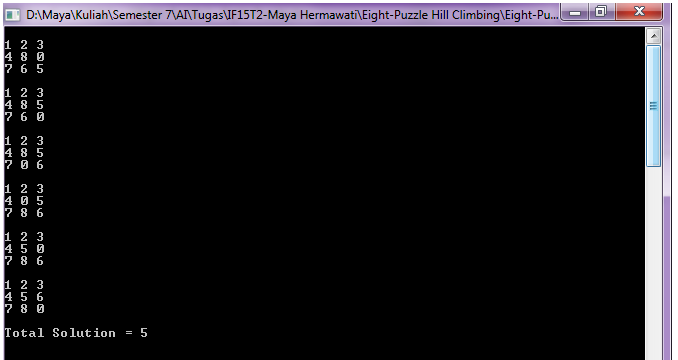
**Program implementation with Hill Climbing strategy**

**for Eight-Puzzle Problem**



**Analisis Algoritma**

Pada 8-puzzle, satu kali proses evaluasi menggunakan Hill Climbing hanya akan melibatkan maksimal 4 *state* untuk kondisi *initial state*, dan maksimal 3 *state* untuk kondisi *state* selain *initial state*. Sehingga *state space* untuk algoritma ini dapat dikatakan relatif sangat kecil. Di bawah ini dijelaskan bagaimana algortima Hill Climbing dijalankan untuk suatu kasus *state* tertentu pada 8-puzzle :

Initial state

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 8 | 0 |
| 7 | 6 | 5 |

Goal state

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 0 |

Untuk menerapkan algoritma Hill Climbing pada 8-puzzle problem, maka harus ditentukan nilai heuristic untuk setiap state yang mungkin. Nilai heuristic ini dapat dicari salah satunya dengan menggunakan cara Manhattan Distance. Manhattan Distance merupakan penjumlahan jarak masing-masing kotak puzzle terhadap posisinya yang benar pada kondisi goal.

Perhitungan Manhattan Distance untuk initial state

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 8 | 0 |
| 7 | 6 | 5 |

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 0 |

h(4) = |0-0| + |0-1| =1

h(8) = |2-2| + |0-2| =2

h(2) = |0-1| + |1-0| =2

h(3) = |1-2| + |1-0| =2

h(1) = |2-0| + |1-0| =3

h(7) = |0-0| + |2-2| =0

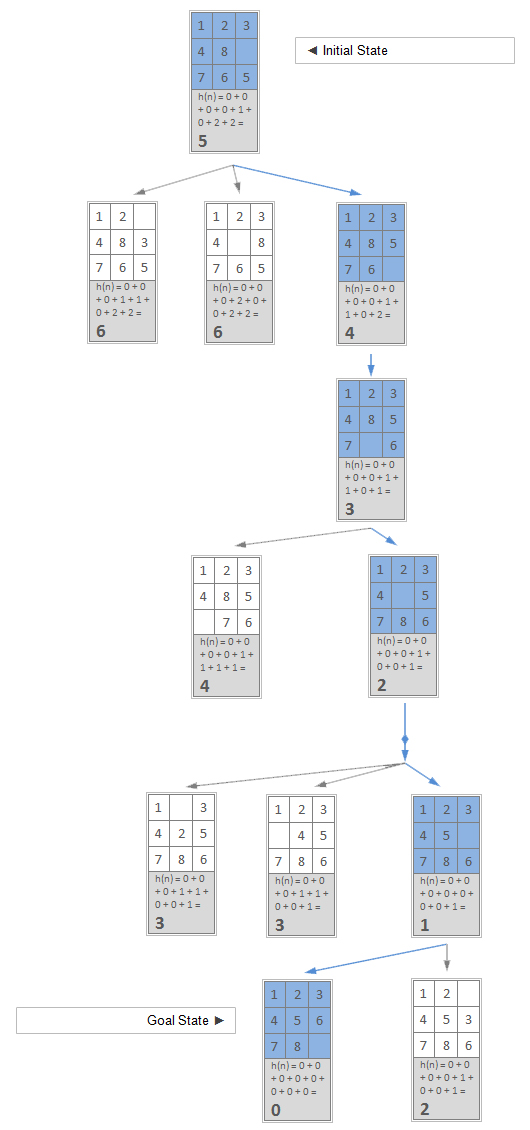
h(6) = |1-2| + |2-1| =2

h(5) = |2-1| + |1-1| =1

h(n) = h(4)+h(8)+h(2)+h(3)+h(1)+h(7)+h(6)+h(5) = 14

Jadi nilai heuristic untuk initial state adalah 14

Perhitungan nilai heuristic ini dilakukan untuk semua kemungkinan state berikutnya. Pada algoritma hill climbing, proses evalusi yang dilakukan akan selalu mengambil nilai *heuristic* yang paling kecil.



Evaluasi pertama dilakukan pada initial state yang memiliki nilai heuristic 5, karena state ini bukan merupakan goal state, maka initial state dijadikan current state untuk dibandingkan dengan state pada solusi selanjutnya. Setelah semua solusi pada kedalaman berikutnya dibangkitkan, maka bandingkan setiap nilai heuristic dengan current state, jika nilai heuristicnya lebih kecil daripada nilai heuristic current state maka pilih state tersebut sebagai solusi. Pada kasus ini, karena 6 tidak lebih kecil dari 5 maka state yang dipilih sebagai solusi adalah yang ketiga dengan nilai heuristic 4, dan begitu selanjutnya hingga didapatkan goal state.

Pada kondisi goal state dengan nilai heuristic 0 (nol), sebab semua kotak sudah berada pada posisinya masing-masing (jarak dengan posisi benar=0).