

```

1  #include <stdlib.h>
2  #include <math.h>
3
4
5  int det();
6  int main(void)
7  {
8      int i,j;
9      int A[3][3]={ {1,-3,2}, {5,6,-1}, {4,-1,3} };
10     int B[] = { {-3}, {13}, {8} };
11
12     int tras[3][3], adj[3][3];          // matriz adjunta de la traspuesta y la
13     float invA[3][3], X[3];            // matriz inversa y la de resultados
14
15
16     // ===== Calcula la Matriz Traspuesta
17     for(i=0; i<3;i++)
18     {
19         for(j=0;j<3;j++)
20         {
21             tras[i][j] = A[j][i];
22         }
23     }
24     // =====
25
26     // ===== Calcula la matriz adjunta de la traspuesta
27
28     adj[0][0] = tras[1][1]*tras[2][2] - tras[2][1]*tras[1][2];
29     adj[0][1] = -tras[0][1]*tras[2][2] - tras[2][1]*tras[0][2];
30     adj[0][2] = tras[0][1]*tras[1][2] - tras[1][1]*tras[0][2];
31
32     adj[1][0] = -tras[1][0]*tras[2][2] - tras[2][0]*tras[1][2];
33     adj[1][1] = tras[0][0]*tras[2][2] - tras[2][0]*tras[0][2];
34     adj[1][2] = -tras[0][0]*tras[1][2] - tras[1][0]*tras[0][2];
35
36     adj[2][0] = tras[1][0]*tras[2][1] - tras[2][0]*tras[1][1];
37     adj[2][1] = -tras[0][0]*tras[2][1] - tras[2][0]*tras[0][1];
38     adj[2][2] = tras[0][0]*tras[1][1] - tras[1][0]*tras[0][1];
39
40     // =====
41
42
43     // ===== Inversa de A =====
44     double a = pow(det(A),-1);
45     printf("1/det: %f\n", a);
46     for(i=0; i<3;i++)
47     {
48         for(j=0;j<3;j++)
49         {
50             invA[i][j] = a*adj[i][j];
51         }
52     }
53     // =====
54     // ===== Calcula Matriz Resultados =====
55     float r=0;
56     for(i=0; i<3;i++)
57     {
58         for(j=0;j<3;j++)
59         {
60             r += invA[i][j] *(float) B[j];
61             printf("%f\n",r);
62         }
63         X[i]=r;
64         r=0;
65     }
66     // =====
67     printf("\B = \n");
68     for(i=0; i<3;i++)
69     {
70         printf("[%d]\n",B[i]);
71     }

```

```

72     printf("A = \n");
73     mostrar(A);
74
75     printf("\nDeterminante de A: %d\n", det(A));
76
77     printf("\nTraspuesta de A = \n");
78     mostrar(tras);
79
80     printf("\nAdjunta de A Traspuesta = \n");
81     mostrar(adj);
82
83     printf("\nInversa de A = \n");
84     for(i=0; i<3;i++)
85     {
86         for(j=0;j<3;j++)
87         {
88             printf("[%1.3f]\t", invA[i][j]);
89         }
90         printf("\n");
91     }
92
93     printf("\nResultado  = \n");
94     for(i=0; i<3;i++)
95     {
96         printf("[%1.2f]\n",X[i]);
97     }
98     return 0;
99 }
100
101 int det(int A[3][3]) // Determinante de Matriz 3x3
102 {
103     return A[0][0]*( A[1][1]*A[2][2] - A[2][1]*A[1][2] ) - A[0][1]*( A[1][0]*A[2][2]
104     - A[2][0]*A[1][2] ) + A[0][2]*( A[1][0]*A[2][1] - A[2][0]*A[1][1] );
105 }
106
107 void mostrar(int A[3][3])
108 {
109     int i, j;
110
111     for(i=0; i<3;i++)
112     {
113         for(j=0;j<3;j++)
114         {
115             printf("[%d]\t", A[i][j]);
116         }
117         printf("\n");
118     }
119 }
120

```