Misal ada Maze seperti ini, Nah kita Disuruh nyari langkah minimal dari Source ke Destinasi.  
Caranya Bagaimana? Yep, BFS (Breadth First Search).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | . | . | . | . | . | . | . |
| D | . | . | . | # | . | # | . |
| # | # | . | . | . | # | . | . |
| . | . | # | # | . | # | . | . |
| # | . | . | . | . | # | . | . |
| . | # | # | . | . | # | . | . |
| . | # | . | # | . | . | S | . |
| . | . | . | . | . | # | . | . |

Note :

S = Posisi Awal

D = Posisi Akhir

. = Jalan Bebas

# = Jalan Buntu

Hasilnya nanti kalau BFS udah dijalankan kira-kira ginilah.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| # | 11 | 10 | 9 | 10 | 9 | 8 | 7 |
| 11 | 10 | 9 | 8 | # | 10 | # | 6 |
| # | # | 8 | 7 | 6 | # | 4 | 5 |
| 9 | 8 | # | # | 5 | # | 3 | 4 |
| # | 7 | 6 | 5 | 4 | # | 2 | 3 |
| 9 | # | # | 4 | 3 | # | 1 | 2 |
| 8 | # | 6 | # | 2 | 1 | **S** | 1 |
| 7 | 6 | 5 | 4 | 3 | # | 1 | 2 |

Sebenarnya intinya BFS (Breadth First Sirch) itu mulai dari induknya dulu (Sourcenya). Nah dari posisi **S** , Kalau 1 langkah bisa kemana aja? Nah, disimpanlah posisi-posisi tadi yang memungkinkan. Kenapa kita simpan? Karena kita butuh posisi ini untuk menentukan 2 langkah selanjutnya dari Source bisa kemana aja. Nanti gitu trus, sampai nyampe ke Destinasi Baru dia berhenti.