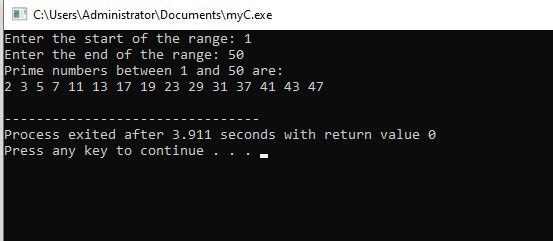
**PF Lab 08**

**Name: Ali Rooman**

**Roll No: 24K-0792**

Q1. Write a C program that generates a sequence of prime numbers within a given range using nested loops.

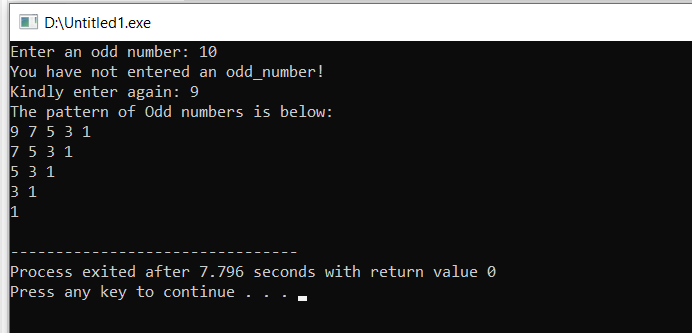
|  |
| --- |
| #include <stdio.h>  int main() {  int start, end, i, j, isPrime;  printf("Enter the start of the range: ");  scanf("%d", &start);  printf("Enter the end of the range: ");  scanf("%d", &end);  printf("Prime numbers between %d and %d are:\n", start, end);  for (i = start; i <= end; i++) {  if (i < 2)  continue;  isPrime = 1;  for (j = 2; j \* j <= i; j++) {  if (i % j == 0) {  isPrime = 0;  break;  }  }    if (isPrime) {  printf("%d ", i);  }  }  printf("\n");  return 0;  } |

**OUTPUT**

Q2. Generate a pattern of odd numbers in decreasing order starting from a user-specified number using nested loops.

|  |
| --- |
| #include <stdio.h>  int main(){  int odd\_num,i,j;  printf("Enter an odd number: ");  scanf("%d",&odd\_num);  while(odd\_num%2==0){  printf("You have not entered an odd\_number! \nKindly enter again: ");  scanf("%d",&odd\_num);  }  printf("The pattern of Odd numbers is below: \n");  for(i=odd\_num;i>0;i=i-2){  for(j=i;j>0;j=j-2){  printf("%d ",j);  }  printf("\n");  }  return 0;  } |

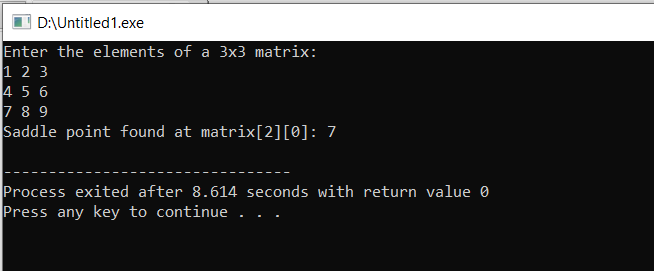
**OUTPUT**

****

Q3. Write a C program to find the saddle point(s) in a given 3x3 matrix. A saddle point is an element that is the smallest in its row and the largest in its column.

|  |
| --- |
| #include <stdio.h>  int main() {      int matrix[3][3],i,j,k;      printf("Enter the elements of a 3x3 matrix:\n");      for(i = 0; i < 3; i++) {          for(j = 0; j < 3; j++) {              scanf("%d", &matrix[i][j]);          }      }      int saddlePointFound = 0;      for(i = 0; i < 3; i++) {          int min = matrix[i][0];          int minCol = 0;          for(j = 1; j < 3; j++) {              if(matrix[i][j] < min) {                  min = matrix[i][j];                  minCol = j;              }          }          int isSaddlePoint = 1;          for(k = 0; k < 3; k++) {              if(matrix[k][minCol] > min) {                  isSaddlePoint = 0;                  break;              }          }          if(isSaddlePoint) {              printf("Saddle point found at matrix[%d][%d]: %d\n", i, minCol, min);              saddlePointFound = 1;          }      }      if(!saddlePointFound) {          printf("No saddle point found in the matrix.\n");      }      return 0;  } |

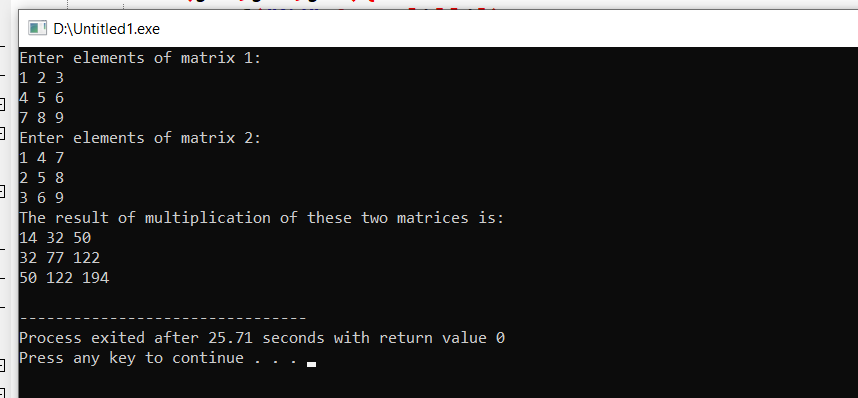
**OUTPUT**

****

Q4. Write a C program to multiply two matrices of size 3x3 and display the result matrix.

|  |
| --- |
| #include<stdio.h>  int main(){  int mat1[3][3];  int mat2[3][3];  int result[3][3];  int i,j,k;  printf("Enter elements of matrix 1: \n");  for (i=0;i<3;i++){  for (j=0;j<3;j++){  scanf("%d",&mat1[i][j]);  }  }  printf("Enter elements of matrix 2: \n");  for (i=0;i<3;i++){  for (j=0;j<3;j++){  scanf("%d",&mat2[i][j]);  }  }  for(i=0;i<3;i++){  for(j=0;j<3;j++){  result[i][j] = 0;  for(k=0;k<3;k++){  result[i][j] = result[i][j] + mat1[i][k] \* mat2[k][j];  }  }  }  printf("The result of multiplication of these two matrices is: \n");  for(i=0;i<3;i++){  for(j=0;j<3;j++){  printf("%d ",result[i][j]);  }  printf("\n");  }  } |

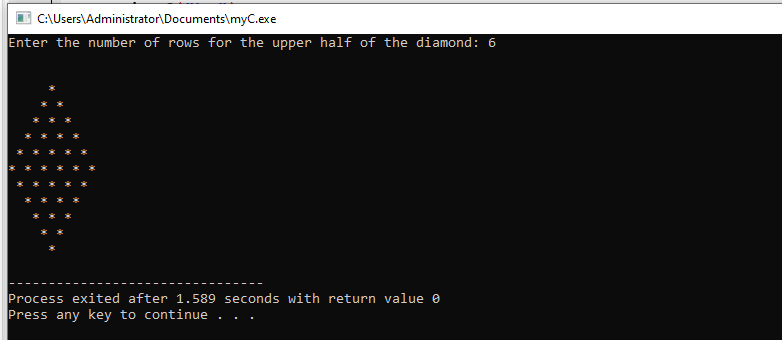
**OUTPUT**

****

Q5. Write a C program to generate a diamond shape pattern using nested loops. The program should take the number of rows for the upper half of the diamond as input from the user.

|  |
| --- |
| #include <stdio.h>  int main() {  int rows;  printf("Enter the number of rows for the upper half of the diamond: ");  scanf("%d", &rows);  printf("\n");  printf("\n");  for (int i = 0; i < rows; i++) {  for (int j = 0; j < rows - i - 1; j++) {  printf(" ");  }  for (int k = 0; k <= i; k++) {  printf("\* ");  }  printf("\n");  }  for (int i = rows - 2; i >= 0; i--) {  for (int j = 0; j < rows - i - 1; j++) {  printf(" ");  }  for (int k = 0; k <= i; k++) {  printf("\* ");  }  printf("\n");  }  return 0;  } |

**OUTPUT**

****