

Código em python:

```
def dijkstra(grid, inicio, objetivo, num_linhas, num_colunas):  
    while pq:  
        #pega o vertice com menor distancia  
        dist_atual, vertice_atual = heapq.heappop(pq)  
  
        #pula oq ja foi visitado  
        if vertice_atual in visitados:  
            continue  
  
        #marca como visitado e adc a S  
        visitados.add(vertice_atual)  
  
        #para se chegar no objetivo  
        if vertice_atual == objetivo:  
            break  
  
        #explorar vizinhos  
        i, j = vertice_atual  
  
        for di, dj in direcoes:  
            ni, nj = i + di, j + dj  
  
            # Verifica os limites do grid  
            if 0 <= ni < num_linhas and 0 <= nj < num_colunas:  
                vizinho = (ni, nj)  
  
                #pula oque ja foi visitado  
                if vizinho in visitados:  
                    continue  
  
                #custo da celula vizinha  
                custo = obter_custo(grid[ni][nj])  
  
                #verifica se é um obstacula para pular  
                if custo == 0: continue
```

Pseudocódigo:

## Algoritmo

início

$d_{11} \leftarrow 0$ ;  $d_{1i} \leftarrow \infty \forall i \in V - \{1\}$ ; < distância origem-origem zero; distâncias a partir da origem infinitas >

$A \leftarrow V$ ;  $F \leftarrow \emptyset$ ; *anterior* ( $i$ )  $\leftarrow 0 \forall i$ ;

enquanto  $A \neq \emptyset$  fazer

início

$r \leftarrow v \in V \mid d_{1r} = \min_{i \in A} [d_{1i}]$  < acha o vértice mais próximo da origem >

$F \leftarrow F \cup \{r\}$ ;  $A \leftarrow A - \{r\}$ ; < o vértice  $r$  sai de **Aberto** para **Fechado** >

$S \leftarrow A \cap N^+(r)$  <  $S$  são os **sucessores** de  $r$  **ainda abertos** >

para todo  $i \in S$  fazer

início

$p \leftarrow \min [d_{1i}^{k-1}, (d_{1r} + v_{ri})]$  < compara o valor anterior com a nova soma >

se  $p < d_{1i}^{k-1}$  então

início

$d_{1i}^k \leftarrow p$ ; *anterior* ( $i$ )  $\leftarrow r$ ; < ganhou a nova distância ! >

fim;

fim;

fim;

fim.