Emre Özkaya
B1905.010057
Section:1

```
#include <stdio.h>

int main() {

  int number;
  int limitOfNumber;
  int numberbox[100];

  printf("*** An Array That Shows The Numbers Entered in Reverse Order ***\n\n" );

  printf("Please specify how many numbers you want to enter : ");
  scanf("%d",&limitOfNumber);

  printf("\nPlease Enter Numbers \n");

  for(number=0;number<limitOfNumber;number++)
  {
    scanf("%d",&numberbox[number]);
  }

  printf("Reverse of numbers entered : ");

  while(number>0)
  {
    printf("%d  ",numberbox[number-1]);
    number -- ;
  }
  return 0;
}
```

```
#include <stdio.h>

int main()
 {
    int elementReferance;
    int element;
    int box[100];
    int box1[100];

    printf("*** Program For Copying Elements Of One Array To Another ***\n\n");
    printf("You can enter size of elements's array. : " );
    scanf("%d",&elementReferance);

    printf("\nPlease Enter Elements \n");

    for(element=0;element<elementReferance;element++)
    {
      scanf("%d",&box[element]);
      box1[element]=box[element];
    }

    printf("\nElements of First Array : ");

    for(element=0;element<elementReferance;element++)
    {
      printf("%d  ",box[element]);
    }

    printf("\n\n");
    printf("Elements of Second Array : ");

    for(element=0;element<elementReferance;element++)
    {
      printf("%d  ",box1[element]);
    }
    printf("\n");

  return 0;
}
```

```c
#include <stdio.h>

int duplicate(int sizeOfArray)
{
  int counter = 0 , nums , finder ;
  int box[sizeOfArray];

  printf("\nPlease Enter Numbers \n");

  for(nums = 0 ; nums < sizeOfArray ; nums++)
  {
    scanf("%d",&box[nums]);
  }

  for( finder = 0 ; finder < sizeOfArray ; finder++)
  {

      for (nums = finder + 1 ; nums < sizeOfArray ; nums++)
      {
           if (box[finder]==box[nums])
             {

                 counter++;
                   break;
             }

     }
 }
 return counter;
}

int main()
{
  int size , duplicateCounter;

  printf("*** Program For Counting A Total Number Of Duplicate Elements In An Array ***\n\n");

  printf("Please Enter A Size Of Array : ");
  scanf("%d",&size);

  duplicateCounter = duplicate(size);
  printf("A Total Number Of Duplicate Elements In An Array : %d",duplicateCounter);

  return 0;
}
```

```
#include <stdio.h>
#include <string.h>
int main()
 {
  char box[100];
  int elements=0;
  int length;

  printf("*** Unique Element Finder ***\n\n");

  printf("Please Enter Your Sentence : ");
  fgets(box,sizeof(box),stdin);
  length = strlen(box);
  printf("These Are All Unique Elements :");
  for(elements = 0 ; elements<length ; elements++)
  {
        if((box[elements]>='A' &&  box[elements]<='Z') || (box[elements]>='a' &&  box[elements]<='z')
|| (box[elements]>='0' && box[elements]<='9') )
        {
          continue;
        }

        else{
            printf("%c",box[elements]);
        }
  }

  return 0;
}
```

```c
#include <stdio.h>

int main() {
  int sizeOfArray;
  int arrayNums,numOfSum,arrayNums1,order,referance;
  int array[100];
  int array1[100];
  int sumOfArray[200];
  printf("*** Merge Two Arrays Of Same Size Sorted In Decending Order ***\n\n" );
  printf("Please Enter Size of Arrays : ");
  scanf("%d",&sizeOfArray);

  printf("\nPlease Enter Numbers of First Array : \n");

  for(arrayNums = 0 ; arrayNums < sizeOfArray ; arrayNums++)
  {
    scanf("%d",&array[arrayNums]);
  }
  printf("\nPlease Enter Numbers of Second Array : \n");

  for(arrayNums = 0 ; arrayNums < sizeOfArray ; arrayNums++)
  {
    scanf("%d",&array1[arrayNums]);
  }

  for (numOfSum = 0 ; numOfSum < sizeOfArray ; numOfSum ++ )
  {
    sumOfArray[numOfSum] = array[numOfSum] ;

  }
  for(arrayNums1 = 0 ; arrayNums1 < sizeOfArray ; arrayNums1++)
  {
    sumOfArray[numOfSum] = array1[arrayNums1] ;
    numOfSum++;
  }

  for (numOfSum = 0 ; numOfSum < (2*sizeOfArray) ; numOfSum++)
  {
    for(order = 0 ; order < (2*sizeOfArray)-1 ; order++)
    {
      if (sumOfArray[order]<=sumOfArray[order+1])
       {
        referance = sumOfArray[order+1];
        sumOfArray[order+1] = sumOfArray[order] ;
        sumOfArray[order] = referance ;
      }
    }
  }
  printf("\nOrder Of Two Arrays :");
  for(order = 0 ; order < 2*sizeOfArray ; order ++)
  {
    printf("%d  ",sumOfArray[order]);
  }
  printf("\n\n");

  return 0;
}
```

```
#include <stdio.h>

void frequency(int sizeOfArray)
{
  int counter, nums , numsOfBox ,placeHolder=-1 ;
  int box[sizeOfArray],frqncyBox[sizeOfArray];


  printf("\nPlease Enter Numbers \n");

  for(nums = 0 ; nums < sizeOfArray ; nums++)
  {
    scanf("%d",&box[nums]);
  }

  for( numsOfBox = 0 ; numsOfBox < sizeOfArray ; numsOfBox++)
  {
     counter = 1;
     for (nums = numsOfBox + 1 ; nums < sizeOfArray ; nums++)
     {
          if (box[numsOfBox]==box[nums])
            {

               counter++;
               frqncyBox[nums] = placeHolder;

            }

     }
     if(frqncyBox[numsOfBox] != placeHolder)
     {
       frqncyBox[numsOfBox] = counter ;
     }
 }
  printf("Count the frequency of each element of an array : \n");
  for(int i = 0; i < sizeOfArray; i++){
     if(frqncyBox[i] != placeHolder){
        printf(" %d", box[i]);
        printf(" --> ");
        printf("%d\n", frqncyBox[i]);
     }
 }

}

int main()
{
  int size , frequencyCounter;
  printf("*** Program in C to count the frequency of each element of an array ***\n\n");
  printf("Please Enter A Size Of Array : ");
  scanf("%d",&size);

  frequency(size);


  return 0;
}
```

```
#include <stdio.h>
void finder(int size);

int main() {
  int refSize;
  printf("*** Finder The Maximum And Minimum Element In An Array ***\n\n");
  printf("Please Enter Size of Array : ");
  scanf("%d",&refSize);
  printf("\n");

  finder(refSize);

  return 0;
}

void finder(int size)
{
  int i ;
  int maxMin[size];
  for (i = 0 ; i<size ;i++) {
    printf("%d.--> ",i+1);
    scanf("%d",&maxMin[i]);
  }

  int max = maxMin[0];
  int min = maxMin[1];
  int j ;
  for ( j = 0 ; j<size ;j++ ) {
    if (maxMin[j]>max) {
      max = maxMin[j] ;
    }
  }
  for ( j = 0 ; j<size ;j++ ) {
    if (maxMin[j]<min) {
      min = maxMin[j] ;
    }
  }
    printf("\nMax element of array : %d\n",max);
    printf("\nMin element of array : %d\n",min);
}
```

```
#include <stdio.h>
int main()
{
  int size_of_array,i,size_of_odd=0,size_of_even=0;
  int wholeArray[100];
  int oddArray[100];
  int evenArray[100];
  printf("*** Program in C To Separate Odd and Even Integers In Separate Arrays ***\n\n");
  printf("Please Enter Your Array's Size : \n");
  scanf("%d",&size_of_array);

  printf("Please Enter Enter Values : \n");
  for(i=0 ; i<size_of_array ; i++)
  {
    scanf("%d",&wholeArray[i]);
  }
  for(i=0 ; i<size_of_array ; i++)
  {
    if(wholeArray[i] % 2 == 0)
    {
      evenArray[size_of_even]=wholeArray[i];
      size_of_even++;
    }
    else
    {
      oddArray[size_of_odd] = wholeArray[i];
      size_of_odd++;
    }
  }

  printf("Even Numbers Are : \n");
  for (i=0 ; i<size_of_even ; i++) {
    printf("%d  ",evenArray[i]);
  }
  printf("\nOdd Numbers Are : \n");
  for (i=0 ; i<size_of_odd ; i++) {
    printf("%d  ",oddArray[i]);
  }
  return 0 ;
}
```

```c
#include <stdio.h>
int main() {
  int size,i,delete_position;
  int number[100];
  printf("*** Program In C To Delete An Element At Desired Position From An Array ***\n\n");
  printf("Please Enter Size Of Array : ");
  scanf("%d",&size);

  printf("Please Enter Values : \n");
  for(i = 0 ; i < size ; i++)
  {
    scanf("%d",&number[i]);
  }
  printf("Your Array is : \n");
  for(i = 0 ; i < size ; i++)
  {
    printf("%d ",number[i]);
  }
  printf("\nPlease Enter Position of which you want to delete :  \n" );
  scanf("%d",&delete_position);

  for (i=delete_position-1 ; i < size-1 ;delete_position++) {
    number[i++]=number[i];
  }
  printf("New order of array : \n");
  for(i = 0 ; i < size-1 ; i++)
  {
    printf("%d ",number[i]);
  }
  return 0;
}
```

```
#include <stdio.h>

int main() {
  int size, i,secondMin,min;
  int second_min[100];
  printf("*** Second Smallest Element Finder In An Array ***\n\n");
  printf("Please Enter Size Of Array : ");
  scanf("%d",&size);

  printf("Please Enter Elements : \n");
  for ( i = 0 ; i < size ; i++) {
    scanf("%d",&second_min[i]);
  }

  if (second_min[0] < second_min[1]) {
        min = second_min[0];
        secondMin = second_min[1];
    }
    else {
      min = second_min[1];
      secondMin = second_min[0];
    }
    for (i = 2; i < size; i++) {
        if (second_min[i] < min) {
        secondMin = min;
        min = second_min[i];
        }
        else if (second_min[i] < secondMin) {
            secondMin = second_min[i];
        }
    }

  printf("Second Smallest Element Is : %d",secondMin );

  return 0;
}
```

```
#include <stdio.h>
#include <stdlib.h>
 int r;
 int c;

void sumOfMatrices(int matrice1[10][10] , int matrice2[10][10]){

   int i , j ;
  int sumMatrice[10][10];
  printf("\n*** Sum of Matrices ***\n\n");
  for (i = 0 ; i < r ; i++) {
    for(j = 0 ; j < c ; j ++){
    printf(" %d ",sumMatrice[i][j] = (matrice1[i][j] + matrice2[i][j]));
  }
  printf("\n");
}
}

void representOfMatrices(int matrice1[10][10] , int matrice2[10][10])
{
  int i,j;
  system("cls");
  printf("*** First Matrice ***\n\n");
  for (i = 0 ; i < r ; i++) {
    for(j = 0 ; j < c ; j ++){
    printf(" %d ",matrice1[i][j]);
  }

  printf("\n" );
  }

  printf("\n*** Second Matrice ***\n\n");
  for (i = 0 ; i < r ; i++) {
    for(j = 0 ; j < c ; j ++){
    printf(" %d ",matrice2[i][j]);
  }
  printf("\n" );
  }

}

void getArray()
{
  int i , j ;
  int matrice1[10][10];
  int matrice2[10][10];
  printf("Please Enter First Matrice's Elements : \n");
  for (i = 0 ; i < r ; i++) {
    for(j = 0 ; j < c ; j ++){
    scanf("%d",&matrice1[i][j]);
  }
  }

  system("cls");
  printf("Please Enter Second Matrice's Elements : \n");
  for (i = 0 ; i < r ; i++) {
    for(j = 0 ; j < c ; j ++){
    scanf("%d",&matrice2[i][j]);
  }
  }
```

```
  representOfMatrices(matrice1, matrice2);
  sumOfMatrices(matrice1 , matrice2);
}

int main()
 {
  int i,j ;
  printf("*** program in C for addition of two Matrices of same size ***\n");
  printf("\nPlease Enter size of matrixs's' :");
  scanf("%d %d",&r,&c);
  printf("              *** All Matrices' Size Are %d | %d ***\n\n",r,c);

  getArray();
  return 0;
}
```

```c
#include <stdio.h>

void get_matrice(int r , int c){
  int i , j ;
  int matrice[r][c];
  int transpose[r][c];
  printf("\nPlease Enter Values\n\n");
  for(i = 0 ; i < r ; i ++)
  {
    for (j = 0 ; j < c ; j ++)
     {
      printf("Please Enter %d Raw %d Element : ",i+1,j+1 );
      scanf("%d",&matrice[i][j]);
    }
  }
  printf("\n*** The Matrice Entered Is Below ***\n\n");
  for(i = 0 ; i < r ; i ++)
  {
    for (j = 0 ; j < c ; j ++)
     {
      printf(" %d ",matrice[i][j]);
    }
    printf("\n");
  }
  for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j) {
            transpose[j][i] = matrice[i][j];
        }

    printf("\n*** Transpose Of The Matrice ***\n\n");
    for (i = 0; i < c; ++i)
        for (j = 0; j < r; ++j) {
            printf("%d  ", transpose[i][j]);
            if (j == r - 1)
                printf("\n");
        }
}

int main()
{
  int raw,column;
  printf("*** Program In C To Find Transpose Of A Given Matrice ***\n\n");
  printf("Please Enter Rows As A Number : ");
  scanf("%d",&raw);
  printf("Please Enter Columns As A Number : ");
  scanf("%d",&column);

  get_matrice(raw , column);

  return 0;
}
```

```
#include <stdio.h>
void right_diagonals(int raw , int column)
{ int i , j , sum = 0 ;

  int matrix[raw][column];

  printf("\nPlease Enter Values\n\n");
  for(i = 0 ; i < raw ; i ++)
  {
    for (j = 0 ; j < column ; j ++)
     {
      printf("Please Enter %d Raw %d Element : ",i+1,j+1 );
      scanf("%d",&matrix[i][j]);
    }
  }

  printf("\n*** The Matrice Entered Is Below ***\n\n");
  for(i = 0 ; i < raw ; i ++)
  {
    for (j = 0 ; j < column ; j ++)
     {
      printf(" %d ",matrix[i][j]);
      if((i==j))
      {
        sum = sum  + matrix[i][j];
          }

    }
    printf("\n");
  }

   printf("\nSum of Right Diagonal Of A Matrix : %d  " , sum);

}
int main() {

        int r,c;
  printf("*** Program In C To Find Sum Of Right Diagonals Of A Matrix ***\n\n");
  printf("Please Enter Rows As A Number : ");
  scanf("%d",&r);
  printf("Please Enter Columns As A Number : ");
  scanf("%d",&c);

  right_diagonals(r,c);


        return 0;
}
```

```
#include <stdio.h>
void sumFunct( int raw , int column)
{
  int i , j , sum = 0 , allSum = 0;
  int matrix[raw][column];

  printf("\nPlease Enter Values  \n\n");

  for (i = 0 ; i < raw ; i++)
   {
                    for(j = 0 ; j < column ; j++)
                    {
                          printf("Please Enter %d Raw %d Element : ",i+1,j+1 );
                          scanf("%d",&matrix[i][j]);
                          allSum = allSum + matrix[i][j];
                    }

  }

  printf("\nYour Matrix Is  \n\n");

  for (i = 0 ; i < raw ; i++)
   {
                    for(j = 0 ; j < column ; j++)
                    {
                          printf("%d  ",matrix[i][j]);
                    }
       printf("\n");
  }

  printf("\n");

    for (i = 0 ; i < raw ; i++)
   {
                    for(j = 0 ; j < column ; j++)
                    {
                          sum = sum + matrix[i][j] ;
                    }
              printf("Sum of %d. raw  : %d ",i+1, sum );
              sum = 0 ;
              printf("\n");
  }

  printf("\n");

   for (j = 0 ; j < column ; j++)
   {
                    for(i = 0 ; i < raw ; i++)
                    {
                          sum = sum + matrix[i][j] ;
                    }
              printf("Sum of %d. column  : %d ",j+1, sum );
              sum = 0 ;
          printf("\n");
}
       printf("\n");
       printf("Sum of all numbers entered : %d ", allSum);
}
```

```
int main() {
int raws,columns ;
  printf("*** Program In C To Find Sum of Rows An Columns Of A Matrix.  ***\n\n");

  printf("Please Enter Raw Of Array : ");
  scanf("%d",&raws);

  printf("Please Enter Column Of Array : ");
  scanf("%d",&columns);

  sumFunct(raws,columns);


      return 0;
}
```

```
#include <stdio.h>
int determinant(int mtrx[3][3]);

int main() {
	int i , j ;
	int matrix[3][3] ;

	printf("*** Program In C To Calculate Determinant Of A 3 x 3 Matrix ***\n\n");

	printf("\nPlease Enter Values  \n\n");

	for (i = 0 ; i < 3 ; i++)
	{
		for(j = 0 ; j < 3 ; j++)
		{
			printf("Please Enter %d Raw %d Element : ",i+1,j+1 );
			scanf("%d",&matrix[i][j]);
		}
	}

	printf("\nDeterminant of A 3 x 3 Matrix : %d\n",determinant(matrix));
	return 0;
}

int determinant(int mtrx[3][3]){
	int i , j ;
	int consultant=0,consultant1=0,consultant2=0,consultant3=0 , cons1=0 , cons2=0;

		cons1 = mtrx[i+1][1] * mtrx[i+2][2];
		cons2 = mtrx[i+1][2] * mtrx[i+2][1];
		consultant1 = mtrx[0][0] * (cons1 - cons2 );

		cons1 = mtrx[i+1][0] * mtrx[i+2][2];
		cons2 = mtrx[i+1][2] * mtrx[i+2][0];
		consultant2 = mtrx[0][1] * (cons1 - cons2) ;

		cons1 = mtrx[i+1][0] * mtrx[i+2][1];
		cons2 = mtrx[i+1][1] * mtrx[i+2][0];
		consultant3 = mtrx[0][2] * (cons1 - cons2) ;

	consultant = consultant1 - consultant2 + consultant3 ;
	return consultant ;
}
```

```
#include <stdio.h>
void equal(int row , int column , int row1 , int column1){
	int matrix[row][column];
	int matrix1[row1][column1];
	int i , j , checker=0;
	/*CALLING*/
	printf("\nPlease Enter Values of First Matrix \n\n");
		 for(i = 0 ; i < row ; i ++)
		{
			for (j = 0 ; j < column ; j ++)
		{
			 printf("Please Enter %d row %d Element : ",i+1,j+1 );
			 scanf("%d",&matrix[i][j]);
		}
		 }

		printf("\nPlease Enter Values Of Second Matrix \n\n");
		 for(i = 0 ; i < row1 ; i ++)
		{
			for (j = 0 ; j < column1 ; j ++)
		{
			 printf("Please Enter %d row %d Element : ",i+1,j+1 );
			 scanf("%d",&matrix1[i][j]);
		}
		 }
	/* DISPLAYING */
  printf("\n*** The First Matrix Entered Is Below ***\n\n");
		 for(i = 0 ; i < row ; i ++)
		 {
			for (j = 0 ; j < column ; j ++)
		{
			printf(" %d ",matrix[i][j]);
		}
	printf("\n");
  }
    printf("\n*** The Second Matrix Entered Is Below ***\n\n");
		 for(i = 0 ; i < row1 ; i ++)
		 {
			for (j = 0 ; j < column1 ; j ++)
		{
			printf(" %d ",matrix1[i][j]);
		}
	printf("\n");
  }
  /*WHETHER*/
  if(row!=row1 || column!=column1)
  {
	printf("\n !!! These Are not Equal , Because their rows or columns different !!!");
  }
  else
  {
		for(i = 0; i < row; i++)
		{
			for(j = 0 ; j < column ; i++)
		{
			if(matrix[row][column] != matrix1[row][column])
				{
					checker = -1;
					break;
				}
```

```
                }
            }
            if(checker==0)
            {
                    printf("\n! These Are Equal ! ");
             }
             else
             {
                 printf("\n! These Are not Equal !");
             }
  }
}

int main(int argc, char** argv) {
        int r , c;
        int r1 , c1;


        printf("*** Program In C To Accept Two Matrices And Check Whether They Are Equal ***\n\n");

        printf("Please Enter Rows And Column Of Your First Matrix : \n");
        scanf("%d %d",&r,&c);

        printf("Please Enter Rows And Column Of Your Second Matrix : \n");
        scanf("%d %d",&r1,&c1);

        equal(r , c , r1 , c1);

        return 0;
}
```

```
#include <stdio.h>

void identity(int row , int column)
{
	int i,j,checker=0;
	int matrix[row][column];
	/*GET*/
	printf("\nPlease Enter Values of Matrix as a %d X %d \n\n",row,column);
		for(i = 0 ; i < row ; i ++)
		{
			for (j = 0 ; j < column ; j ++)
		{
			 printf("Please Enter %d row %d Element : ",i+1,j+1 );
			 scanf("%d",&matrix[i][j]);
		}
		 }
	/*DISPLAY*/
	printf("\n*** The Matrix Entered Is Below %d X %d ***\n\n",row,column);
		 for(i = 0 ; i < row ; i ++)
		 {
			for (j = 0 ; j < column ; j ++)
		{
			printf(" %d ",matrix[i][j]);
		}
	printf("\n");
  }
  /*cHECKER*/
	for(i = 0 ; i < row ; i ++)
	{
			for (j = 0 ; j < column ; j ++)
		{
			if(i==j && matrix[i][j]!=1)
				{
				checker=-1;
				break;
				}
		}
			if(i != j && matrix[i][j] != 0)
			{
				checker=-1;
				break;
			}

	}
	if(checker==0)
	  {
		printf("Given matrix is an identity matrix");
	  }
    else
		{
			printf("Given matrix is not an identity matrix");
		}
}
int main() {
	int r , c;

	printf("*** Program in C to check whether a given matrix is an identity matrix ***\n\n");

	printf("Please Enter Rows And Column Of Your Matrix : \n");
	scanf("%d %d",&r,&c);
```

```
	if(r!=c)
	{
		printf("Error !!! You Should Enter Same Column And Row !!!");
	}
	else
	{
		identity(r,c);
	}

return 0;
}
```

```
#include <stdio.h>
int main() {
	int size , i , counter=-1;
	int maxCounter = 0;
	int index = 0 ;
	int majorityArray[100];

	printf("***Program In C To Find Majority Element Of An Array ***\n\n");

	printf("Please Enter Size Of Your Array : ");
	scanf("%d",&size);

	printf("\nPlease Enter Elements Of Your Array\n\n");
	for(i=0 ; i < size ; i++)
	{
		printf("Please Enter %d. Element : ",i+1);
		scanf("%d",&majorityArray[i]);
	}

   for (int i = 0; i < size; i++)
	{
	int counter = 0;
		for (int j = 0; j < size ; j++)
			{
		if (majorityArray[i] == majorityArray[j])
		{
			counter++;
				}
	   }
	if (counter > maxCounter)
		{
	   maxCounter = counter;
	   index = i;
	}
   }

   if (maxCounter > size / 2)
   {
	printf("\nMajority Element is : %d",majorityArray[index]);
	}
   else
	{
	printf("There isn't Majority Element");
	}

	return 0;
}
```

```
#include <stdio.h>

int subset(int arr[] , int arr_size , int arr1[] , int arr1_size)
{
	int i , j ;
	for (i = 0; i < arr1_size; i++)
	{
	for (j = 0; j < arr_size; j++)
		{
		if(arr1[i] == arr[j])
			{
			break;
			}
	}
   if(j == arr_size)
	  return 0;
	}

   return 1;
}

int subset1(int arr[] , int arr_size , int arr1[] , int arr1_size)
{
	int i , j ;

	for (i = 0; i < arr_size; i++)
	{
	 for (j = 0; j < arr1_size; j++)
		{
		if(arr[i] == arr1[j])
			{
			break;
			}
	   }
   if(j == arr1_size)
	{
	return 0;
	}

	}
return 1;
}

int main()
{
	int size , size1 , i , j ,consultant;
	int array[100];
	int array1[100];
	printf("*** Program in C to check whether an array is subset of another array ***\n\n");

	printf("Please Enter Size Of First Array : ");
	scanf("%d",&size);

	printf("Please Enter Size Of Second Array : ");
	scanf("%d",&size1);

	printf("Please Enter Values of First Array : \n");
	for(i = 0 ; i < size ; i ++)
	{
		scanf("%d",&array[i]);
```

```
	}

	printf("Please Enter Values of Second Array : \n");
	for(i = 0 ; i < size1 ; i ++)
	{
		scanf("%d",&array1[i]);
	}
	if(size1<size)
	{
		consultant = subset(array,size,array1,size1);
			if(consultant==1)
			{
				printf("The second array is the subset of first array.");
			}
			else
			{
				printf("There is not subset of in these arrays.");
			}
	}
	else
	{
		consultant = subset1(array,size,array1,size1);
			if(consultant==1)
			{
				printf("The first array is the subset of second array.");
			}
			else
			{
				printf("There is not subset of in these arrays.");
			}
	}

	return 0;
}
```

```
#include <stdio.h>
void combintn(int arr1[], int hold[], int st, int end, int index, int r)
{
      int i , j ;
   if (index != r)
   {
       for (i=st; i<=end && end-i+1 >= r-index; i++)
   {
      hold[index] = arr1[i];
      combintn(arr1, hold, i+1, end, index+1, r);
   }

   }
   else
      {
            for (j=0; j<r; j++)

            printf("%d ", hold[j]);
            printf("\n");
            return ;
      }
}
int main()
{
      int size , r , i , index = 0 , start = 0 , finish = 0;
      int array[100];
      printf("*** program to print all possible combinations of r elements in a given array ***\n\n");
      printf("Please Enter Size Of Array : ");
      scanf("%d",&size);

      printf("\nPlease Enter Values Of Array \n\n");
      for(int i = 0 ; i < size ; i ++)
      {
            printf("%d . Element : ",i+1);
            scanf("%d",&array[i]);
      }
      printf("\nPlease Enter Value Of R : ");
   scanf("%d",&r);

   int holder[r];

      printf("\nThe Given Array Is Below \n");
      printf("===>");
      for(i = 0; i < size ; i++)
            {
                  printf("%d ", array[i]);
            }
      printf("\n==============================\n");

      printf("The combinations are \n\n");

      combintn(array, holder, 0, size-1, 0, r);

}
```