

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

Problema – mapeamento

- a) Cálculo do menor “custo” (soma menor)
- b) mostrar o caminho

Partida: [0][0]

chegada: última linha, última col.

Movimentos possíveis:



OBS: importante definir adequadamente os parâmetros da função de modo a poder definir submatrizes (limite superior, esquerdo, inferior, direito, etc.)

```
#include <iostream>
using namespace std;
```

```
int m[7][7] = { { 4, 7, 5, 0, 3, 8, 13},
                 { 3, 5, 2, 1, 6, 1, 4},
                 { 1, 3, 9, 11, 9, 2, 3},
                 {17, 3, 8, 4, 1, 8, 14},
                 { 2, 0, 9, 5, 11, 0, 4},
                 { 6, 4, 0, 2, 1, 10, 3},
                 {7, 13, 2, 0, 7, 1, 2} };
```



ben

Menor custo: 30

Caminho:

0 0
1 0
2 0
2 1
3 1
4 1
5 1
5 2
5 3
6 3
6 4
6 5
6 6

benedito@Benedito-Aspire3:~/C\$

```
#include <iostream>
using namespace std;
int m[7][7] = { { 4, 7, 5, 0, 3, 8,13},
                { 3, 5, 2, 1, 6, 1, 4},
                { 1, 3, 9,11, 9, 2, 3},
                {17, 3, 8, 4, 1, 8,14},
                { 2, 0, 9, 5,11, 0, 4},
                { 6, 4, 0, 2, 1,10, 3},
                {7, 13, 2, 0, 7, 1, 2} };

int cam_min_custo(int mat[7][7], int sup, int esq, int inf, int dir)
{

}

int main ()
{
    int menor_cam;

    menor_cam = cam_min_custo(m,0,0,6,6);
    // ou menor_cam = cam_min_custo(m,3,3,6,6) etc. para testar submatrizes

    cout<< menor_cam << endl;

}
```