



Given:

$h(S) = 4.0$	$h(H) = 4.0$
$h(A) = 2.0$	$h(G) = 0.0$
$h(C) = 4.0$	$h(E) = 2.0$
$h(K) = 5.0$	$h(I) = 5.0$
$h(B) = 3.0$	$h(F) = 1.0$
$h(D) = 3.0$	$h(J) = 6.0$
$h(L) = 6.0$	

Uniform Cost Search for Beginner (Solve it in lexicographical order)

- Tentukan root node
 - Root node adalah S
 - Masukkan root node ke Priority Queue
 - Priority Queue = {S}
- Pilih node pertama di Priority Queue
 - Node pertama adalah S
 - Hilangkan S dari Priority Queue
 - Priority Queue = {}
- Goal Test: Cek apakah node pertama di Priority Queue yang dipilih merupakan goal?
 - Kalau goal, solusi ditemukan
 - Kalau tidak, lanjut ke langkah selanjutnya
- Lalu expand node yang dipilih (turunkan anaknya)
 - Children node dari node S adalah A, C, K
 - Hitung path cost $g(n)$ dari root node (S) ke setiap children node
 - Contoh path cost $g(n)$ dari S ke E adalah 3 yang merupakan penjumlahan cost dari S – C (1), C – D (1), D – E (1)
 - Masukkan children node ke Priority Queue
 - Urutkan sesuai nilai path cost $g(n)$ yang paling kecil
 - Tandai orang tua (parent) dari setiap node
 - Contoh A^S melambangkan S adalah orang tua dari A
 - Priority Queue = { C^S , A^S , K^S } // 1, 2, 2

➤ Yang berarti C^S memiliki cost $g(n)$ 1, A^S memiliki cost $g(n)$ 2 dan seterusnya.

- Lakukan repetisi dari langkah kedua sampai goal ditemukan di langkah ketiga

Hasil lengkap dari contoh di atas:

- Priority Queue = {S}
- Visit S, Priority Queue = $\{C^S, A^S, K^S\}$ // 1, 2, 2
- Visit C^S , Priority Queue = $\{A^S, D^C, K^S\}$ // 2, 2, 2
- Visit A^S , Priority Queue = $\{D^C, K^S, B^A\}$ // 2, 2, 4
- Visit D^C , Priority Queue = $\{K^S, E^D, B^A\}$ // 2, 3, 4
- Visit K^S , Priority Queue = $\{E^D, L^K, B^A\}$ // 3, 3, 4
- Visit E^D , Priority Queue = $\{L^K, B^A, F^E\}$ // 3, 4, 4
- Visit L^K , Priority Queue = $\{B^A, F^E\}$ // 4, 4
- Visit B^A , Priority Queue = $\{F^E, H^B, G^B\}$ // 4, 6, 7
- Visit F^E , Priority Queue = $\{G^B, H^B, G^B\}$ // 5, 6, 7
- Visit G^B , Priority Queue = $\{H^B, G^B\}$ // 6, 7

Path = S, C, D, E, F, G

Cost $g(n)$ = 5

Visited node = S, C, A, D, K, E, L, B, F, G