Project
Sudoku Solver

Atma Anand (SC12B156)
Anupam Das (SC12B144)

Sub: C Programming Lab **Department**: Physical Sciences
Date of Submission: November 5, 2013

Ojectives:

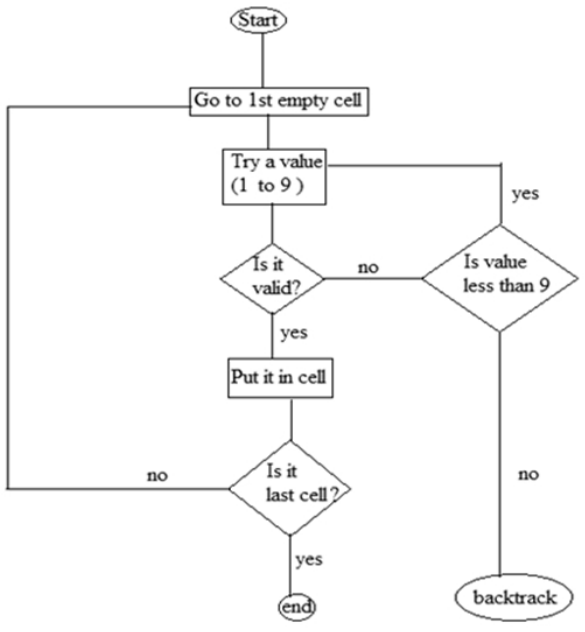
•To implement an approach to solve computationally intensive problems like Sudoku.
•To attempt solve the classical Sudoku puzzle by conventional brute force and Backtracking method.
•To display the thus obtained Solution in a 24BPP image (1024x1024).

Softwares Used:

• **DevC** IDE in Windows OS

•**IrfanView** - An Advanced Image Viewer

Flowchart:

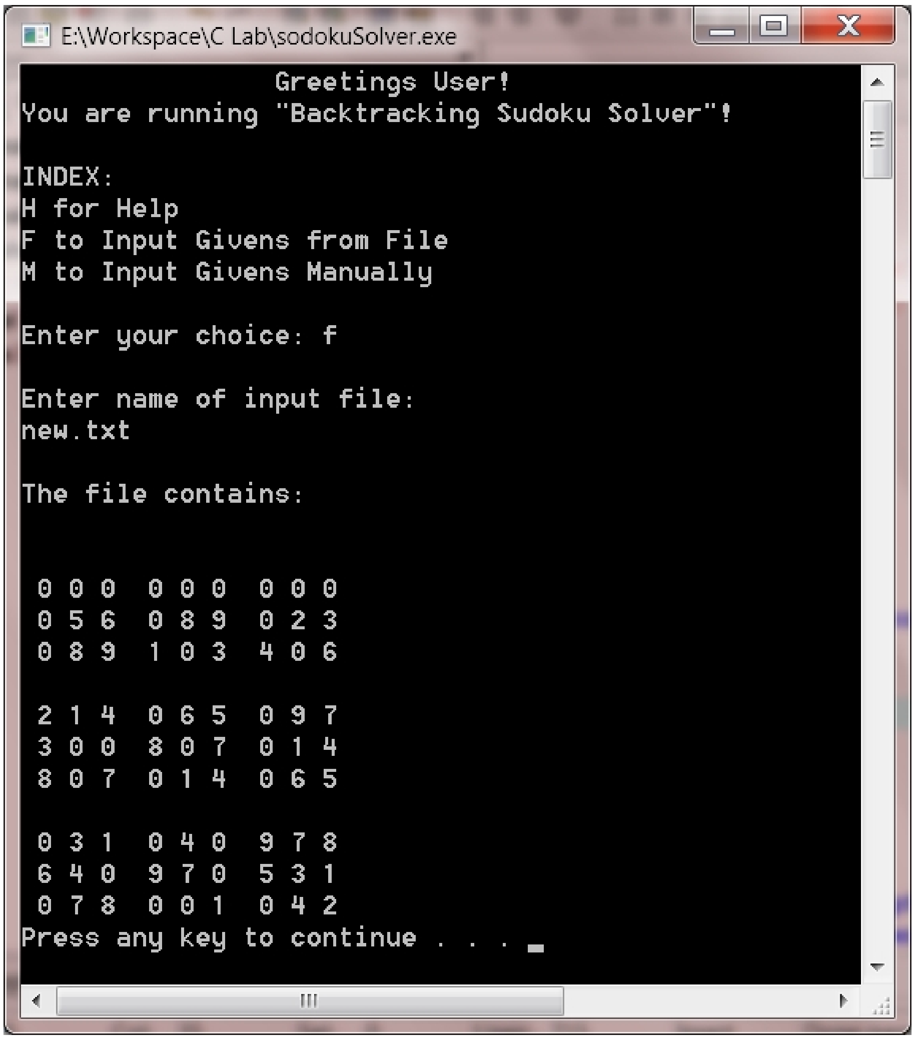


Brief Logic:

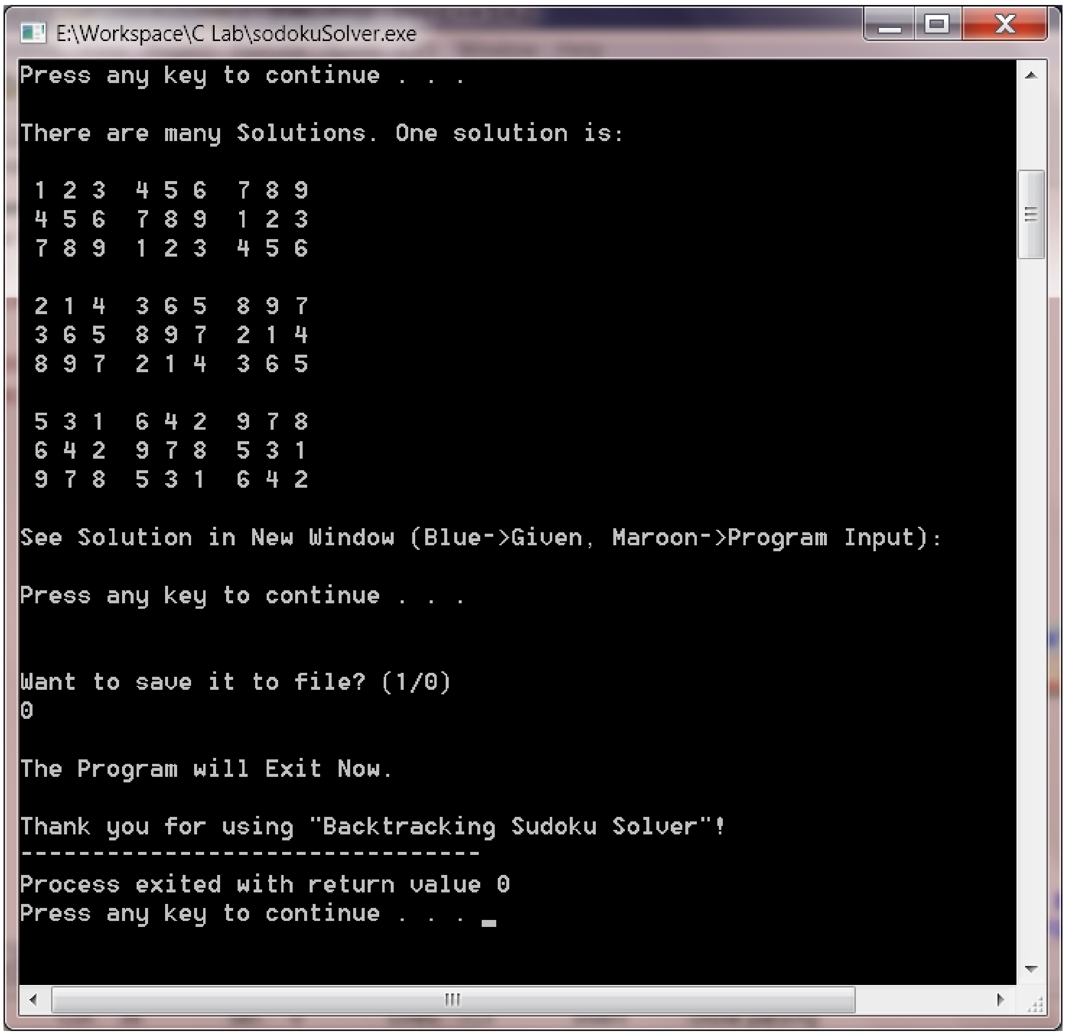
•We need to check our input to see if it is valid.
  ·Are the rows ok? Are the columns ok? Are the boxes ok?
  ·If any are invalid, then we can return false.
•We also need to reduce our options.
  ·Currently our options are only based on the initial board we filled in.
  ·We want to further reduce our options by looking at what is in each row, column and box.
•After all of that is done, we want to fill in the next best choice to make a guess.
  ·What cell would be the best cell to make a guess at?
•If its possible, set the answer to the next available option.
•Recursively solve the puzzle.
•If that choice worked, were done!
•Otherwise, undo that choice and redo our options.
•If none of our options worked, then its not solvable!

•The image was formed using precise pixel seeking and elementary geometry concepts.

Output Screens:

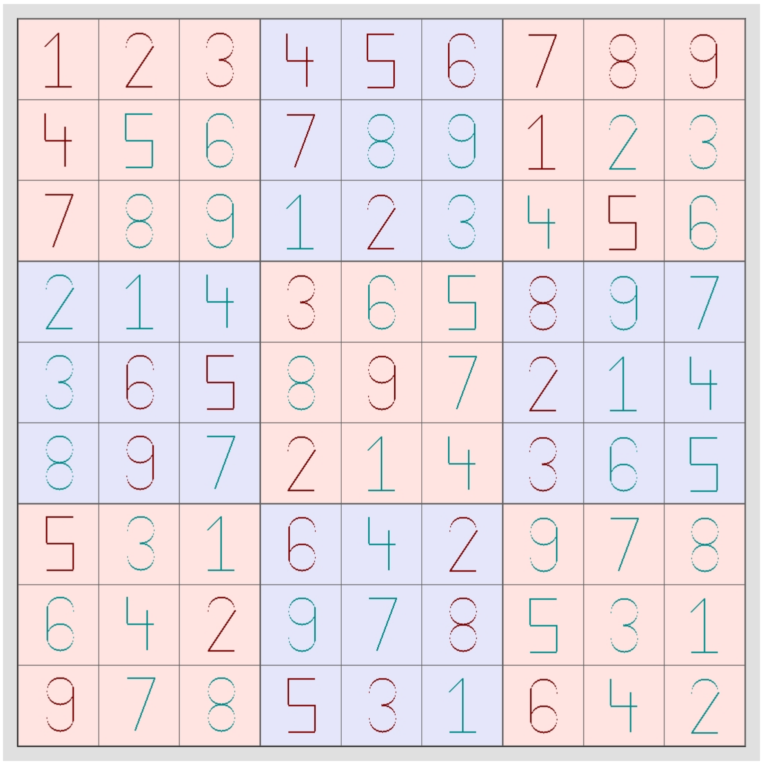


```
                Greetings User!
You are running "Backtracking Sudoku Solver"!

INDEX:
H for Help
F to Input Givens from File
M to Input Givens Manually

Enter your choice: f

Enter name of input file:
new.txt

The file contains:


 0 0 0  0 0 0  0 0 0
 0 5 6  0 8 9  0 2 3
 0 8 9  1 0 3  4 0 6

 2 1 4  0 6 5  0 9 7
 3 0 0  8 0 7  0 1 4
 8 0 7  0 1 4  0 6 5

 0 3 1  0 4 0  9 7 8
 6 4 0  9 7 0  5 3 1
 0 7 8  0 0 1  0 4 2
Press any key to continue . . .
```



```
Press any key to continue . . .

There are many Solutions. One solution is:

 1 2 3  4 5 6  7 8 9
 4 5 6  7 8 9  1 2 3
 7 8 9  1 2 3  4 5 6

 2 1 4  3 6 5  8 9 7
 3 6 5  8 9 7  2 1 4
 8 9 7  2 1 4  3 6 5

 5 3 1  6 4 2  9 7 8
 6 4 2  9 7 8  5 3 1
 9 7 8  5 3 1  6 4 2

See Solution in New Window (Blue->Given, Maroon->Program Input):

Press any key to continue . . .


Want to save it to file? (1/0)
0

The Program will Exit Now.

Thank you for using "Backtracking Sudoku Solver"!
--------------------------------
Process exited with return value 0
Press any key to continue . . .
```

...



1	2	3	4	5	6	7	8	9
4	5	6	7	8	9	1	2	3
7	8	9	1	2	3	4	5	6
2	1	4	3	6	5	8	9	7
3	6	5	8	9	7	2	1	4
8	9	7	2	1	4	3	6	5
5	3	1	6	4	2	9	7	8
6	4	2	9	7	8	5	3	1
9	7	8	5	3	1	6	4	2

Result

The program successfully implements all the objectives.

Discussion:

•We have successfully implemented a program to Solve any given Sudoku.
•This can be used to Generate Sudoku by trying different possible permutations of a solved Sudoku, delete cells according to intended difficulty and check if a unique solution exists.
•The image making algorithm can be implemented to make more complex patterns.

Limitations:

•Highly resource intensive program.
•Complexity: $O(n^9)$, where n is the no. of blank cells! (Max. 64 cells can be blank.)
•Redundant coding (Too long.)
•Lack of interactive input from user.
•The image formed lacks certain pixels due to round off errors.

Program Code:

```
//(Be Patient Till the Last!)

#include <stdio.h>
#include <windows.h>
#include <shellapi.h>
#include <stdlib.h>
#include <math.h>
#define s 1024
#define c1 54

int un[9][9],f=0,sol[9][9];

void analyser()
{
        int r,c,k,i,j;
        for(r=0;r<9;r++)
          for(c=0;c<9;c++)
          {
                k=un[r][c];
                if(k<1 || k>9)
                   continue;
                for(i=0;i<9;i++)
                  if(un[r][i]==k && i!=c)
                  {
                        f=-1;
                        return;
                  }
                for(i=0;i<9;i++)
                  if(un[i][c]==k && i!=r)
                  {
                        f=-1;
                        return;
```

```
                    }
                for(i=(r/3)*3;i<((r/3)*3+3);i++)
                    for(j=(c/3)*3;j<((c/3)*3+3);j++)
                        {
                        if(r==i && c==j)
                                        continue;
                                if(un[i][j]==k)
                                {
                                        f=-1;
                                        return;
                                }
                        }
          }
}

void solver(int so[9][9],int r,int c)
{
        int t[9][9],i,j,k;
        if(f>1 || f<0)
                return;
        for(i=0;i<9;i++)
          for(j=0;j<9;j++)
             t[i][j]=so[i][j];
        if(t[r][c]<1 || t[r][c]>9)
        {
        k=1;
        start:
        for(;k<10;k++)
        {
                for(i=0;i<9;i++)
                  if(t[r][i]==k || t[i][c]==k)
                  {
                          k++;
                          goto start;
                  }
                for(i=(r/3)*3;i<((r/3)*3+3);i++)
                    for(j=(c/3)*3;j<((c/3)*3+3);j++)
                        if(t[i][j]==k)
                                {
                                        k++;
                                        goto start;
                                }
                t[r][c]=k;
                if(r==8 && c==8)
                {
                        ++f;
                        if(f==1)
                          for(i=0;i<9;i++)
                                for(j=0;j<9;j++)
                                sol[i][j]=t[i][j];
                        return;
                }
                else if(c==8)
                        solver(t,r+1,0);
                else
                        solver(t,r,c+1);
                }
        }
        else
        {
                if(r==8 && c==8)
                {
                        ++f;
                        if(f==1)
                           for(i=0;i<9;i++)
                                   for(j=0;j<9;j++)
                                           sol[i][j]=t[i][j];
                        return;
                }

                else if(c==8)
                        solver(t,r+1,0);
                else
                        solver(t,r,c+1);
        }
}

void imager(unsigned char *im)
{
        int i,j,a,n=20,b,w=108,h,k,al,be,r=0,g=102,bl=0;
        //BACKGROUND COLOR
        for(i=n;i<s-n;i++)
        {
                for(j=n;j<s-n;j++)
                {
                        b=(((i-n)/328)+((j-n)/328))%2;
                        if(b)
                        {
                                *(im+(s-i-1)*s*3+j*3)=250;
                                *(im+(s-i-1)*s*3+j*3+1)=230;
                                *(im+(s-i-1)*s*3+j*3+2)=230;
                        }
                        else
                        {
                                *(im+(s-i-1)*s*3+j*3)=225;
                                *(im+(s-i-1)*s*3+j*3+2)=255;
                                *(im+(s-i-1)*s*3+j*3+1)=228;
                        }
                }
        }
        //GRID LINES
        a=0;
        for(b=n;b<s-n;b+=109)
        {
                        black:
                        for(j=n;j<s-n;j++)
                        {
                                *(im+(b)*s*3+(j)*3)=96;
                                *(im+(b)*s*3+(j)*3+1)=96;
                                *(im+(b)*s*3+(j)*3+2)=96;
                                *(im+(j)*s*3+(b)*3)=96;
                                *(im+(j)*s*3+(b)*3+1)=96;
                                *(im+(j)*s*3+(b)*3+2)=96;
                        }
                        ++a;
                        if(a%4==0)
                        {
                                ++b;
                                goto black;
                        }
        }
        //NUMBERS
        for(a=0;a<9;a++)
        {
                for(b=0;b<9;b++)
                {
                        if(un[a][b]==0)
                        {
                          r=128;
                          bl=g=0;
                          n=sol[a][b];
                        }
                        else
                        {
                          r=0;
                          bl=139;
                          g=139;
                          n=sol[a][b];
                        }
                        if(n<=0 || n>9)
                            continue;
                        h=109*b+a/3+21;
                        k=109*a+b/3+21;
                        switch(n)
                        {
                                case 1:
                                        for(i=k+w/6;i<k+5*w/6;i++)
                                        {
                                                *(im+(s-i-1)*s*3+(w/2+h)*3)=bl;
                                                *(im+(s-i-1)*s*3+(w/2+h)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(w/2+h)*3+2)=r;
                                                *(im+(s-i-1)*s*3+(w/2+h-1)*3)=bl;
                                                *(im+(s-i-1)*s*3+(w/2+h-1)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(w/2+h-1)*3+2)=r;
                                        }
                                        i=k+5*w/6;
                                        for(j=h+w/3;j<h+2*w/3;j++)
                                        {
                                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-2)*s*3+j*3+2)=r;
                                        }
                                        for(i=h+w/3;i<h+w/2;i++)
                                        {
                                                j=s-1-h-w/2-k-w/6+i;
                                                *(im+(j)*s*3+(i)*3)=bl;
                                                *(im+(j)*s*3+(i)*3+1)=g;
                                                *(im+(j)*s*3+(i-1)*3+2)=r;
                                                *(im+(j)*s*3+(i-1)*3)=bl;
                                                *(im+(j)*s*3+(i-1)*3+1)=g;
                                                *(im+(j)*s*3+(i)*3+2)=r;
                        }
                        break;
                case 2:
                        al=h+w/2;
                        be=k+w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/3;j<=h+2*w/3;j++)
                        {
                                i=-3*(j-h-2*w/3)/2+k+w/3;
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-1)*s*3+(j+1)*3)=bl;
                                *(im+(s-i-1)*s*3+(j+1)*3+1)=g;
                                *(im+(s-i-1)*s*3+(j+1)*3+2)=r;
                                i++;
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-1)*s*3+(j+1)*3)=bl;
                                *(im+(s-i-1)*s*3+(j+1)*3+1)=g;
                                *(im+(s-i-1)*s*3+(j+1)*3+2)=r;
                        }
                        i=k+5*w/6;
                        for(j=h+w/3;j<h+2*w/3;j++)
                        {
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        break;
                case 3:
                        al=h+w/2;
                        be=k+w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/2-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        al=h+w/2;
                        be=k+2*w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/2-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        break;
                case 8:
                        al=h+w/2;
                        be=k+w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        al=h+w/2;
                        be=k+2*w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        break;
                case 9:
                        al=h+w/2;
                        be=k+w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        al=h+w/2;
                        be=k+2*w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(i=k+w/3-5;i<k+2*w/3+5;i++)
                        {
                                *(im+(s-i-1)*s*3+(w/3*2+h)*3)=bl;
                                *(im+(s-i-1)*s*3+(w/3*2+h)*3+1)=g;
                                *(im+(s-i-1)*s*3+(w/3*2+h)*3+2)=r;
                                *(im+(s-i-1)*s*3+(w/3*2+h-1)*3)=bl;
                                *(im+(s-i-1)*s*3+(w/3*2+h-1)*3+1)=g;
                                *(im+(s-i-1)*s*3+(w/3*2+h-1)*3+2)=r;
                        }
                        break;
                case 6:
                        al=h+w/2;
                        be=k+w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        al=h+w/2;
                        be=k+2*w/3;
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be+sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(j=h+w/3-1;j<=h+2*w/3;j++)
                        {
                                i=be-sqrt(w*w/36.0-(j-al)*(j-al));
                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                *(im+(s-i-2)*s*3+j*3+2)=r;
                        }
                        for(i=k+w/3-5;i<k+2*w/3+5;i++)
                        {
                                *(im+(s-i-1)*s*3+(w/3+h)*3)=bl;
                                *(im+(s-i-1)*s*3+(w/3+h)*3+1)=g;
                                *(im+(s-i-1)*s*3+(w/3+h)*3+2)=r;
                                *(im+(s-i-1)*s*3+(w/3+h+1)*3)=bl;
                                *(im+(s-i-1)*s*3+(w/3+h+1)*3+1)=g;
                                *(im+(s-i-1)*s*3+(w/3+h+1)*3+2)=r;
                        }
                        break;
                case 4:
                        for(i=k+w/6;i<k+w/2;i++)
                        {
                                *(im+(s-i-1)*s*3+(w/3+h)*3)=bl;
                                *(im+(s-i-1)*s*3+(w/3+h)*3+1)=g;
                                *(im+(s-i-1)*s*3+(w/3+h)*3+2)=r;
                                *(im+(s-i-1)*s*3+(w/3+h-1)*3)=bl;
```

```
                        *(im+(s-i-1)*s*3+(w/3+h-1)*3+1)=g;
                        *(im+(s-i-1)*s*3+(w/3+h-1)*3+2)=r;
                }
                for(i=k+w/3;i<k+5*w/6;i++)
                {
                        *(im+(s-i-1)*s*3+(7*w/12+h)*3)=bl;
                        *(im+(s-i-1)*s*3+(7*w/12+h)*3+1)=g;
                        *(im+(s-i-1)*s*3+(7*w/12+h)*3+2)=r;
                        *(im+(s-i-1)*s*3+(7*w/12+h-1)*3)=bl;
                        *(im+(s-i-1)*s*3+(7*w/12+h-1)*3+1)=g;
                        *(im+(s-i-1)*s*3+(7*w/12+h-1)*3+2)=r;
                }
                i=k+w/2;
                for(j=h+w/3;j<h+2*w/3;j++)
                {
                        *(im+(s-i-1)*s*3+(j)*3)=bl;
                        *(im+(s-i-1)*s*3+(j)*3+1)=g;
                        *(im+(s-i-1)*s*3+j*3+2)=r;
                        *(im+(s-i-2)*s*3+(j)*3)=bl;
                        *(im+(s-i-2)*s*3+(j)*3+1)=g;
                        *(im+(s-i-2)*ss*3+j*3+2)=r;
                }
                break;
        case 7:
                i=k+w/6;
                for(j=h+w/3;j<h+2*w/3;j++)
                {
                        *(im+(s-i-1)*s*3+(j)*3)=bl;
                        *(im+(s-i-1)*s*3+(j)*3+1)=g;
                        *(im+(s-i-1)*s*3+j*3+2)=r;
                        *(im+(s-i-2)*s*3+(j)*3)=bl;
                        *(im+(s-i-2)*s*3+(j)*3+1)=g;
                        *(im+(s-i-2)*s*3+j*3+2)=r;
                }
                for(i=k+w/6;i<k+5*w/6;i++)
                {
                        j=-3*(i-k-w/6)/8+h+2*w/3;
                        *(im+(s-i-1)*s*3+(j)*3)=bl;
                        *(im+(s-i-1)*s*3+(j)*3+1)=g;
                        *(im+(s-i-1)*s*3+j*3+2)=r;
                        *(im+(s-i-1)*s*3+(j+1)*3)=bl;
                        *(im+(s-i-1)*s*3+(j+1)*3+1)=g;
                        *(im+(s-i-1)*s*3+(j+1)*3+2)=r;
                }
                break;
        case 5:
                i=k+w/6;
                for(j=h+w/3;j<h+2*w/3;j++)
                {
                        *(im+(s-i-1)*s*3+(j)*3)=bl;
                        *(im+(s-i-1)*s*3+(j)*3+1)=g;
                        *(im+(s-i-1)*s*3+j*3+2)=r;
                        *(im+(s-i-2)*s*3+(j)*3)=bl;
                        *(im+(s-i-2)*s*3+(j)*3+1)=g;
                        *(im+(s-i-2)*s*3+j*3+2)=r;
                }
                i=k+5*w/6;
                for(j=h+w/3;j<h+2*w/3;j++)
```

```
                                        {
                                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-2)*s*3+j*3+2)=r;
                                        }
                                        for(i=k+w/6;i<k+w/2;i++)
                                        {
                                                *(im+(s-i-1)*s*3+(w/3+h)*3)=bl;
                                                *(im+(s-i-1)*s*3+(w/3+h)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(w/3+h)*3+2)=r;
                                                *(im+(s-i-1)*s*3+(w/3+h-1)*3)=bl;
                                                *(im+(s-i-1)*s*3+(w/3+h-1)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(w/3+h-1)*3+2)=r;
                                        }
                                        for(i=k+w/2;i<k+5*w/6;i++)
                                        {
                                                *(im+(s-i-1)*s*3+(2*w/3+h)*3)=bl;
                                                *(im+(s-i-1)*s*3+(2*w/3+h)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(2*w/3+h)*3+2)=r;
                                                *(im+(s-i-1)*s*3+(2*w/3+h-1)*3)=bl;
                                                *(im+(s-i-1)*s*3+(2*w/3+h-1)*3+1)=g;
                                                *(im+(s-i-1)*s*3+(2*w/3+h-1)*3+2)=r;
                                        }
                                        i=k+w/2;
                                        for(j=h+w/3;j<h+2*w/3;j++)
                                        {
                                                *(im+(s-i-1)*s*3+(j)*3)=bl;
                                                *(im+(s-i-1)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-1)*s*3+j*3+2)=r;
                                                *(im+(s-i-2)*s*3+(j)*3)=bl;
                                                *(im+(s-i-2)*s*3+(j)*3+1)=g;
                                                *(im+(s-i-2)*s*3+j*3+2)=r;
                                        }
                                        break;

                        }
                }
        }
        //WHITE WRAPING
        for(i=0;i<s;i++)
        {
                for(j=0;j<19;j++)
                {
                        for(a=0;a<3;a++)
                        {
                                *(im+(s-i-1)*s*3+j*3+a)=224;
                                *(im+(i)*s*3+(s-j-1)*3+a)=224;
                                *(im+(s-j-1)*s*3+i*3+a)=224;
                                *(im+j*s*3+i*3+a)=224;
                        }
                }
        }
}

int main()
```

```
{
        unsigned char *im,p;
        char c='f';
        im=(unsigned char*)malloc(s*s*3);
        int i,j,k;
        FILE *fp1,*fp2,*fp3;
        char in[30],out[30];
        inception:
        printf("                 Greetings User! \n");
        printf("You are running \"Backtracking Sudoku Solver\"!\n");
        printf("\nINDEX: \n");
        printf("H for Help \nF to Input Givens from File \n");
        printf("M to Input Givens Manually \n\n");printf("Enter your choice: ");
        c=getchar();
        if(c=='f' || c=='F')
        {
                printf("\nEnter name of input file: \n");
                getchar();
                gets(in);
                fp3=fopen(in,"r");
        if(fp3==NULL)
        {

                perror("Operation Failed.");
                return 1;
        }
                printf("\nThe file contains: \n");
        for(i=0;i<9;i++)
        {
                if(i%3==0)
                        printf("\n");
                printf("\n");
                for(j=0;j<9;j++)
                {
                        fscanf(fp3,"%d",*(un+i)+j);
                        sol[i][j]=un[i][j];
                        if(j%3==0)
                                printf("  ");
                        if(sol[i][j]>0 && sol[i][j]<10)
                                printf("%d ",sol[i][j]);
                        else printf("0 ");
                }
        }
                printf("\n");
                system("PAUSE");
                close(fp3);
        }
        else if(c=='m' || c=='M')
        {
                printf("\nEnter the Givens (0 for Blank Square): \n");
                for(i=0;i<9;i++)
                        for(j=0;j<9;j++)
                {
                        scanf("%d",*(un+i)+j);
                        sol[i][j]=un[i][j];
                }
        }
        else
```

```
{
        printf("\nGivens are the non-empty cells given in a Sudoku.");
        printf("\nYou can Enter the Givens either Manually or from a File.\n");
        printf("In both cases 0 represents a Blank cell.\n");
        printf("The Program will Restart Now.\n");
        getchar();
        system("PAUSE");
        system("cls");
        goto inception;
}
analyser();
solver(sol,0,0);
if(f==1)
{
        printf("\nThe Only Solution is:");
        for(i=0;i<9;i++)
        {
                if(i%3==0)
                        printf("\n");
                printf("\n");
                for(j=0;j<9;j++)
                {
                if(j%3==0)
                                printf(" ");
                printf("%d ",sol[i][j]);
                }
        }
        printf("\n");
}
else if(f>1)
{
        printf("\nThere are many Solutions. One solution is:");
        for(i=0;i<9;i++)
        {
                if(i%3==0)
                        printf("\n");
                printf("\n");
                for(j=0;j<9;j++)
                {
                if(j%3==0)
                                printf(" ");
                printf("%d ",sol[i][j]);
                }
        }
        printf("\n");
}
else
{
        printf("\n\nSorry! No Solution exists! \n");
        printf("\nYou have input the follwing givens(New Window):\n\n");
}

fp2=fopen("image1618.bmp","wb");
fp1=fopen("header1","rb");
if(fp1==NULL || fp2==NULL)
{
        perror("Image Operation Failed.\n");
}
```

```
        else
        {
        for(i=0;i<c1;i++)
        {
                fscanf(fp1,"%c",&p);
                fprintf(fp2,"%c",p);
        }
        close(fp1);
        imager(im);
        fwrite(im, sizeof(char),(s*s*3)*sizeof(char), fp2);
        close(fp2);

        if(f>1)
        printf("\nSee New Window(Blue->Given,Maroon->Program Input):\n\n");
        system("PAUSE");
        ShellExecute ( 0,"open", "image1618.bmp", NULL,NULL,SW_NORMAL);
        }
        free(im);

        if(f>0)
        {
                printf("\n\nWant to save it to file? (1/0)\n");
        scanf("%d",&k);
        if(k)
        {
                printf("Enter name of output file: \n");
                getchar();
                gets(out);
                fp3=fopen(out,"w");
                if(fp3==NULL)
                {
                        perror("Operation Failed.");
                        return 1;
                }
                for(i=0;i<9;i++)
                {
                        for(j=0;j<9;j++)
                                fprintf(fp3,"%d ",sol[i][j]);
                        fprintf(fp3,"\n");
                }
                close(fp3);
        }
        }
        printf("\nThe Program will Exit Now.\n");
        printf("\nThank You for using \"Backtracking Sudoku Solver\"!");

        return 0;
}
```