```
/////////////////////
//
//  www.maknoon.com
//
/////////////////////

//-----------------------------------------
// The Header files are:
//-----------------------------------------

#include<iostream.h>
#include<stdio.h>
#include<math.h>
#include<ctype.h>
#include<string.h>
#include<stdlib.h>
#include<iomanip.h>

/*--------------------------------------------------------*/

/****************************/
/*Calculating the Magnitude */
/****************************/

double Magnitude(double a, double b)     // Function to calculate the magnitude

   {
      double M;                          // Magnitude's variable

      M=sqrt(pow(a,2)+pow(b,2));         // Equation for calculating the magnitude

      if(M>255)                          // Check if the magnitude is greater than 255
      {

        // In case magnitude >255, the program will make it 255

       M=255;

      }
      return M;                          //It will return the value of the magnitude
   }

/*----------------------------(Done BY Omar)------------------------------------*/

/**************************/
/* Calculating the Angle  */
/**************************/

double Angle(double c, double d)       // Function to calculate the angle

  {
    double Theta ;                     // Angle's variable

    Theta=atan2(c,d);                  // Equation to calculate the angle

   Theta=255*(M_PI+Theta)/(2*M_PI);    // Convert Theta into integer between (0-255)

   return Theta;                       // It will return the value of the angle
  }

/*----------------------------(Done BY Omar)------------------------------------*/


//-----------------------------------------
// The Main Program                                    -
//-----------------------------------------


int main(int argc,char *argv[])

 {

/********************/
/* Reading Variables */
/********************/
   FILE *fp,*fp2;              //files decleration
   char ignore[45],*xv;
  int array[3],Rows,Cols;
  int  i,j;                   // for loop variables
  double dx,dy;               // Horizontal & Vertical edge's variables


/********************/
/* Reading the file */
/********************/

 if (argc !=2)                      //you have to support the file name and the program name

    {

      cout <<" Please the command should be like this (ProgramName ImageName)"<<endl;

      exit(0);
    }

      fp=fopen(argv[1],"r");       // Opening the filename

 if (fp == NULL)                   // check the contents of the file

    {
      cout <<"The File "<<argv[1]<<" Cannot be found"<<endl;

      exit(0);
    }


/***********************************************/
/***ignoring the first two lines in the file ***/
/***********************************************/

     for(i=0; i<45; i++)
       fscanf (fp,"%c", &ignore[i]);

/********************************************************/
/***reading the width and the height of the file*********/
/***and assign it to two variables ( width & height)*****/
/********************************************************/


   for(i=0; i<3; i++)
    {
       fscanf (fp,"%d", &array[i]);
       if(!i) cout << " ";
       if(i) cout << endl;
    }

     Rows = array[0];     // Get the width value
     Cols = array[1];     // Get the height value


         int  input[Rows][Cols],         // declearation of  the input image
              output[Rows][Cols],        // declearation of the output
          output3[Rows][Cols],       // declearation of the output of the extra features
              output2[Rows][Cols];       // dedlearation of the angled output



/***************************************************/
/****storing the matrix into 2 dimensional array****/
/***************************************************/

      for(i=0;i<Rows;i++)
    for( j=0;j<Cols;j++)
         fscanf (fp,"%d", &input[i][j]);


/*---------------------------(Done BY Jumaa & Omar)------------------------------------*/



/*********************/
/* the main interface*/
/*********************/
```

```
  char choice[1],      // String variables for user choice
    choice2[1];


// initializing the ouptut & the angled output

    for(i=0;i<Rows;i++)
     for(j=0;j<Cols;j++)
       output[i][j]=0;    // Finishing initializing



/*------------------------------------------------------------------------------------*/

 start:

/***********/
/* welcome */
/***********/

 char k;
  do {

    system("clear");    // clear the previous text
    cout<<"                                                             \n"
        <<"     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
        <<"    @                                                      @ \n"
        <<"    @           Etisalat Colleage of Engineering           @ \n"
        <<"    @          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@          @ \n"
        <<"    @                     Presents                         @ \n"
        <<"    @                  (Year 3 Project)                    @ \n"
        <<"    @                                                      @ \n"
        <<"    @    (^_^) Finding Edges in a Greyscale Image (^_^)    @ \n"
        <<"    @                                                      @ \n"
    <<"    @    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@    @ \n"
        <<"    @   @          Copyrights (2002-2003)              @   @ \n"
        <<"    @   @         coded by:1125,1133 & 1147            @   @ \n"
        <<"    @   @       Supervised by: Dr.Jamal Zemerly        @   @ \n"
    <<"    @    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@    @ \n"
    <<"    @                                                      @ \n"
    <<"    @         (^_^) -> (To Start press c then Enter) !     @ \n"
        <<"    @                                                      @ \n"
        <<"     @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n";
    cin>>k;


  }while(k!='c' && k!='C');  // check if character is c or not

/*------------------------------------------------------------------------------------*/


  system("clear"); // clear the previous text


/************************************************/
/* introduction of the program (improved later) */
/************************************************/

 do{

    cout<<"                                                               \n\n"
        <<"              @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@             \n"
    <<"             @                                      @            \n"
        <<"             @  (^_^)  PROGRAM INTRODUCTION  (^_^)  @            \n"
    <<"             @                                      @            \n"
        <<"              @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@             \n"
        <<"    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
        <<"   @                                                           @ \n"
        <<"   @           Finding Edges in a Greyscale Image              @ \n"
        <<"   @         -------------------------------------             @ \n"
        <<"   @                                                           @ \n"
        <<"   @  This program will help you find edges in an image        @ \n"
        <<"   @  with a PGM format by using a set of edge operators.      @ \n"
        <<"   @                                                           @ \n"
        <<"   @  Facts about the five operators:                          @ \n"
        <<"   @  -------------------------------                          @ \n"
        <<"   @   - All of them are well known in Image Processing        @ \n"
        <<"   @   - 2 masks for each operator to filter the image in 2D   @ \n"
        <<"   @                                                           @ \n"
        <<"   @  The Algorithims:                                         @ \n"
        <<"   @  ----------------                                         @ \n"
        <<"   @   - Filter the image in 2D                                @ \n"
        <<"   @   - Get dx,dy (Horizaontal & Vertical edge values)        @ \n"
        <<"   @   - Calculate the Magnitude & Angle                       @ \n"
        <<"   @                                                           @ \n"
        <<"   @  Output:                                                  @ \n"
        <<"   @  -------                                                  @ \n"
    <<"   @   - Edge detected image                                   @ \n"
        <<"   @   - Thresholded image                                     @ \n"
        <<"   @   - Angled image                                          @ \n"
        <<"   @                                                           @ \n"
        <<"   @  Extra Features:                                          @ \n"
    <<"   @  ---------------                                          @ \n"
    <<"   @   - Superimposing                                         @ \n"
        <<"   @   - Thinning (under construction)                         @ \n"
        <<"   @                                                           @ \n"
        <<"   @                          (^_^) -> (To Continue press c )  @ \n"
        <<"   @                                   (Then Enter)            @ \n"
        <<"   @                                                           @ \n"
    <<"    @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ \n";
     cin>>k;

    }while(k!='c' && k!='C');   // check if character is c or not



/*--------------------------------------------(Done BY Omar)-----------------------------------
-*/



 system("clear");    //clear the previous text


/******************************************/
/* Displaying the original image using XV */
/******************************************/
  xv =new char[256];
  strcpy(xv, "xv -ge +450+10 -na Original ");
  strcat(xv, argv[1]);
  strcat(xv, "&");
  system (xv);
  fclose(fp);


/*--------------------------------------------(Done BY Jumaa)----------------------------------
--*/




    /********************/
    /* USER INTERFACE 1 */
    /********************/
    p1:

    cout<<"                                                                    \n\n"
        <<"                  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
    <<"                 @                                                 @  \n"
        <<"                 @      Finding Edges in a Greyscale Image         @  \n"
    <<"                 @                                                 @  \n"
        <<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
    <<"                 @                                                 @  \n"
    <<"                 @  Select one of the following processes:         @  \n"
    <<"                 @                                                 @  \n"
        <<"                 @  1) Filter the image using (Sobel Operator)     @  \n"
        <<"                 @  2) Filter the image using (Prewitt Operator)   @  \n"
        <<"                 @  3) Filter the image using (Robert Operator)    @  \n"
        <<"                 @  4) Filter the image using (Gradient Operator)  @  \n"
        <<"                 @  5) Filter the image using (Kirsch Operator)    @  \n"
        <<"                 @  6) End the Program Or press Ctrl+ c            @  \n"
        <<"                 @                                                 @  \n"
        <<"                 @                                                 @  \n"
        <<"                 @  Enter  a number between (1-6)                  @  \n"
        <<"                 @                                                  @  \n"
        <<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
    <<"                  # Your Choice is No: ";

     cin>>choice;
     cout<<"\n";

     system("clear"); // clear the previous text

/*---------------------------(Done BY Omar)------------------------------------*/

//In case of Invalid Input

 if(strlen(choice)>1)    // Check if the length of the string is greater than 1 character
   {

    cout<<"                                                                           \n"
        <<"          *-------------------------------------------------------------*\n"
        <<"          |                                                             |\n"
        <<"          |          !!! Invalid Input !!!  :(                          |\n"
        <<"          |   Please enter an appropriate choice in the space provided  |\n"
    <<"          |   N.B: your choice must be an integer number between(1-6)    |\n"
        <<"          |   any other choices will not be accepted.                   |\n"
        <<"          |                                                             |\n"
    <<"          *-------------------------------------------------------------*\n";

    goto p1;            // The program will move to where the p1: function is
   }

 int select;            // Decleration of an integer variable

 select=atoi(choice);   // Function to convert the string into ingteger and stor it in the vari
able (select)

/*---------------------------(Done BY Omar)------------------------------------*/

 switch(select)

{


 case 1:

          /*****************************/
          /* Apply the Sobel operator  */
          /*****************************/

   // Filter the image &
   // Calculate the value of dx & dy

       for(i=1;i<Rows-1;i++)
        for(j=1;j<Cols-1;j++)
      {

    // Calculate the value of the horizontal edge

      dx=   input[i-1][j-1]+2*input[i-1][j]+input[i-1][j+1]
      -(input[i+1][j-1]+2*input[i+1][j]+input[i+1][j+1]);

       // Calculate the value of the vertical edge

      dy=   input[i-1][j+1] - input[i-1][j-1]
      + 2*input[i][j+1] - 2*input[i][j-1]
      + input[i+1][j+1] - input[i+1][j-1];

       // Calculate the magnitude and the angle

      output[i][j]=(int)Magnitude(dx,dy);       // Function to calculate the magnitude
      output2[i][j]=(int)Angle(dx,dy);               // Function to calculate the angle

      }// End of the 2nd loop

                                                  break;
/*---------------------------(Done BY Omar)------------------------------------*/

 case 2:
```

```
          /****************************/
          /* Applay Prewitt operator  */
          /****************************/

   // Filter the image &
   // Calculate the value of dx & dy

       for(i=1;i<Rows-1;i++)
        for(j=1;j<Cols-1;j++)
      {

      // Calculate the value of the horizontal edge

      dx=   input[i-1][j-1]+input[i-1][j]+input[i-1][j+1]
          -(input[i+1][j-1]+input[i+1][j]+input[i+1][j+1]);

     // Calculate the value of the vertical edge

      dy=   input[i-1][j+1] - input[i-1][j-1]
      + input[i][j+1]   - input[i][j-1]
      + input[i+1][j+1] - input[i+1][j-1];

     // Calculate the magnitude and the angle

    output[i][j]=(int)Magnitude(dx,dy);        // Function to calculate the magnitude
    output2[i][j]=(int)Angle(dx,dy);                // Function to calculate the angle

      }// end of the 2nd loop

                                                   break;
/*----------------------------(Done BY Omar)------------------------------------*/

 case 3:

          /*****************************/
          /* Apply the Robert operator */
          /*****************************/

  // Filter the image &
  // Calculate the value of dx & dy

       for(i=1;i<Rows;i++)
        for(j=0;j<Cols-1;j++)
      {

    // Calculate the value of the horizontal edge

    dx=  input[i-1][j]   - input[i][j+1];

    // Calculate the value of the vertical edge

        dy=  input[i-1][j+1] - input[i][j];

    // Calculate the magnitude and the angle

    output[i][j]=(int)Magnitude(dx,dy);
    output2[i][j]=(int)Angle(dx,dy);

      }// End of the 2nd loop

                                                  break;
/*----------------------------(Done BY Majid)-----------------------------------*/
 case 4:

          /*******************************/
          /* Apply the Gradient operator */
          /*******************************/

  // Filter the image &
  // Calculate the value of dx & dy

       for(i=1;i<Rows;i++)
        for(j=0;j<Cols-1;j++)
      {

    // Calculate the value of the horizontal edge
```

```
      dx=  input[i-1][j]   - input[i][j+1];

    // Calculate the value of the horizontal edge

      dy=  input[i-1][j+1] - input[i][j];

    // Calculate the magnitude and the angle

    output[i][j]=(int)Magnitude(dx,dy);        // Function to calculate the magnitude
    output2[i][j]=(int)Angle(dx,dy);                // Function to calculate the angle

      }// End of the 2nd loop
                                                 break;
/*---------------------------(Done BY Majid)-----------------------------------*/

 case 5:

           /*****************************/
           /* Apply the Kirsch operator */
           /*****************************/

   // Filter the image &
   // Calculate the value of dx & dy

       for(i=1;i<Rows-1;i++)
        for(j=1;j<Cols-1;j++)
      {

    // Calculate the value of the horizontal edge

    dx =   (input[i-1][j-1]*3)  + (input[i-1][j]*3)  + (input[i-1][j+1]*3)
     + (input[i][j-1]*3)    + (input[i][j]*0)    + (input[i][j+1]*3)
         + (input[i+1][j-1]*-5) + (input[i+1][j]*-5) + (input[i+1][j+1]*-5) ;

    // Calculate the value of the Vertical edge

    dy =   (input[i-1][j-1]*-5) + (input[i-1][j]*3)  + (input[i-1][j+1]*3)
         + (input[i][j-1]*-5)   + (input[i][j]*0)    + (input[i][j+1]*3)
         + (input[i+1][j-1]*-5) + (input[i+1][j]*3)  + (input[i+1][j+1]*3) ;


    // Calculate the magnitude and the angle

    output[i][j]=(int)Magnitude(dx,dy);        // Function to calculate the magnitude
    output2[i][j]=(int)Angle(dx,dy);                // Function to calculate the angle

 }// End of the 2nd loop

                                                 break;
/*---------------------------(Done BY Jumaa)-----------------------------------*/


 case 6:
   system("clear");
         goto end;     // The program will move to where the end: function is

/*----------------------------------------------------------------------------*/

 default:
     cout  <<"          *-------------------------------------------------------------*\n"
           <<"          |                                                             |\n"
           <<"          |                  !!! Invalid Input !!!  :(                  |\n"
           <<"          |   Please enter an appropriate choice in the space provided   |\n"
       <<"          |   N.B: your choice must be an integer number between(1-4)    |\n"
           <<"          |   any other choices will not be accepted.                    |\n"
           <<"          |                                                             |\n"
       <<"          *-------------------------------------------------------------*\n";

     goto p1;  // The program will move to where the p1: function is


}//End of switch

/*---------------------------(Done BY Omar)-----------------------------------*/
/*----------------------------------------------------------------------------*/
```

```
     system("clear");          // clear the previous text

    /********************/
    /* USER INTERFACE 2 */
    /********************/

   P2:
   cout <<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
    <<"                @                                                  @  \n"
        <<"                @      Finding Edges in a Greyscale Image          @  \n"
    <<"                @                                                  @  \n"
        <<"                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
    <<"                @                                                  @  \n"
        <<"                @ Select one of the following output:              @  \n"
        <<"                @                                                  @  \n"
        <<"                @  1)Detected edge image                           @  \n"
        <<"                @  2)Thresholded image                             @  \n"
        <<"                @  3)Angled image (Extra)                          @  \n"
        <<"                @                                                  @  \n"
        <<"                @     Enter  a number between (1-3)                @  \n"
        <<"                @     Press Esc to return back                     @  \n"
        <<"                @                                                  @  \n"
        <<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
        <<"                  # Your Choice is No: ";

    cin>>choice;
    system("clear");          // clear the previous text


 // For Esc process
    int E;
    E=(int)choice[0];         // convert the character into integer
    if(E==27)
      goto p1;

 //In case of Invalid Input

    if(strlen(choice)>1)     // Check if the length of the string is greater than 1 character
   {
     cout  <<"          *------------------------------------------------------------* \n"
           <<"          |                                                            | \n"
       <<"          |         !!! Invalid Input !!!  :(                          | \n"
           <<"          |   Please enter appropriate choice in the space provided    | \n"
       <<"          |   N.B: your choice must be an integer number between(1-3) | \n"
           <<"          |   any other choices will not be accepted.                  | \n"
       <<"          |                                                            | \n"

       <<"          *------------------------------------------------------------* \n";

    goto P2;                // The program will move to where the P2: function is

   }


 select=atoi(choice);      // Function to convert the string into integer and stor it in the va
riable (select)

/*---------------------------(Done BY Omar)-------------------------------------*/

switch(select)

{

 case 1:

          /************************************/
          /* Getting the edge detected image  */
          /************************************/

/*****************/
/* file writing  */
/*****************/

  fp2= fopen("image","w");
  fprintf (fp2,"P2\n# SSSoftware Year3 Project\n");     //Print The "P2" and The Copy rights in
to the file
  fprintf (fp2,"%d %d\n%d\n",Rows,Cols,array[2]);       //Print The Size and Max greyscale leve
```

```
l (255)

/******************/
/*Print The  matrix*/
/******************/

  for (i=0;i<Rows;i++)
      for ( j=0;j<Cols;j++)
    {
      fprintf (fp2,"%d ", output[i][j]);
          if (j % 17 == 0)
      fprintf (fp2,"\n");

       }

/**************************************/
/* Displaying the new image using XV  */
/**************************************/

  strcpy(xv, "xv -ge +450+10 -na Detected image&");
  system (xv);
  fclose(fp2);

                                                  break;

/*---------------------------(Done BY Omar & Jumma)------------------------------------*/

 case 2:

          /**************************/
          /* applying the threshold */
          /**************************/
  int T;

  do{
    system("clear");
    cout<<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
        <<"                @                                                 @  \n"
        <<"                @      Finding Edges in a Greyscale Image         @  \n"
    <<"                @                                                 @  \n"
        <<"                @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
    <<"                @                                                 @  \n"
    <<"                @   (^_^) -> Please Enter the threshold value !   @  \n"
        <<"                @                                                 @  \n"
        <<"                @ N.B: (it should be a number between (0-255))    @  \n"
        <<"                @                                                 @  \n"
        <<"                 @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
        <<"                  # The threshold value is : ";
       cin>>T;
      cout<<"\n\n";
  }while(T<0 || T>255);

               for(i=1;i<Rows-1;i++)
                 for(j=1;j<Cols-1;j++)
                 {
                 if(T>output[i][j])
                    output[i][j]=0;
                 if(T<output[i][j])
                 output[i][j]=255;
                 }


// Getting the output after thresholding the image

/*****************/
/* file writing  */
/*****************/
  fp2= fopen("image","w");
  fprintf (fp2,"P2\n# SSSoftware Year3 Project\n");     //Print The "P2" and The Copy rights in
to the file
  fprintf (fp2,"%d %d\n%d\n",Rows,Cols,array[2]);       //Print The Size and Max greyscale leve
l (255)

/******************/
/*Print The  matrix*/
/******************/
```

```
  for (i=0;i<Rows;i++)
      for ( j=0;j<Cols;j++)
    {
      fprintf (fp2,"%d ", output[i][j]);
      if (j % 17 == 0)
      fprintf (fp2,"\n");
    }

/*************************************/
/* Displaying the new image using XV  */
/*************************************/

  strcpy(xv, "xv -ge +450+10 -na Thresholded image&");
  system (xv);
  fclose(fp2);
  cout<<"\n\n";

                                                  break;
/*---------------------------(Done BY Majid & Jumaa)------------------------------------*/

 case 3:

          /************************/
          /* Get the Angled image */
          /************************/


/*****************/
/* file writing  */
/*****************/

  fp2= fopen("image","w");
  fprintf (fp2,"P2\n# SSSoftware Year3 Project\n");     //Print The "P2" and The Copy rights in
to the file
  fprintf (fp2,"%d %d\n%d\n",Rows,Cols,array[2]);       //Print The Size and Max greyscale leve
l (255)

/******************/
/*Print The  matrix*/
/******************/

  for (i=0;i<Rows;i++)
      for ( j=0;j<Cols;j++)
    {
      fprintf (fp2,"%d ", output2[i][j]);
      if (j % 17 == 0)
      fprintf (fp2,"\n");
    }

/*************************************/
/* Displaying the new image using XV  */
/*************************************/

  strcpy(xv, "xv -ge +450+10 -na Angled image&");
  system (xv);
  fclose(fp2);

                                                  break;
/*---------------------------(Done BY Omar)------------------------------------*/

 default:

     cout  <<"          *-----------------------------------------------------------*\n"
           <<"          |                                                           |\n"
       <<"          |               !!! Invalid Input !!!  :(                  |\n"
           <<"          |   Please enter appropriate choice in the space provided   |\n"
       <<"          |   N.B: your choice must be an integer number between(1-3) |\n"
           <<"          |   any other choices will not be accepted.                 |\n"
       <<"          |                                                           |\n"
       <<"          *-----------------------------------------------------------*\n\n";
         goto P2;

}//End of 2nd switch


/*--------------------------------------------------------*/
/*--------------------------------------------------------*/
```

```
/******************/
/* Extra Features */
/******************/


P3:

   cout<<"                   @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
       <<"                  @                                                  @  \n"
       <<"                  @      Finding Edges in a Greyscale Image         @  \n"
       <<"                  @                                                  @  \n"
       <<"                  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
       <<"                  @                                                  @  \n"
       <<"                  @ Select one of the following features:           @  \n"
       <<"                  @                                                  @  \n"
       <<"                  @  1)Superimposing                                 @  \n"
       <<"                  @  2)Thinning (Extra)                              @  \n"
       <<"                  @  3)End                                           @  \n"
       <<"                  @                                                  @  \n"
       <<"                  @     Enter  a number between (1-3)               @  \n"
       <<"                  @                                                  @  \n"
       <<"                  @                                                  @  \n"
       <<"                   @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@   \n"
       <<"                    # Your Choice is No: ";

    cin>>choice;
    cout<<"\n";

    system("clear");  // clear the previous text


//In case of Invalid Input

 if(strlen(choice)>1)
   {
     cout  <<"          *-----------------------------------------------------------*\n"
           <<"          |                                                           |\n"
       <<"          |            !!! Invalid Input !!!  :(                   |\n"
           <<"          |   Please enter appropriate choice in the space provided   |\n"
       <<"          |   N.B: your choice must be an integer number between(1-3) |\n"
           <<"          |   any other choices will not be accepted.                 |\n"
       <<"          |                                                           |\n"
       <<"          *-----------------------------------------------------------*\n";
         goto P3;
   }


select=atoi(choice);

/*---------------------------(Done BY Omar)------------------------------------*/

switch(select)

{

 case 1:

       /***************************************************/
       /* Get the Superimposed image                       */
       /***************************************************/


/*****************/
/* file writing  */
/*****************/


  fp2= fopen("image2","w");
  fprintf (fp2,"P2\n# SSSoftware Year3 Project\n");     //Print The "P2" and The Copy rights in
to the file
  fprintf (fp2,"%d %d\n%d\n",Rows,Cols,array[2]);       //Print The Size and Max greyscale leve
l (255)


/*****************/
/* Superimposing */
```

```
/*****************/

  //copy the output image to another 2D array

for (i=0;i<Rows;i++)
      for ( j=0;j<Cols;j++)
    output3[i][j]=output[i][j];

  //superimosing with the original

for (i=0;i<Rows;i++)
      for ( j=0;j<Cols;j++)
   {
     if(output3[i][j]!=255)
       output3[i][j]=input[i][j];
   }


/******************/
/*Print The  matrix*/
/******************/

  for (i=0;i<Rows;i++)
    {
      for ( j=0;j<Cols;j++)
    {
      fprintf (fp2,"%d ", output3[i][j]);
      if (j % 17 == 0)
        fprintf (fp2,"\n");
    }
    }


/**************************************/
/* Displaying the new image using XV  */
/**************************************/

     strcpy(xv, "xv -ge +450+10 -na Output image2&");
     system (xv);
     fclose(fp2);
     cout<<"\n\n";

                                                   break;
/*---------------------------(Done BY Omar)------------------------------------*/



                           // Thinning process
 case 2:

       cout<<"          *------------------------------------------------------------*\n"
           <<"          |                                                            |\n"
           <<"          |            !!! UNDER CONSTRUCTION !!!                      |\n"
           <<"          |        !!!!  Due to Time Limitation !!!!                   |\n"
           <<"          |                                                            |\n"
           <<"          *------------------------------------------------------------*\n";
        cout<<"\n\n"; break;

                                              // In case ending the program
 case 3:
         goto end;

 default:                                        // In case of invalid input

       cout<<"          *---------------------------------------------------------------*\n"
           <<"          |                                                               |\n"
       <<"          |                !!! Invalid Input !!!  :(                      |\n"
           <<"          |   Please enter an appropriate choice in the space provided   |\n"
       <<"          |   N.B: your choice must be an integer number between(1-3)    |\n"
           <<"          |   any other choices will not be accepted.                     |\n"
       <<"          |                                                               |\n"

       <<"          *--------------------------------------------------------------*\n";
         goto P3;


}// End of progress
```

```
/*--------------------------------------------------------*/
/*--------------------------------------------------------*/


end:
      cout<<"                                                                     \n"
          <<"              @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@      \n"
      <<"             @                                                 @     \n"
          <<"             @    (^_^) THANKS FOR USING OUR PROGRAM (^_^)    @     \n"
      <<"             @                                                 @     \n"
          <<"              @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@      \n"
          <<"          @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"

          <<"         @                                                         @ \n"
          <<"         @ #If you want to change the operator please Enter (Y/N)  @ \n"
          <<"         @     N.B: your choice must be either (Y) or (N)          @ \n"
          <<"         @          Capital or small letter gives the same result  @ \n"
          <<"         @                                                         @ \n"
          <<"           @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  \n"
          <<"            # Your Choice is: ";
       cin>>choice2;

       system("clear"); // clear the previous text


// In case of invalid input

if(strlen(choice2)>1)
 {

       cout<<"          *---------------------------------------------------------------*\n"
           <<"          |                                                               |\n"
       <<"          |                  !!! Invalid Input !!!  :(                    |\n"
           <<"          |   Please enter  an appropriate choice in the space provided   |\n"
       <<"          |   N.B: Any other choices will not be accepted.                |\n"
       <<"          |                                                               |\n"

       <<"          *---------------------------------------------------------------*\n";
                        goto end;
 }

char f;
f=choice2[0];

switch(f)

{
    case'Y':
    case'y':
      system("clear");
        goto p1;

    case'N':
    case'n':
             break;
    default:

       cout<<"          *-----------------------------------------------------------*\n"
           <<"          |                                                           |\n"
           <<"          |              !!! Invalid Input !!!  :(                    |\n"
           <<"          |   Please enter appropriate choice in the space provided   |\n"
       <<"          |   N.B: Any other choices will not be accepted.           |\n"
           <<"          |                                                           |\n"
           <<"          *-----------------------------------------------------------*\n";
                        goto end;

}// end of 3rd switch


return 0;  // END OF THE PROGRAM

}
```