# Bab 1 : Deskripsi Singkat

Program ini merupakan program untuk menyelesaikan permainan 15-Puzzle, yaitu permainan yang terdiri dari 15 kotak yang berisi angka dari 1 sampai 15 yang berada pada kotak 4x4. Permainan ini bertujuan untuk menyusun kotak-kotak tersebut agar tersusun berurutan seperti gambar pada di bawah ini.

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
| 13 | 14 | 15 |  |

**Gambar 1.** Susunan akhir persoalan 15-Puzzle

Untuk menyelesaikan permainan tersebut, program ini menggunakan algoritma Branch and Bound.

# Bab 2 : Cara Kerja Algoritma Branch and Bound

Berikut merupakan implementasi Algoritma Branch and Bound pada program ini.

1. Buat sebuah *node root* yang berisi matriks 4x4 yang merupakan kondisi awal puzzle (kotak kosong akan dianggap sebagai kotak 16)
2. Hitung Kurang(i) untuk tiap i dari 1 sampai 16, yaitu banyaknya kotak yang berada “setelah” kotak (kotak berada setelah kotak didefinisikan sebagai keadaan di mana kotak berada di posisi kotak yang seharusnya, kotak berada di posisi kotak yang seharusnya, dan ) hitung , dengan bernilai 0 jika kotak kosong berada pada daerah yang diarsir di bawah ini, dan 1 jika sebaliknya.

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Gambar 2.** Tabel untuk menentukan nilai X

1. Jika bernilai ganjil, maka persoalan tidak bisa diselesaikan, hentikan algoritma. Jika genap, persoalan dapat diselesaikan.
2. Buat sebuah *priority queue* yang berisi node, di mana anggota *queue* yang memiliki cost paling kecil akan diproses lebih dulu.
3. Cari nilai *cost* untuk *node* yang sedang diproses. Nilai *cost* untuk node x dapat dicari menggunakan persamaan berikut

Dengan merupakan panjang lintasan dari root ke node dan merupakan banyaknya kotak yang tidak menempati posisi yang benar.

1. Masukkan *node* ke *priority queue.*
2. Ambil satu *node* dalam *priority queue*.
3. Cek apakah *node* merupakan solusi. Dapat dicek dengan apakah atau apakah jika tidak menyimpan nilai
4. Jika *node* solusi, tetapkan suatu batas B sehingga untuk *node* yang *cost-*nya lebih dari B tidak akan diproses. Bandingkan *cost* *node* ini dengan solusi yang sudah ada sebelumnya jika ada. Jika *cost node* ini lebih kecil, maka *node* ini dapat menjadi solusi sementara. Jika sebelumnya belum ada solusi, maka *node* ini dapat menjadi solusi sementara.
5. Lakukan *expand* pada *node* yan sedang diproses, yaitu mencari langkah yang mungkin dari keadaan *puzzle* pada *node* tersebut, lalu keadaan *puzzle* setelah melakukan satu langkah tersebut akan menjadi *child* dari *node* yang sedang diproses. Kemudian, buang *node* dari *priority queue.*
6. Lakukan langkah 5 dan 6 untuk semua *child* yang baru saja dibuat, lalu kembali ke langkah 7 sampai tidak ada *node* yang dapat diambil dari *priority queue*.
7. *Node* solusi pada program ini hanya menyimpan satu langkah sebelum mencapai *node* tersebut, sehingga untuk mendapatkan semua solusi, lakukan *tracking* dari *node* solusi ke *node* akar dan simpan tiap-tiap langkah dalam suatu *list*. Kemudian, *reverse list* tersebut untuk mendapatkan langkah dari keadaan awal *puzzle* sampai *puzzle* selesai.

# Bab 3 : Kode Program

# Bab 4 : Masukan dan Luaran Program

Masukan program ini berupa *file* berekstensi *.txt*. Perlu diperhatikan bahwa pada program ini, arah UP, DOWN, LEFT, dan RIGHT mengacu pada pergerakan kotak kosong (jika kotak kosong ke atas, langkahnya ialah UP dan seterusnya)

1. *File* *solvable\_1.txt*

*File solvable\_1.txt* berisi teks berikut.

|  |
| --- |
| 1 2 3 4  5 6 16 8  9 10 7 11  13 14 15 12 |

Dengan masukan *file* tersebut, keluarannya ialah seperti berikut.



**Gambar 3.** Hasil program ketika menjalankan *solvable\_1.txt*

*Textbox* pada bagian bawah GUI berisi teks berikut.

|  |
| --- |
| 1. Initial Puzzle Matrix:  1 2 3 4  5 6 16 8  9 10 7 11  13 14 15 12  2. Kurang(i) values:  Kurang(1) = 0  Kurang(2) = 0  Kurang(3) = 0  Kurang(4) = 0  Kurang(5) = 0  Kurang(6) = 0  Kurang(7) = 0  Kurang(8) = 1  Kurang(9) = 1  Kurang(10) = 1  Kurang(11) = 0  Kurang(12) = 0  Kurang(13) = 1  Kurang(14) = 1  Kurang(15) = 1  Kurang(16) = 9  3. Sum of Kurang(i) + X = 16  Solution Steps:  DOWN  1 2 3 4  5 6 7 8  9 10 16 11  13 14 15 12  RIGHT  1 2 3 4  5 6 7 8  9 10 11 16  13 14 15 12  DOWN  1 2 3 4  5 6 7 8  9 10 11 12  13 14 15 16  Steps : 3  Time elapsed to generate solution: 9 ms  Number of generated nodes = 10 |

1. *File solvable\_2.txt*

*File solvable\_2.txt* berisi teks berikut.

|  |
| --- |
| 16 1 2 3  6 7 8 4  5 9 10 11  13 14 15 12 |

Dengan masukan *file* tersebut, keluaran program ialah seperti berikut.



**Gambar 4.** Hasil program ketika menjalankan *solvable\_2.txt*

*Textbox* pada bagian bawah GUI berisi teks berikut.

|  |
| --- |
| 1. Initial Puzzle Matrix:  16 1 2 3  6 7 8 4  5 9 10 11  13 14 15 12  2. Kurang(i) values:  Kurang(1) = 0  Kurang(2) = 0  Kurang(3) = 0  Kurang(4) = 0  Kurang(5) = 0  Kurang(6) = 2  Kurang(7) = 2  Kurang(8) = 2  Kurang(9) = 0  Kurang(10) = 0  Kurang(11) = 0  Kurang(12) = 0  Kurang(13) = 1  Kurang(14) = 1  Kurang(15) = 1  Kurang(16) = 15  3. Sum of Kurang(i) + X = 24  Solution Steps:  RIGHT  1 16 2 3  6 7 8 4  5 9 10 11  13 14 15 12  RIGHT  1 2 16 3  6 7 8 4  5 9 10 11  13 14 15 12  RIGHT  1 2 3 16  6 7 8 4  5 9 10 11  13 14 15 12  DOWN  1 2 3 4  6 7 8 16  5 9 10 11  13 14 15 12  LEFT  1 2 3 4  6 7 16 8  5 9 10 11  13 14 15 12  LEFT  1 2 3 4  6 16 7 8  5 9 10 11  13 14 15 12  LEFT  1 2 3 4  16 6 7 8  5 9 10 11  13 14 15 12  DOWN  1 2 3 4  5 6 7 8  16 9 10 11  13 14 15 12  RIGHT  1 2 3 4  5 6 7 8  9 16 10 11  13 14 15 12  RIGHT  1 2 3 4  5 6 7 8  9 10 16 11  13 14 15 12  RIGHT  1 2 3 4  5 6 7 8  9 10 11 16  13 14 15 12  DOWN  1 2 3 4  5 6 7 8  9 10 11 12  13 14 15 16  Steps : 12  Time elapsed to generate solution: 91 ms  Number of generated nodes = 28 |

1. *File solvable\_3.txt*

*File solvable\_3.txt* berisi teks berikut.

|  |
| --- |
| 1 6 2 4  5 16 3 8  9 7 15 11  13 14 10 12 |

Dengan masukan *file* tersebut, keluaran program ialah seperti berikut.



**Gambar 5.** Hasil program ketika menjalankan *solvable\_3.txt*

*Textbox* pada bagian bawah GUI berisi teks berikut.

|  |
| --- |
| 1. Initial Puzzle Matrix:  1 6 2 4  5 16 3 8  9 7 15 11  13 14 10 12  2. Kurang(i) values:  Kurang(1) = 0  Kurang(2) = 0  Kurang(3) = 0  Kurang(4) = 1  Kurang(5) = 1  Kurang(6) = 4  Kurang(7) = 0  Kurang(8) = 1  Kurang(9) = 1  Kurang(10) = 0  Kurang(11) = 1  Kurang(12) = 0  Kurang(13) = 2  Kurang(14) = 2  Kurang(15) = 5  Kurang(16) = 10  3. Sum of Kurang(i) + X = 28  Solution Steps:  LEFT  1 6 2 4  16 5 3 8  9 7 15 11  13 14 10 12  DOWN  1 6 2 4  9 5 3 8  16 7 15 11  13 14 10 12  DOWN  1 6 2 4  9 5 3 8  13 7 15 11  16 14 10 12  RIGHT  1 6 2 4  9 5 3 8  13 7 15 11  14 16 10 12  RIGHT  1 6 2 4  9 5 3 8  13 7 15 11  14 10 16 12  UP  1 6 2 4  9 5 3 8  13 7 16 11  14 10 15 12  LEFT  1 6 2 4  9 5 3 8  13 16 7 11  14 10 15 12  DOWN  1 6 2 4  9 5 3 8  13 10 7 11  14 16 15 12  LEFT  1 6 2 4  9 5 3 8  13 10 7 11  16 14 15 12  UP  1 6 2 4  9 5 3 8  16 10 7 11  13 14 15 12  UP  1 6 2 4  16 5 3 8  9 10 7 11  13 14 15 12  RIGHT  1 6 2 4  5 16 3 8  9 10 7 11  13 14 15 12  UP  1 16 2 4  5 6 3 8  9 10 7 11  13 14 15 12  RIGHT  1 2 16 4  5 6 3 8  9 10 7 11  13 14 15 12  DOWN  1 2 3 4  5 6 16 8  9 10 7 11  13 14 15 12  DOWN  1 2 3 4  5 6 7 8  9 10 16 11  13 14 15 12  RIGHT  1 2 3 4  5 6 7 8  9 10 11 16  13 14 15 12  DOWN  1 2 3 4  5 6 7 8  9 10 11 12  13 14 15 16  Steps : 18  Time elapsed to generate solution: 11520 ms  Number of generated nodes = 9178 |

1. *File unsolvable\_1.txt*

*File unsovable\_1.txt* berisi teks berikut.

|  |
| --- |
| 5 1 3 4  9 2 7 8  10 6 15 11  13 16 14 12 |

Dengan masukan *file* tersebut, keluaran program ialah seperti berikut.



**Gambar 6.** Hasil program ketika menjalankan *unsolvable\_1.txt*

*Textbox* pada bagian bawah GUI berisi teks berikut.

|  |
| --- |
| 1. Initial Puzzle Matrix:  5 1 3 4  9 2 7 8  10 6 15 11  13 16 14 12  2. Kurang(i) values:  Kurang(1) = 0  Kurang(2) = 0  Kurang(3) = 1  Kurang(4) = 1  Kurang(5) = 4  Kurang(6) = 0  Kurang(7) = 1  Kurang(8) = 1  Kurang(9) = 4  Kurang(10) = 1  Kurang(11) = 0  Kurang(12) = 0  Kurang(13) = 1  Kurang(14) = 1  Kurang(15) = 4  Kurang(16) = 2  3. Sum of Kurang(i) + X = 21  Puzzle cannot be solved |

1. *File unsolvable\_2.txt*

*File unsolvable\_2.txt* berisi teks berikut.

|  |
| --- |
| 7 1 2 3  6 16 8 4  5 9 10 11  13 14 15 12 |

Dengan masukan *file* tersebut, keluaran program ialah seperti berikut.



**Gambar 7.** Hasil program ketika menjalankan *unsolvable\_2.txt*

*Textbox* pada bagian bawah GUI berisi teks berikut.

|  |
| --- |
| 1. Initial Puzzle Matrix:  7 1 2 3  6 16 8 4  5 9 10 11  13 14 15 12  2. Kurang(i) values:  Kurang(1) = 0  Kurang(2) = 0  Kurang(3) = 0  Kurang(4) = 0  Kurang(5) = 0  Kurang(6) = 2  Kurang(7) = 6  Kurang(8) = 2  Kurang(9) = 0  Kurang(10) = 0  Kurang(11) = 0  Kurang(12) = 0  Kurang(13) = 1  Kurang(14) = 1  Kurang(15) = 1  Kurang(16) = 10  3. Sum of Kurang(i) + X = 23  Puzzle cannot be solved |