

## **Laporan Tugas Kecil 3**

### **Algoritma Branch and Bound**

**Dibuat untuk memenuhi tugas pada mata kuliah Strategi Algoritma IF2211**

**Semester 2 Tahun 2021/2022**



**Oleh**

**Willy Wilsen – 13520160**

## 1. Algoritma Branch and Bound pada 15-Puzzle

Misalkan diberikan 15-Puzzle sebagai berikut.

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

Dengan susunan akhir yang diinginkan sebagai berikut.

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

### Langkah awal:

Mula-mula tentukan apakah puzzle dapat mencapai solusi dengan menggunakan teorema sebagai berikut.


$$\sum_{i=1}^{16} Kurang(i) + X$$

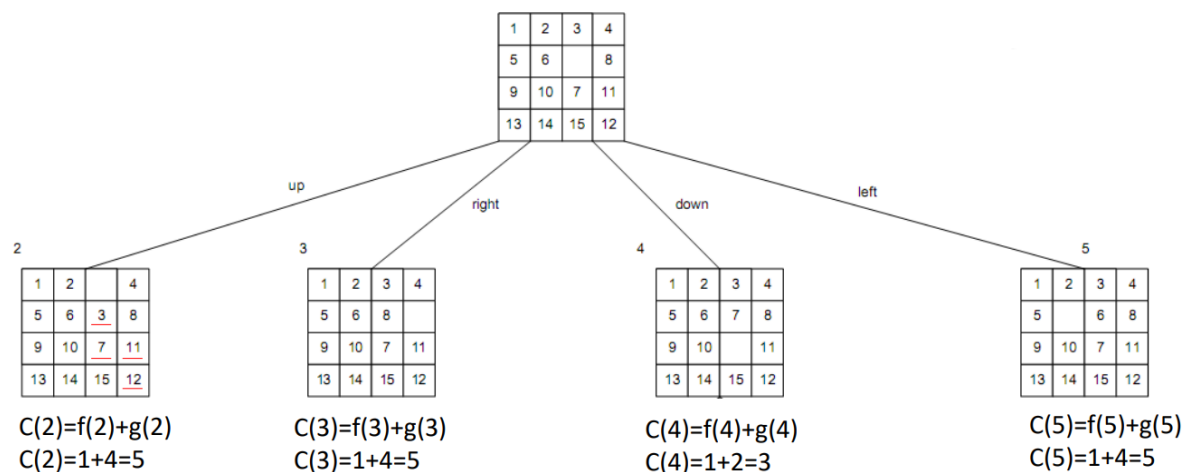
Kurang(i) adalah banyaknya ubin bernomor j sedemikian hingga  $j < i$  dan Posisi(j) > Posisi(i) dengan Posisi(i) adalah posisi ubin bernomor i pada susunan yang diperiksa.  $X = 1$  jika sel kosong pada posisi awal terdapat pada sel yang diarsir seperti gambar diatas. Puzzle dapat mencapai solusi apabila seluruh  $Kurang(i) + X$  bernilai genap. Pada persoalan puzzle di atas didapatkan nilai dari seluruh  $Kurang(i) + X = 15 + 1 = 16$  sehingga dapat mencapai solusi.

## Algoritma Branch and Bound:

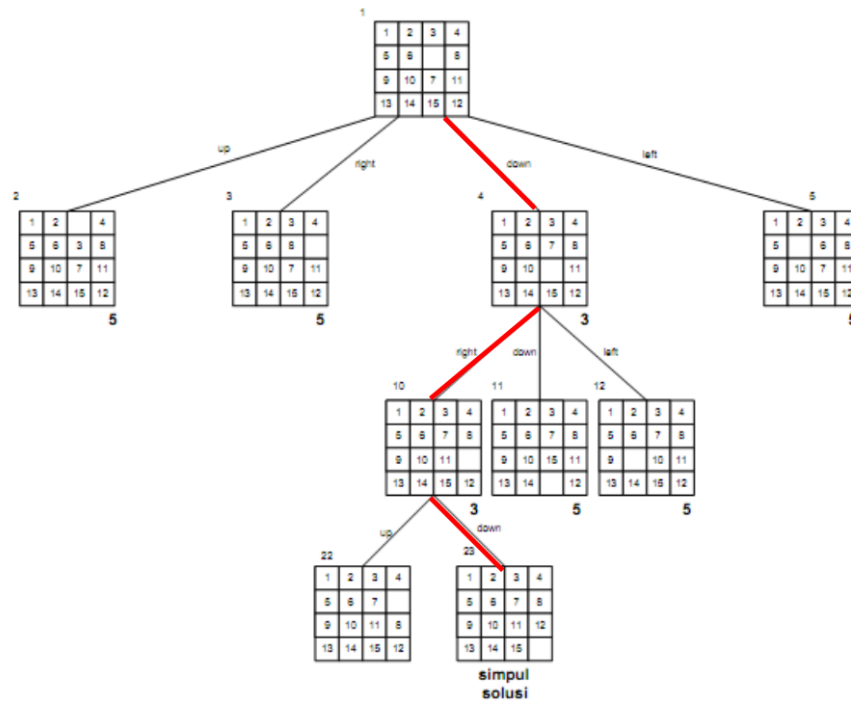
Algoritma Branch and Bound untuk menyelesaikan 15-Puzzle adalah sebagai berikut.

1. Masukkan simpul akar (puzzle awal) ke dalam antrian PriorityQueue Q. Jika simpul akar adalah simpul solusi, maka solusi telah ditemukan dan stop. Jika bukan, maka bangkitkan semua simpul anaknya dalam hal ini yaitu up, right, down, dan left kemudian cost setiap anak dihitung dan semua simpul anak dimasukkan ke dalam Q.
2. Jika Q kosong, maka stop.
3. Jika Q tidak kosong, pilih dari antrian Q simpul i yang mempunyai nilai cost  $c(i)$  paling kecil. Jika terdapat beberapa simpul i yang memenuhi, pilih satu secara sembarang. Cost simpul i pada 15-Puzzle dapat dihitung sebagai berikut.  
$$c(i) = f(i) + g(i)$$
  
 $f(i)$  adalah panjang lintasan dari simpul akar ke i  
 $g(i)$  adalah jumlah ubin tidak kosong yang tidak terdapat pada susunan puzzle solusi
4. Jika simpul i adalah simpul solusi berarti solusi sudah ditemukan dan stop.
5. Jika simpul i bukan simpul solusi, maka bangkitkan semua simpul anaknya.
6. Untuk setiap anak j dari simpul i, hitung  $c(j)$  dan masukan semua simpul anak ke dalam Q
7. Kembali ke langkah 2

Berikut adalah contoh perhitungan cost untuk contoh persoalan puzzle di atas.



Maka, didapatkan urutan pemeriksaan simpul dan pembentukan pohon ruang status dari persoalan 15-Puzzle di atas adalah sebagai berikut.



Simpul-E	Simpul Hidup
1	4,2,3,5
4	10,2,3,5,11,12
10	23,2,3,5,11,12,22
23	Solusi ketemu

## 2. Screenshoot input/output program

Nama File	Input	Output
testcase1.txt	<pre> 15-Puzzle With Branch &amp; Bound Algorithm (With Heuristic Technique)  Made By 13520160 Willy Wilsen  Input your 15-Puzzle filename (.txt): testcase1.txt </pre>	<pre> Nilai untuk setiap Kurang(i): 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  Jumlah total Kurang(i) + X adalah 16  Simpul 1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 12  Simpul 4 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 12  Simpul 6 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 12  Simpul 10 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 12  Execution Time: 0 ms Total simpul yang terbentuk adalah sebanyak 10 simpul </pre>

testcase2.txt	<pre> ----- 15-Puzzle With Branch &amp; Bound Algorithm (With Heuristic Technique) ----- Made By 13520160 Willy Wilsen ----- Input your 15-Puzzle filename (.txt): testcase2.txt </pre>	<pre> Nilai untuk setiap Kurang(i): Kurang(1) = 0 Kurang(2) = 0 Kurang(3) = 2 Kurang(4) = 2 Kurang(5) = 1 Kurang(6) = 4 Kurang(7) = 6 Kurang(8) = 0 Kurang(9) = 0 Kurang(10) = 1 Kurang(11) = 0 Kurang(12) = 2 Kurang(13) = 4 Kurang(14) = 2 Kurang(15) = 0 Kurang(16) = 6  Jumlah total Kurang(i) + X adalah 31  15-Puzzle tidak dapat diselesaikan </pre>
testcase3.txt	<pre> ----- 15-Puzzle With Branch &amp; Bound Algorithm (With Heuristic Technique) ----- Made By 13520160 Willy Wilsen ----- Input your 15-Puzzle filename (.txt): testcase3.txt </pre>	<pre> Nilai untuk setiap Kurang(i): Kurang(1) = 0 Kurang(2) = 1 Kurang(3) = 1 Kurang(4) = 1 Kurang(5) = 0 Kurang(6) = 3 Kurang(7) = 0 Kurang(8) = 0 Kurang(9) = 0 Kurang(10) = 0 Kurang(11) = 0 Kurang(12) = 1 Kurang(13) = 4 Kurang(14) = 1 Kurang(15) = 0 Kurang(16) = 3  Jumlah total Kurang(i) + X adalah 16  Simpul 1  2 3 6 4 1 5 7 8 13 9 10 12 14 11 15  Simpul 101  1 2 3 4 5 6 8 9 10 7 12 13 14 11 15  Simpul 105  1 2 3 4 5 6 7 8 9 10 12 13 14 11 15  Simpul 107  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  Simpul 109  1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  Execution Time: 0 ms Total simpul yang terbentuk adalah sebanyak 110 simpul </pre>

testcase4.txt	<pre> ----- 15-Puzzle With Branch &amp; Bound Algorithm (With Heuristic Technique) ----- Made By 13520160 Willy Wilsen ----- Input your 15-Puzzle filename (.txt): testcase4.txt </pre>	<pre> Nilai untuk setiap Kurang(i): Kurang(1) = 0 Kurang(2) = 1 Kurang(3) = 2 Kurang(4) = 1 Kurang(5) = 0 Kurang(6) = 3 Kurang(7) = 0 Kurang(8) = 1 Kurang(9) = 1 Kurang(10) = 1 Kurang(11) = 0 Kurang(12) = 1 Kurang(13) = 1 Kurang(14) = 1 Kurang(15) = 0 Kurang(16) = 9  Jumlah total Kurang(i) + X adalah 23  15-Puzzle tidak dapat diselesaikan </pre>
testcase5.txt	<pre> ----- 15-Puzzle With Branch &amp; Bound Algorithm (With Heuristic Technique) ----- Made By 13520160 Willy Wilsen ----- Input your 15-Puzzle filename (.txt): testcase5.txt </pre>	<pre> Simpul 4250 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  Simpul 4253 1 2 3 4 5 6 7 8 10 14 15 13 9 12 11  Simpul 4256 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15  Execution Time: 13 ms Total simpul yang terbentuk adalah sebanyak 4259 simpul </pre> <p>Dapat dilihat pada simpul 4250 dan 4253, kedua susunan puzzle berbeda.</p> <p>Hal ini dikarenakan simpul 4253 adalah simpul hidup yang diturunkan dari simpul lain dengan cost yang diproses dahulu kemudian dilanjutkan dengan simpul 4256 yang merupakan solusi.</p>

### 3. Source program

Pada Algoritma Branch and Bound kali ini, dipilih bahasa pemrograman Java dalam implementasinya.

#### 4. Berkas teks contoh persoalan 15-Puzzle

Nama File	Isi
testcase1.txt	1 2 3 4 5 6 0 8 9 10 7 11 13 14 15 12
testcase2.txt	7 3 6 4 1 5 2 8 13 0 10 12 14 9 11 15
testcase3.txt	2 3 6 4 1 5 7 8 13 9 10 12 0 14 11 15
testcase4.txt	3 2 6 4 1 5 0 8 9 10 7 12 13 14 11 15
testcase5.txt	2 3 6 4 1 5 7 8 13 10 12 15 14 0 9 11

#### 5. Alamat github repository kode program

<https://github.com/TubesForLyfe/Tugas-Kecil-3-Strategi-Algoritma>

Poin	Ya	Tidak
1. Program berhasil dikompilasi	v	
2. Program berhasil running	v	
3. Program dapat menerima input dan menuliskan output	v	
4. Luaran sudah benar untuk semua data uji	v	
5. Bonus dibuat	v	