Week: 02.01.2024- 13.01.2024

13.Maximum and minimum number :

```
#include <stdio.h>
 #include <conio.h>


int main()
{
   int a[1000],i,n,min,max;

   printf("Enter size of the array : ");
   scanf("%d",&n);

   printf("Enter elements in array : ");
   for(i=0; i<n; i++)
   {
      scanf("%d",&a[i]);
   }

   min=max=a[0];
   for(i=1; i<n; i++)
   {
        if(min>a[i])
                        min=a[i];
                         if(max<a[i])
                          max=a[i];
   }
    printf("minimum of array is : %d",min);
         printf("\nmaximum of array is : %d",max);


   return 0;
}
```

14. Third largest element:

```
#include<stdio.h>
void thirdLargest(int arr[],int arr_size);
int main()
{
```

```
    int n,a[1000],i;
    scanf("%d",&n);
    for(i=0;i<n;i++)
    {
        scanf("%d",&a[i]);
    }
    thirdLargest(a,n);
    return 0;
}
void thirdLargest(int arr[],int arr_size)
{
    int i,j,temp;
    for(i=0;i<arr_size-1;i++)
    {
        for(j=0;j<arr_size-i-1;j++)
        {
            if(arr[j]>arr[j+1])
            {
                temp=arr[j];
                arr[j]=arr[j+1];
                arr[j+1]=temp;
            }
        }
    }
    printf("The third Largest element is %d",arr[arr_size-3]);
}
```

16. Search Elements in array :

```
#include <stdio.h>
int main()
{
    int arr[50],i,n,ele;
    printf("enter the size of array:");
    scanf("%d",&n);
    printf("elements in array:");
    for(i=0;i<n;i++)
    {
        scanf("%d",&arr[i]);
    }
    printf("enter element to search:");
    scanf("%d",&ele);
    for(i=0;i<n;i++)
```

```
    {
        if(arr[i]==ele)
        {
            printf("%d found at position %d",ele,i+1);
            return 0;
        }
    }
    printf("element not found");
}
```

17. Missing numbers:

```
 #include <stdio.h>
 void findMissing(int arr[], int N)
{
     int temp[N + 1];

    for (int i = 0; i <= N; i++) {

        temp[i] = 0;

    }


    for (int i = 0; i < N; i++) {

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

    }


    int ans;

    for (int i = 0; i <= N; i++) {

        if (temp[i] == 0)

            ans = i + 1;

    }

    printf("%d", ans);
}
 int main()
In{
```

```
    int arr[] = { 1, 3, 7, 5, 6, 2 };

    int n = sizeof(arr) / sizeof(arr[0]);

    findMissing(arr, n);
}
```

18. Repeated numbers:

```
#include<stdio.h>

 void main()
 {
  int i,arr[50],j,no;

  printf("Enter size of array: ");
  scanf("%d",&no);
  printf("Enter any %d elements in array: ",no);
  for(i=0;i<no;i++)
  {
   scanf("%d",&arr[i]);
  }
  printf("Duplicate elements are: ");
  for(i=0; i<no; i++)
   {
    for(j=i+1;j<no;j++)
    {
    if(arr[i]==arr[j])
    {
    printf("%d\n",arr[i]);
    }
   }
   }

 }
```

19. Sort 0s 1s 2s

```
#include <stdio.h>

void swap(int* a, int* b);
```

```
void sort012(int a[], int arr_size)
{
    int lo = 0;
    int hi = arr_size - 1;
    int mid = 0;
        while (mid <= hi) {
        switch (a[mid]) {
            // If the element is 0
        case 0:
            swap(&a[lo++], &a[mid++]);
            break;
            // If the element is 1
        case 1:
            mid++;
            break;
            // If the element is 2
        case 2:
            swap(&a[mid], &a[hi--]);
            break;
        }
    }
}
```

```
void swap(int* a, int* b)
{

    int temp = *a;

    *a = *b;

    *b = temp;
}


void printArray(int arr[], int arr_size)
{

    int i;

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

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

int main()
{

    int arr[] = { 0, 1, 1, 0, 1, 2, 1, 2, 0, 0, 0, 1 };

    int arr_size = sizeof(arr) / sizeof(arr[0]);

    int i;


    sort012(arr, arr_size);


    printArray(arr, arr_size);


    getchar();

    return 0;
}
```

20. Numbers are same or not:

```
include <stdio.h>
int main()
{
    int n1, n2, i , j, count = 0;
    printf("enter size of array 1 : ");
    scanf("%d",&n1);
    int arr1[n1];
    printf("enter elements of array 1 : ");
    for(i=0; i<n1; i++)
    {
        scanf("%d",&arr1[i]);
    }
    printf("enter size of array 2 : ");
    scanf("%d",&n2);
    int arr2[n2];
    printf("enter elements of array 2 : ");
    for(i=0; i<n2; i++)
    {
        scanf("%d",&arr2[i]);
    }
    for(i=0; i<n1; i++)
    {
        for(j=0; j<n2; j++)
        {
            if(arr1[i]==arr2[j])
            {
                count++;
                break;
            }
        }
    }
    if((count==n1)&&(count==n2))
    {
        printf("Arrays are same");
    }
    else
    {
        printf("Arrays are not same");
    }
    return 0;
}
```

21. Rotate by 1

```
include<stdio.h>

int main()
{
   int arr[] = { 10, 20, 30, 40, 50, 60, 70};
   int n = sizeof(arr)/sizeof(arr[0]);
   int k = 3;

   int temp[k];
   for(int i=0; i<k; i++)
     temp[i] = arr[i];

   int x = k;
   for(int i=0; x < n; i++){
       arr[i] = arr[x++];
   }
   x = 0;

   for(int i=n-k; i<n; i++)
      arr[i] = temp[x++];
   for (int i = 0; i < n; i++)
       printf("%d ", arr[i]);

   return 0;
}
```

22. Rotate by k

```
#include <stdio.h>

void rotateArray(int arr[], int n, int k) {
   // Create a temporary array to store rotated elements
   int temp[k];

   // Store the last k elements in the temporary array
   for (int i = 0; i < k; i++) {
       temp[i] = arr[n - k + i];
   }

   // Shift the remaining elements to the right
```

```
    for (int i = n - 1; i >= k; i--) {
        arr[i] = arr[i - k];
    }

    // Copy the temporary array elements to the beginning of the array
    for (int i = 0; i < k; i++) {
        arr[i] = temp[i];
    }
}

int main() {
    int n, k;

    printf("Enter the size of the array: ");
    scanf("%d", &n);

    int arr[n];

    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    printf("Enter the number of positions to rotate by: ");
    scanf("%d", &k);

    // Rotate the array
    rotateArray(arr, n, k);

    // Print the rotated array
    printf("Rotated array:\n");
    for (int i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }

    return 0;
}
```

21.b Rotate array by 1

```
#include <stdio.h>
 void rotate(int arr[], int n)
{
```

```
    // store the last element in a variable

    int last_el = arr[n - 1];

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

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

    // assign the last element to first element

    arr[0] = last_el;
}


int main()
{

    int arr[] = { 1, 2, 3, 4, 5 }, i;

    int n = sizeof(arr) / sizeof(arr[0]);


    printf("Given array is\n");

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

        printf("%d ", arr[i]);
        rotate(arr, n);
      printf("\nRotated array is\n");

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

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


    return 0;
```