CODE: CSA0238

NAME: LOSHINI.L

REG NO: 192571015

Day : Arrays (6-8-2025)

1. **Write a program to read and print elements of an array.**

   IPO

- **Input:**
  Get a value as input say n
- **Process:**

  Use a loop to read n elements and store them in the array

  Use another loop to print all the elements
- **Output:**

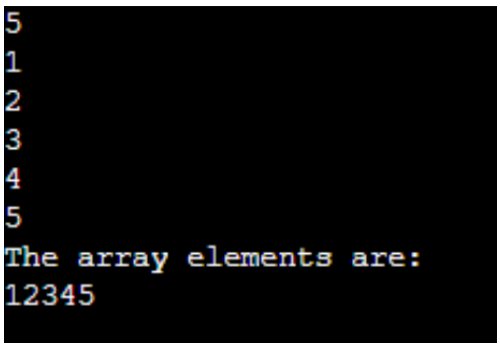
  The output is to print elements of an array.

  Program

```
#include <stdio.h>
void main()
{
   int arr[100],i,n;

   scanf("%d",&n);

   for(i=0;i<n;i++)
   {
      scanf("%d",&arr[i]);
   }
   printf("The array elements are:\n");
   for(i=0;i<n;i++)
```

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

Output



2. Write a program to find the sum of elements of an array.

IPO

- **Input:**
  Get a value as input say n
  Get n elements and store them in the array
- **Process:**

  Use a loop to read n elements and store them in the array

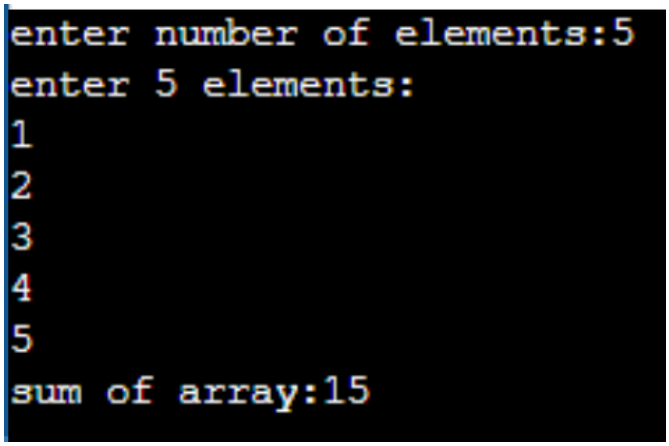
  Use another loop to calculate the sum of all elements
- **Output:**
  The output is to print the sum of elements of the array

Program

```
#include<stdio.h>
void main()
{
   int a[5]={1,2,3,4,5};
   int i,n,sum=0;
```

```
printf("enter number of elements:");
scanf("%d",&n);
printf("enter %d elements:\n",n);
for(i=0;i<n;i++)
{
   scanf("%d",&a[i]);
   sum=sum+a[i];
}
printf("sum of array:%d\n",sum);
}
```

Output



3. Write a program to find the maximum and minimum element in an array.

IPO

- **Input:**
  Get a value as input say n
  Get n elements and store them in the array

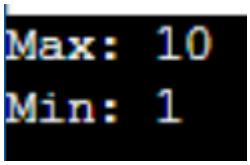
- **Process:**
  Use a loop to read n elements and store them in the array
  Initialize max and min to the first element of the array
  Traverse the array to compare and update max and min

- **Output:**
  The output is to print the maximum and minimum element in the array

  Program

```
#include <stdio.h>
void main()
{
   int arr[5] = {1, 3, 2, 5, 10};
   int n=5;
   int i,max,min;
   max=arr[0];
   for(i=0;i<n;i++)
   {
      if(arr[i]>max)
      {
         max=arr[i];
      }
   }
   min=arr[0];
   for(i=0;i<n;i++)
   {
      if(arr[i]<min)
      {
         min=arr[i];
      }
   }

   printf("Max: %d\n", max);
   printf("Min: %d\n", min);
}
```

  Output



4. Write a program to reverse an array.

IPO

- Input:
  Get a value as input say n
  Get n elements and store them in the array

- Process:

  Use a loop to read n elements and store them in the array

  Use another loop to print elements from the last index to the first index

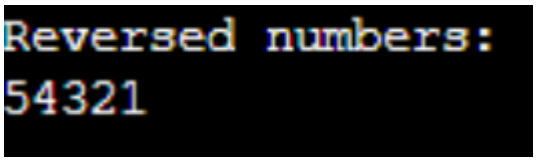
- Output:
  The output is to print the array elements in reverse order

  Program

```
#include <stdio.h>
void main()
{
   int arr[100]={1,2,3,4,5};
   int n=5, i;
   for(i = 0; i < n; i++)
   {
      scanf("%d", &arr[i]);
   }

   printf("Reversed numbers:\n");
   for(i = n - 1; i >= 0; i--)
   {
```

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

Output



5. Write a program to search for an element in an array (linear search).

IPO

- Input:

  Get a value as input say n

- Process:

  Use a loop to compare each element of the array with the search element

  If a match is found, print the position

  If not found, print"Element not found"

- Output:

  Print the position if the element is found

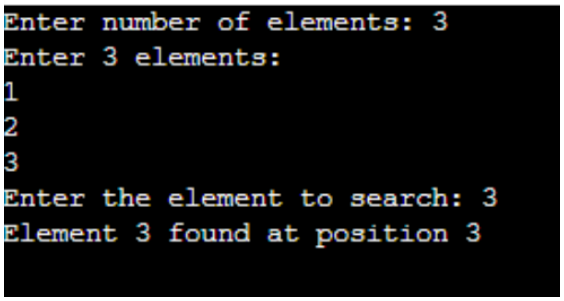
  Print "Element not found" if it doesn’t found in the array

  Program

```
#include <stdio.h>

void main()

{

   int a[100], n, i, b, found = 0;

   printf("Enter number of elements: ");
```

```
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for(i = 0; i < n; i++)
    {
        scanf("%d", &a[i]);
    }
    printf("Enter the element to search: ");
    scanf("%d", &b);
    for(i = 0; i < n; i++)
    {
        if(a[i] == b)
        {
            found = 1;break;
        }
    }


    if(found)
        printf("Element %d found at position %d\n", b, i + 1);
    else
        printf("Element not found in the array.\n");
}
```

Output



6. Write a program to sort an array in ascending order.

IPO

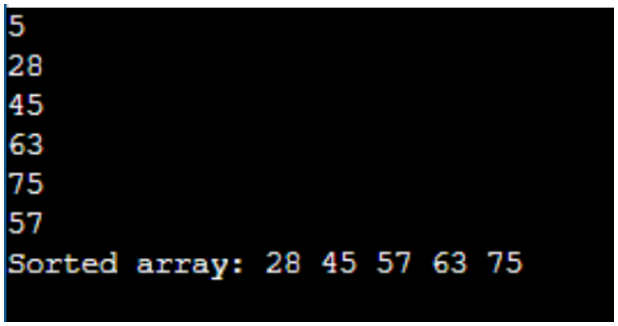
- **Input:**
  Get a value n (number of elements)
  Get n elements into an array
- **Process:**
  Use nested loops to compare and swap elements if needed
  Sort array in ascending order
- **Output:**
  Print the sorted array in ascending order

Program

```
#include<stdio.h>
void main()
{
   int a[100], n, i, j, temp;
   scanf("%d", &n);
   for(i = 0; i < n; i++)
   scanf("%d", &a[i]);
   for(i = 0; i < n - 1; i++)
   {
      for(j = i + 1; j < n; j++)
      {
         if(a[i] > a[j])
         {
```

```
            temp = a[i];
            a[i] = a[j];
            a[j] = temp;
        }
    }
  }
  printf("Sorted array: ");
  for(i = 0; i < n; i++)
  printf("%d ", a[i]);
}
```

Output



7. Write a program to insert an element in an array.

IPO

- **Input:**

  Get number of elements n

  Get array elements

  Get element to insert and position to insert

- **Process:**

  Shift elements to the right from the given position

  Insert new element

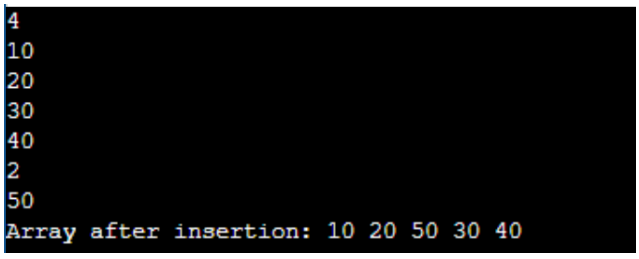
- **Output:**

  Print array after insertion

Program

```
#include <stdio.h>
void main()
{
   int a[100], n, i, x, y;
   scanf("%d", &n);
   for(i = 0; i < n; i++)
   scanf("%d", &a[i]);
   scanf("%d", &y);
   scanf("%d", &x);
   for(i = n; i > y; i--)
      a[i] = a[i - 1];
      a[y] = x;
   n++;
   printf("Array after insertion: ");
   for(i = 0; i < n; i++)
   printf("%d ", a[i]);
}
```

Output



8. Write a program to delete an element from an array.

IPO

- Input:

  Get number of elements n

  Get array elements

  Get the position to delete

- Process:

  Shift all elements to the left from that position

- Output:

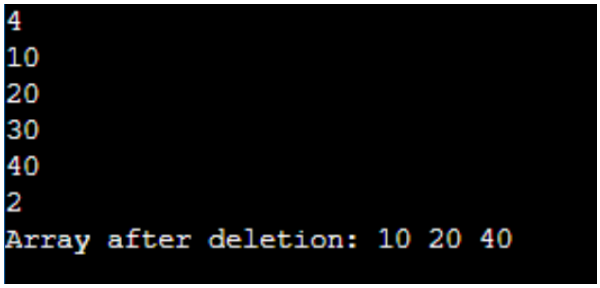
  Print array after deletion

  Program

```
#include<stdio.h>
void main()
{
   int a[100],n,i,p;

   scanf("%d",&n);
   for(i=0;i<n;i++)
   scanf("%d",&a[i]);
   scanf("%d",&p);
   for(i=p;i<n-1;i++)
      a[i]=a[i+1];
      n--;
   printf("Array after deletion: ");
   for(i=0;i<n;i++)
```

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

Output



9. Write a program to find the frequency of elements in an array.

- Input:

  Get number of elements n

  Get array elements

- Process:

  Use a loop to count how many times each unique element appears

  Mark counted elements

- Output:

  Print frequency of each unique element

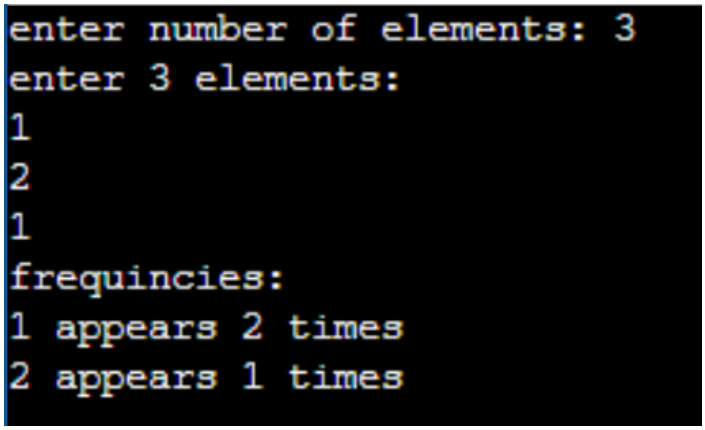
Program

```
#include<stdio.h>
void main()
{
    int a[100], n, i, j, count;
    printf("enter number of elements: ");
    scanf("%d", &n);
```

```
printf("enter %d elements:\n", n);
for(i = 0; i < n; i++)
{
   scanf("%d", &a[i]);
}
printf("frequincies:\n");
for(i = 0; i < n; i++)
{
   count = 0;
   for(j = 0; j < i; j++)
   {
      if(a[i] == a[j])
      {
         count = 1;
         break;
      }
   }
   if(count == 0)
   {
      count = 1;
      for(j = i + 1; j < n; j++)
      {
         if(a[i] == a[j])
            count++;
```

```
            }
            printf("%d appears %d times\n", a[i], count);
        }
    }
}
```

Output



10. Write a program to merge two arrays.

IPO

- Input:

  Get two values n1 and n2

  Get n1 elements in first array

  Get n2 elements in second array

- Process:

  Copy elements of both arrays into a third array

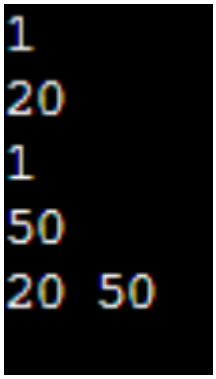
- Output:

  Print the merged array

  Program

```
#include <stdio.h>
void main()
{
   int a1[50], a2[50], me[100];
   int n1, n2, i, j;

   scanf("%d", &n1);
   for(i = 0; i < n1; i++)
      scanf("%d", &a1[i]);

   scanf("%d", &n2);
   for(i = 0; i < n2; i++)
      scanf("%d", &a2[i]);

   for(i = 0; i < n1; i++)
      m[i] = a1[i];

   for(j = 0; j < n2; j++)
      m[i + j] = a2[j];

   for(i = 0; i < n1 + n2; i++)
      printf("%d ", m[i]);
}
```

  Output


1
20
1
50
20 50