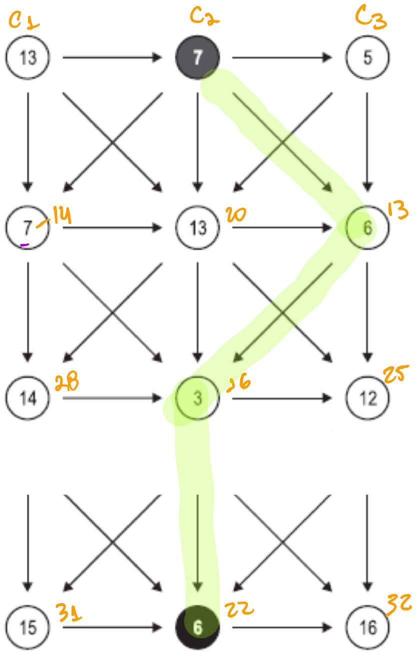


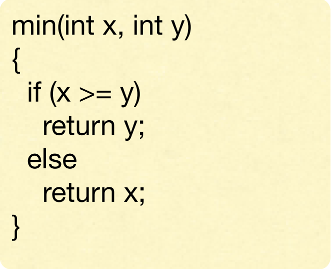
TRI-GRAPH

$x = \min(c_1, c_2) + x$

$y = \min\Big(\min(c_1, x), \min(c_2, c_3)\Big) + y$

$z = \min\Big(\min(c_2, c_3), y\Big) + z$

- Fazer comparações com a lista de adjacências



```
min(int x, int y)
{
  if (x >= y)
    return y;
  else
    return x;
}
```

```
while(1)
  {
    scanf("%d", &n);

    if(n == 0)
      break;

    int grafo[n][3];

    for (i = 0; i < n; ++i)
        scanf("%d %d %d",
             &grafo[i][0], &grafo[i][1], &grafo[i][2]);

    grafo[0][0] = 999999;
    grafo[0][2] += grafo[0][1];

    int custo[n][3];

    for (i = 0; i < 3; ++i)                    3
      custo[0][i] = grafo[0][i];

    for (i = 1; i < n; ++i)                    n - 1
      {
        for (j = 0; j < 3; ++j)                3
          {

            if(j == 0)
              custo[i][j] = min(custo[i-1][j], custo[i-1][j+1]) + grafo[i][j];
                   x                  c1              c2                x
            if(j == 1)
              custo[i][j] = min(         c1              x
                   y              min(custo[i-1][j-1], custo[i][j-1]),
                                  min(custo[i-1][j], custo[i-1][j+1]))+grafo[i][j];
                                          c2              c3                y
            if(j == 2)
              custo[i][j] = min(          c3              c2
                   z              min(custo[i][j-1], custo[i-1][j-1]),
                                  custo[i-1][j]) + grafo[i][j];
                                        y               z
          }
      }
    ++t;
```

i0	i1	i2
~~13~~ 999999	7	~~5~~ 12
7	13	6
14	3	12
15	6	16

9999	7	12
7	13	6
14	3	12
15	6	16

$(n-1)(3)+3$

$3n - 3 + 3$

$\boxed{3n}$

COMPLEXIDADE

$O(n)$

A CADA LINHA

```
#include <stdio.h>
#define DEBUG if(1)

int
min(int x, int y)
{
  if (x >= y)
    return y;
  else
    return x;
}

int
main()
{
  int n, i, j, t;
  t=0;

  while(1)
    {
      scanf("%d", &n);

      if(n == 0)
        break;

      int grafo[n][3];

      for (i = 0; i < n; ++i)
          scanf("%d %d %d",
                &grafo[i][0], &grafo[i][1], &grafo[i][2]);

      grafo[0][0] = 999999;
      grafo[0][2] += grafo[0][1];

      int custo[n][3];

      for (i = 0; i < 3; ++i)
        custo[0][i] = grafo[0][i];

      for (i = 1; i < n; ++i)
        {
          for (j = 0; j < 3; ++j)
            {

              if(j == 0)
                custo[i][j] = min(custo[i-1][j], custo[i-1][j+1]) + grafo[i][j];
```

```
int
main()
{
  int n, i, j, t;
  t=0;

  while(1)
    {
      scanf("%d", &n);

      if(n == 0)
        break;

      int grafo[n][3];

      for (i = 0; i < n; ++i)
          scanf("%d %d %d",
                &grafo[i][0], &grafo[i][1], &grafo[i][2]);

      grafo[0][0] = 999999;
      grafo[0][2] += grafo[0][1];

      int custo[n][3];

      for (i = 0; i < 3; ++i)
        custo[0][i] = grafo[0][i];

      for (i = 1; i < n; ++i)
        {
          for (j = 0; j < 3; ++j)
            {

              if(j == 0)
                custo[i][j] = min(custo[i-1][j], custo[i-1][j+1]) + grafo[i][j];

              if(j == 1)
                custo[i][j] = min(
                                   min(custo[i-1][j-1], custo[i][j-1]),
                                   min(custo[i-1][j], custo[i-1][j+1]))+grafo[i][j];
              if(j == 2)
                custo[i][j] = min(
                                   min(custo[i][j-1], custo[i-1][j-1]),
                                   custo[i-1][j]) + grafo[i][j];
      }

    }
   ++t;


   printf("%d. %d\n",t, custo[n-1][1]);
  }



 return 0;
}
```