

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <ctype.h>
4 #define MAX 15
5
6 int main(void){
7
8     int mat[MAX][MAX],
9         n, x, y, vx, vy,
10        lim, limx, limy,
11        sumfil[MAX], sumcol[MAX], sumdia[2],
12        aux, flag;
13
14     char opt;
15
16     do{
17         puts("=====");
18         puts("      CUADRADO MAGICO      ");
19         puts("=====");
20
21         do{
22             printf("\nIntroduce el valor de N (max. 15 e impar): ");
23             scanf("%d", &n);
24         }while(n < 1 || n > MAX || n % 2 == 0);
25
26         fflush(stdin); // Limpiar buffer Windows
27         fpurge(stdin); // Limpiar buffer MacOS
28
29         for(int f = 0; f < n; f++){
30             for(int c = 0; c < n; c++){
31                 mat[f][c] = -1;
32             }
33         }
34
35         /* Establecemos posiciones y limites de la matriz */
36         x = n / 2;
37         y = 0;
38         lim = n * n;
39         limx = n - 1;
40         limy = n - 1;
41
42         for(int cont = 1; cont ≤ lim; cont++){
43             mat[y][x] = cont;
44             vx = x;
45             vy = y;
46             x++;
47
48             if(x > limx)
49                 x = 0;
50
51             y--;
52
53             if(y < 0)
54                 y = limy;
55
56             if(mat[y][x] > -1){
57                 x = vx;
58                 y = vy + 1;
59             }
60         }
```

```
61
62  /* SUMA COLUMNAS Y COMPROBACION */
63  for(int f = 0; f < n; f++){
64      aux = 0;
65      for(int c = 0; c < n; c++){
66          aux += mat[f][c];
67      }
68      sumcol[f] = aux;
69  }
70
71  for(int f = 1; f < n; f++){
72      if(sumcol[0] == sumcol[f]){
73          flag = 1;
74      } else {
75          flag = 0;
76      }
77  }
78
79  /* SUMA FILAS Y COMPROBACION */
80  for(int c = 0; c < n; c++){
81      aux = 0;
82      for(int f = 0; f < n; f++){
83          aux += mat[f][c];
84      }
85      sumfil[c] = aux;
86  }
87
88  for(int c = 1; c < n; c++){
89      if(sumfil[0] == sumfil[c]){
90          flag = 1;
91      } else {
92          flag = 0;
93      }
94  }
95
96  /* SUMA DIAGONALES Y COMPROBACION */
97  for(int i = 0; i < n; i++){
98      sumdia[0] += mat[i][i];
99  }
100
101  for(int i = 0; i < n; i++){
102      sumdia[1] += mat[i][n-i-1];
103  }
104
105  if(sumdia[0] == sumdia[1]){
106      flag = 1;
107  } else {
108      flag = 0;
109  }
110
111  /* RESULTADOS */
112  printf("\nCUADRADO MAGICO %dx%d\n", n, n);
113  for(int f = 0; f < n; f++){
114      for(int c = 0; c < n; c++){
115          printf("%5d", mat[f][c]);
116      }
117      printf("\n");
118  }
119
120  printf("\nSuma de sus diagonales: ");
```

```
121     for(int i = 0; i < 2; i++)
122         printf("%5d", sumdia[i]);
123
124     printf("\nSuma de sus    columnas: ");
125     for(int c = 0; c < n; c++)
126         printf("%5d", sumcol[c]);
127
128     printf("\nSuma de sus        filas: ");
129     for(int f = 0; f < n; f++)
130         printf("%5d", sumfil[f]);
131
132     if(flag == 1)
133         printf("\n\nEl cuadrado es magico.\n");
134     else
135         printf("\n\nEl cuadrado NO es magico.\n");
136
137     do{
138         printf("\nOtra ejecucion (S/N): ");
139         opt = toupper(getchar());
140     }while(opt != 'S' && opt != 'N');
141
142     }while(opt != 'N');
143
144     printf("\n");
145     system("pause");
146     return 0;
147 }
148
```