
   int min;
   int n;
   printf("Enter the number of elements : ");
   scanf("%d",&n);
   int arr_3305[n];
   printf("Enter the number in the array : ");

   for(int i=0;i<n;i++){
       scanf("%d",&arr_3305[i]);
   }
   max=arr_3305[0];
   min=arr_3305[0];
   for(int i=0;i<n;i++){
       if(arr_3305[i] < min){
           min=arr_3305[i];
       }
       if(arr_3305[i] > max){
           max=arr_3305[i];
       }
   }

   // display
   printf("The arrray is ");
   for(int i=0;i<n;i++){
       printf("%d ",arr_3305[i]);
   }
   printf("\n");
   printf("The maximum element in the array is : %d\n",max);
   printf("The minimum element in the array is : %d\n",min);
   return 0;
   }
```

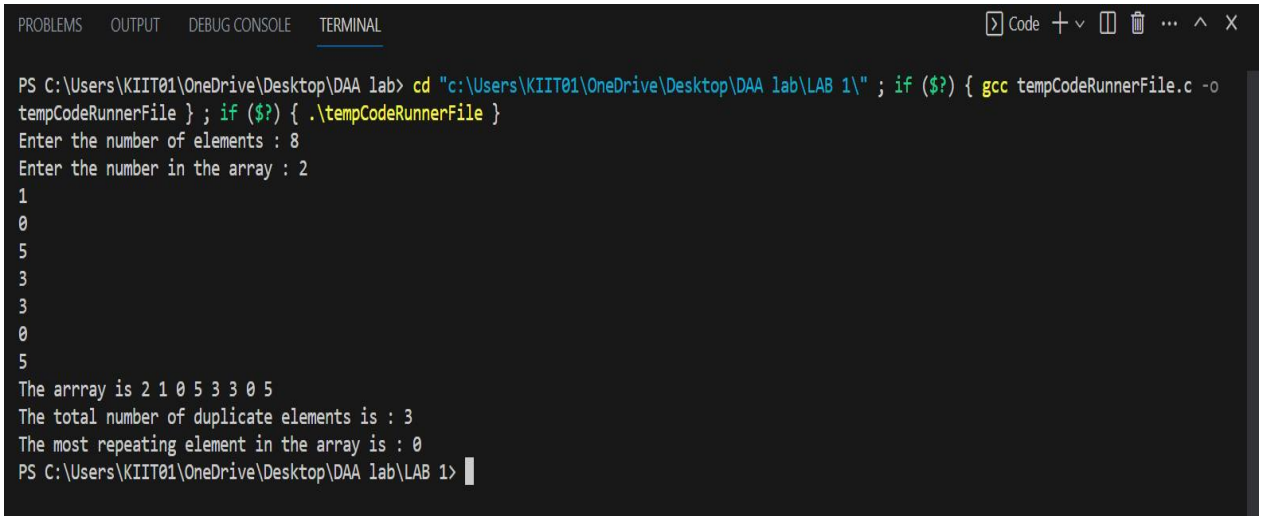
Output

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL                                   Code

PS C:\Users\KIIT01\OneDrive\Desktop\DAA lab> cd "c:\Users\KIIT01\OneDrive\Desktop\DAA lab\LAB 1\" ; if ($?) { gcc tempCodeRunnerFile.c -o
tempCodeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Enter the number of elements : 8
Enter the number in the array : 2 1 0 5 3 3 0 5
The arrray is 2 1 0 5 3 3 0 5
The maximum element in the array is : 5
The minimum element in the array is : 0
PS C:\Users\KIIT01\OneDrive\Desktop\DAA lab\LAB 1>
```

Aim of the Experiment 2: To store random numbers into an array of N integers, where the array must contains some duplicates and then do the following:
a) Find out the total number of duplicate elements.
b) Find out the most repeating element in the array.

```c
#include <stdio.h>
int main(){
    int n;
    printf("Enter the number of elements : ");
    scanf("%d",&n);
    int arr_3305[n];
    printf("Enter the number in the array : ");

    for(int i=0;i<n;i++){
        scanf("%d",&arr_3305[i]);
    }
    //to dispaly the array
    printf("The arrray is ");
    for(int i=0;i<n;i++){
        printf("%d ",arr_3305[i]);
    }
    printf("\n");
    //a) Find out the total number of duplicate elements.
    int count=0;
    for(int i=0;i<n;i++){
        for(int j=i+1;j<n;j++){
            if(arr_3305[i]==arr_3305[j]){
                count++;
            }
        }
    }
    printf("The total number of duplicate elements is : %d\n",count);
    //b) Find out the most repeating element in the array.
    int max=0;
    int max_count=0;
    for(int i=0;i<n;i++){
        int count=0;
        for(int j=i+1;j<n;j++){
            if(arr_3305[i]==arr_3305[j]){
                count++;
            }
        }
        if(count>max_count){
            max_count=count;
            max=arr_3305[i];
        }
    }
    printf("The most repeating element in the array is : %d\n",max);
}
```

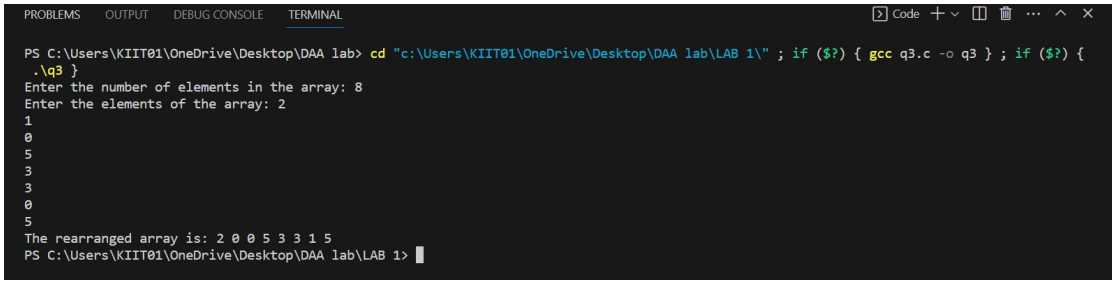
Output



Aim of the Experiment 3: To rearrange the elements of an array of N integers such that all even numbers are followed by all odd numbers

```c
#include <stdio.h>
int main()
{
    int n, i, j, temp;
    printf("Enter the number of elements in the array: ");
    scanf("%d", &n);
    int arr_3305[n];
    printf("Enter the elements of the array: ");
    for (i = 0; i < n; i++)
        scanf("%d", &arr_3305[i]);
    for (i = 0; i < n; i++)
    {
        if (arr_3305[i]%2!= 0)
        {
            for (j=i+1; j<n; j++)
            {
                if (arr_3305[j] % 2 == 0)
                {
                    temp = arr_3305[i];
                    arr_3305[i] = arr_3305[j];
                    arr_3305[j] = temp;
                    break;
                }
            }
        }
    }
    printf("The rearranged array is: ");
    for (i = 0; i < n; i++)
        printf("%d ", arr_3305[i]);
    return 0;
}
```

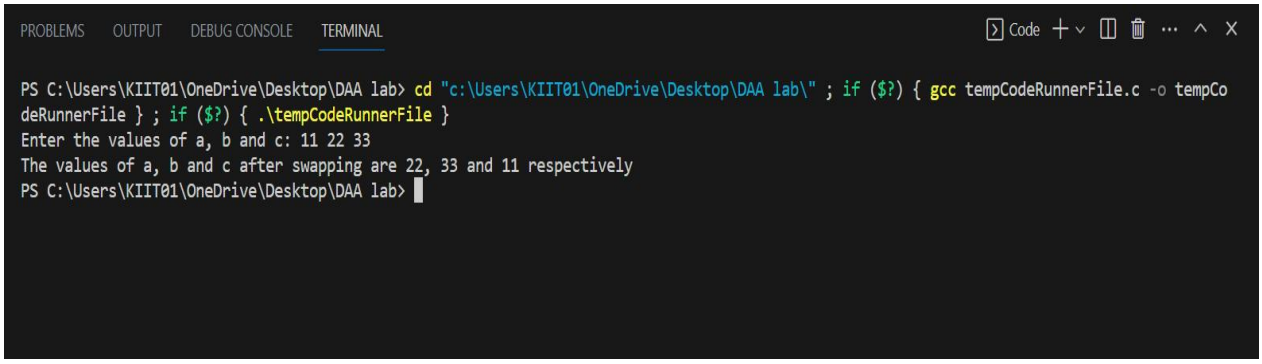
Output



Aim of the Experiment 4: To take three variable (a, b, c) as separate parameters and swapvalues stored so that value a goes to b, b to c and c to a by using a SWAP(x,y) function that swaps/exchanges the numbers x and y.

```c
#include <stdio.h>

void swap(int *a,int *b)
{
   int temp_3305=*a;
   *a=*b;
   *b=temp_3305;
}
int main()
{
   int a_3305,b_3305,c_3305;
   printf("Enter the values of a, b and c: ");
   scanf("%d%d%d",&a_3305,&b_3305,&c_3305);
   swap(&a_3305,&b_3305);
   swap(&b_3305,&c_3305);
   printf("The values of a, b and c after swapping are %d, %d and %d
          respectively\n",a_3305,b_3305,c_3305);
   return 0;
}
```

Output



Aim of the Experiment 5: To take A an n*n square matrix array as an input and then WAP by using appropriate user defined functions for the following:
a) Find the number of nonzero elements in A
b) Find the sum of the elements above the leading diagonal.
c) Display the elements below the minor diagonal.
d) Find the product of the diagonal elements.

```c
#include <stdio.h>
int main(){
    int n;
    printf(" Enter the matrix row/column: ");
    scanf("%d",&n);
    int arr_3305[n][n];
    printf("Enter the elements in the array : ");
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            scanf("%d",&arr_3305[i][j]);
        }
    }
    //to display the array
    printf("The array is \n");
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            printf("%d ",arr_3305[i][j]);
        }
        printf("\n");
    }
    // nonzero elements in A
    int count=0;
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            if(arr_3305[i][j]!=0){
                count++;
            }
        }
    }
    printf("The number of nonzero elements in the array
is : %d\n",count);
    // sum of the elements above the leading diagonal.
    int sum=0;
    for(int i=0;i<n;i++){
        for(int j=i+1;j<n;j++){
            sum+=arr_3305[i][j];
        }
    }
    printf("The sum of the elements above the leading diagonal
is : %d\n",sum);

    // display the elements below the minor diagonal.
    printf("The elements below the minor diagonal are : ");
    for(int i=0;i<n;i++){
        for(int j=0;j<i;j++){
            printf("%d ",arr_3305[i][j]);
        }
    }
    printf("\n");
    //product of the diagonal elements.
    int product=1;
```

```
    for(int i=0;i<n;i++){
        for(int j=0;j<n;j++){
            if(i==j){
                product*=arr_3305[i][j];
            }
        }
    }
    printf("The product of the diagonal elements is : %d\n",product);
}
```

Output