

Implementasi Algoritma *Greedy* dalam Pencarian Solusi *Puzzle* Rush Hour (*Greedy Best-First Search*) dengan Analisis Waktu

Jonathan Levi - 13523132

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jalan Ganesha 10 Bandung

E-mail: jonathanlevi12345678@gmail.com , 13523132@std.stei.itb.ac.id

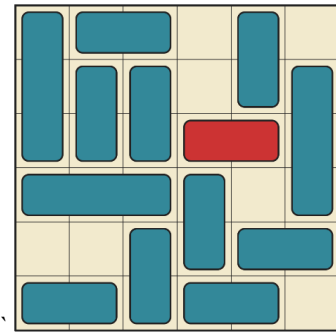
Abstract—*Pathfinding* merupakan salah satu permasalahan inti yang perlu diselesaikan dalam program yang melibatkan pencarian jalur, misalnya pada dunia robotika. Salah satu algoritma *pathfinding* yang tidak rumit, tetapi memiliki waktu penyelesaian masalah yang cukup efisien adalah algoritma *Greedy Best-First Search*. Makalah ini mengkaji penerapan algoritma *Greedy Best-First Search* sebagai metode *pathfinding* untuk menyelesaikan *puzzle* Rush Hour, yang mana *piece* target—biasanya berwarna merah—harus mencapai tempat keluar dengan kendaraan-kendaraan pada grid sebagai rintangan.

Keywords—*Greedy Best-First Search*; *Pathfinding*; *Rush Hour*

I. PENDAHULUAN

Pathfinding merupakan salah satu permasalahan inti yang perlu diimplementasikan dalam program yang melibatkan pencarian jalur, misalnya pada dunia robotika. Ada berbagai algoritma *pathfinding* yang dapat digunakan untuk memecahkan masalah pencarian jalur, misalnya algoritma Dijkstra, A*, *Breadth-First Search* (BFS), *Depth-First Search* (DFS), *Uniform Cost Search*, dan *Greedy Best-First Search*. Salah satu algoritma *pathfinding* yang tidak rumit untuk dipahami dan diimplementasikan, tetapi masih memiliki waktu penyelesaian yang lebih singkat daripada algoritma lainnya adalah algoritma *Greedy Best-First Search*.

Ada beberapa *puzzle* yang dapat mengaplikasikan algoritma *Greedy Best-First Search*, misalnya 8-puzzle atau 15-puzzle, Rubik's Cube 2x2, Word Ladder, dan Rush Hour. Rush Hour adalah *puzzle* yang berasal dari Jepang, permainan ini diciptakan oleh Nobuyuki "Nob" Yoshigara pada tahun 1970. Dalam permainan Rush Hour, *piece* target—biasanya berwarna merah—harus mencapai tempat keluar dengan *piece-piece*—mobil dan truk—pada papan atau *board* (biasanya berukuran *grid* 6x6) sebagai rintangan [1]. *Game* Rush Hour dan variasinya dapat ditemukan di toko aplikasi (*application store*) pada ponsel pintar, dengan nama yang berbeda-beda. Berikut ini adalah salah satu contoh *puzzle* Rush Hour. Ini merupakan salah satu Rush Hour berukuran *grid* 6x6 yang paling sulit.



Gambar 1. Salah satu *puzzle* Rush Hour 6x6 tersulit. Sumber: <https://www.michaelfogleman.com/rush/#HardestPuzzles>

II. TEORI DASAR

A. Algoritma Greedy (*Rakus*)

Algoritma *greedy* adalah salah satu algoritma yang paling terkenal dan salah satu algoritma paling sederhana dalam memecahkan persoalan yang berhubungan dengan optimasi. Cara kerja algoritma ini adalah mencari solusi optimal (nilai maksimal atau minimal yang tersedia) pada setiap langkah, tanpa mempertimbangkan konsekuensi nanti [2].

Misalnya, dalam masalah penukaran uang, jika ingin menukar koin 15 dengan beberapa koin 1, 7, 9. Jika memakai algoritma *greedy*, maka keputusan yang dipilih adalah sebagai berikut:

1. Jumlah nilai koin yang dibutuhkan sekarang: 15.
Nilai terbesar yang bisa dipilih: $9 < 15$.
2. Jumlah nilai koin yang dibutuhkan sekarang: $15 - 9 = 6$.
Nilai terbesar yang bisa dipilih: $9 > 6$.
Ganti ke nilai terbesar kedua.
Nilai terbesar yang bisa dipilih: $7 > 6$.
Ganti ke nilai terbesar ketiga.
Nilai terbesar yang bisa dipilih: $1 < 6$.
3. Jumlah nilai koin yang dibutuhkan sekarang: $6 - 1 = 5$.
Nilai terbesar yang bisa dipilih: $1 < 5$.

4. Jumlah nilai koin yang dibutuhkan sekarang: $5-1 = 4$.
Nilai terbesar yang bisa dipilih: $1 < 4$.
5. Jumlah nilai koin yang dibutuhkan sekarang: $4-1 = 3$.
Nilai terbesar yang bisa dipilih: $1 < 3$.
6. Jumlah nilai koin yang dibutuhkan sekarang: $3-1 = 2$.
Nilai terbesar yang bisa dipilih: $1 < 2$.
7. Jumlah nilai koin yang dibutuhkan sekarang: $2-1 = 1$.
Nilai terbesar yang bisa dipilih: $1 < 1$.
8. Jumlah nilai koin yang dibutuhkan sekarang: $1-1 = 0$.
Pertukaran koin selesai.

Dari tahapan di atas, koin 15 ditukar dengan koin 9, 1, 1, 1, 1, dan 1.

Namun, meskipun algoritma ini cukup efisien, ini tidak selalu menjamin solusi yang optimal. Solusi yang optimal seharusnya adalah: koin 15 ditukar dengan koin 7, 7, dan 1.

Algoritma *Greedy Best-First Search* adalah algoritma penentuan rute yang juga memanfaatkan algoritma *greedy* dalam penggunaannya. Nilai yang diambil oleh algoritma *greedy* ini adalah suatu nilai heuristik (*heuristic cost*) [3].

B. Penentuan Rute

Penentuan Rute atau *pathfinding* adalah suatu algoritma yang menentukan jalur dari suatu titik atau simpul (*node*) awal ke titik atau simpul tujuan. Ada berbagai algoritma *pathfinding* yang dapat digunakan untuk memecahkan masalah penentuan rute. Algoritma-algoritma ini digolongkan berdasarkan penggunaan heuristik, yaitu *informed search* dan *uninformed search*.

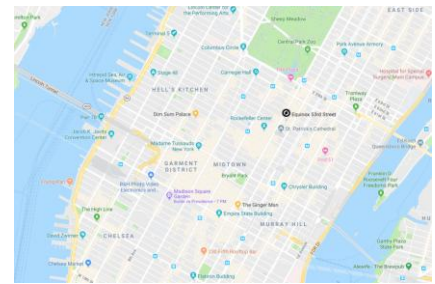
Informed search atau *heuristic search* adalah jenis algoritma penentuan rute yang menggunakan heuristik, misalnya algoritma A* dan *Greedy Best-First Search*. *Uninformed search* adalah algoritma penentuan rute yang tidak menggunakan heuristik, misalnya algoritma *Breadth-First Search* (BFS), *Depth-First Search* (DFS), *Depth Limited Search* (DLS), *Iterative Deepening Search* (IDS), *Uniform Cost Search* (UCS), Dijkstra [3].

C. Heuristik

Ada beberapa buah fungsi heuristik berupa jarak yang dapat dipakai oleh algoritma *Greedy Best-First Search* untuk mendapatkan nilai heuristik (*heuristic cost*), misalnya *Manhattan distance*, *Euclidean distance*, *Chebyshev distance* [4].

1. Manhattan Distance

Manhattan distance atau *taxicab distance* adalah jumlah dari jarak/selisih horizontal dan jarak/selisih vertikal yang dibutuhkan untuk mencapai sebuah tujuan. Fungsi heuristik ini disebut "*Manhattan distance*" karena mirip dengan kota Manhattan dengan jalannya yang berbentuk *grid*.



Gambar 2. Peta kota Manhattan. Sumber: <https://streeteasy.com/blog/nyc-street-numbers-how-manchattan-streets-are-numbered/>

4	3	2	3	4
3	2	1	2	3
2	1		1	2
3	2	1	2	3
4	3	2	3	4

Gambar 3. Ilustrasi jarak dari pusat menggunakan *Manhattan distance*. Sumber: Dokumen Pribadi.

Rumus dari *Manhattan distance* adalah

$$d = |x_{node} - x_{tujuan}| + |y_{node} - y_{tujuan}|$$

2. Euclidean Distance

Euclidean distance adalah jarak langsung (tarik garis lurus) yang dibutuhkan sampai ke tujuan. Fungsi heuristik ini sama dengan mencari sisi miring menggunakan Teorema Pythagoras.

$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
2	1		1	2
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$

Gambar 4. Ilustrasi jarak dari pusat menggunakan *Euclidean distance*. Sumber: Dokumen Pribadi.

Rumus dari *Euclidean distance* adalah

$$d = \sqrt{(|x_{node} - x_{tujuan}|)^2 + (|y_{node} - y_{tujuan}|)^2}$$

3. Chebyshev Distance

Chebyshev distance adalah jarak maksimum antara jarak/selisih horizontal dan jarak/selisih vertikal.

2	2	2	2	2
2	1	1	1	2
2	1		1	2
2	1	1	1	2
2	2	2	2	2

Gambar 5. Ilustrasi jarak dari pusat menggunakan *Chebyshev distance*. Sumber: Dokumen Pribadi.

Rumus dari *Chebyshev distance* adalah

$$d = \max(|x_{node} - x_{tujuan}|, |y_{node} - y_{tujuan}|) .$$

D. Greedy Best-First Search

Algoritma *Greedy Best-First Search* adalah variasi dari algoritma pencarian *Best-First Search* yang menggunakan algoritma *greedy*. Algoritma *Best-First Search* adalah algoritma yang memilih simpul yang paling menjanjikan untuk membangkitkan simpul selanjutnya berdasarkan fungsi evaluasi tertentu. Fungsi evaluasi yang digunakan tergantung pada algoritma *Best-First Search* yang digunakan [5].

Algoritma *Greedy Best-First Search* adalah algoritma penentuan jalur yang mudah untuk dipahami dan diimplementasikan. Meskipun tidak memberikan jumlah langkah yang paling optimal, algoritma *Greedy Best-First Search* pada umumnya tetap lebih cepat daripada algoritma lain, seperti *Uniform Cost Search* dan *A**, yang lebih rumit. Pernyataan ini akan dibuktikan dalam bagian IV. Pengujian.

Algoritma ini bekerja dengan cara memadukan konsep algoritma *greedy* dengan fungsi heuristik. Algoritma *greedy* akan mengambil nilai heuristik (*heuristic cost*) terkecil pada setiap tahap, tanpa mempertimbangkan konsekuensi dari hasil akhir, misalnya jumlah langkah yang ada menjadi lebih banyak [3]. Fungsi evaluasi dari algoritma *Greedy Best-First Search* adalah

$$f(n) = h(n) .$$

Karena dapat mencari solusi dengan cepat dan konsepnya yang lebih mudah dipahami, algoritma ini cocok untuk diaplikasikan dalam kasus yang lebih mementingkan dapat mencari solusi dengan cepat daripada mencari langkah paling optimal.

III. PEMBAHASAN

A. Aturan Permainan Rush Hour

Rush Hour adalah sebuah *puzzle* dua dimensi pada *board* dengan ukuran *grid* tertentu, biasanya 6x6. *Piece-piece* yang ada pada *board* hanya boleh memiliki satu baris dan dua/lebih kolom (horizontal), atau satu kolom dan dua/lebih baris

(vertikal). *Piece* hanya dapat bergerak secara horizontal—jika *piece* diletakkan secara horizontal—atau vertikal—jika diletakkan vertikal—. *Piece* target—umumnya berwarna merah—harus keluar melalui pintu keluar, sedangkan *piece-piece* lainnya tidak boleh keluar.

B. Konfigurasi Puzzle

Untuk merealisasikan konsep tersebut beserta dengan algoritma penyelesaian masalahnya, petakan terlebih dahulu susunan-susunan *piece* dengan:

- Setiap huruf merepresentasikan sebuah *piece*
- Huruf P sebagai *piece* target yang akan keluar
- Huruf K sebagai pintu keluar
- Titik '.' sebagai *tile* kosong
- Jumlah baris, jumlah kolom, dan jumlah *piece* (selain *piece* target/merah), bila perlu.

Contoh konfigurasinya adalah sebagai berikut:

6 6

11

AAB..F

..BCDF

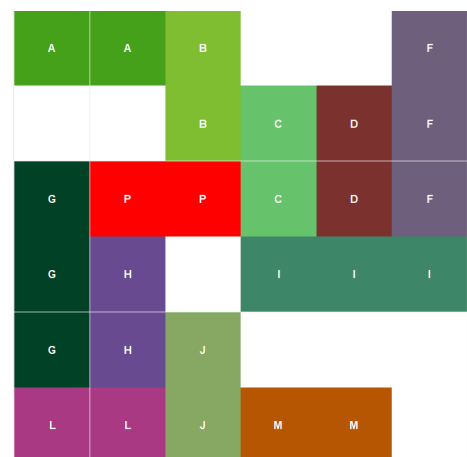
GPPCDFK

GH.III

GHJ...

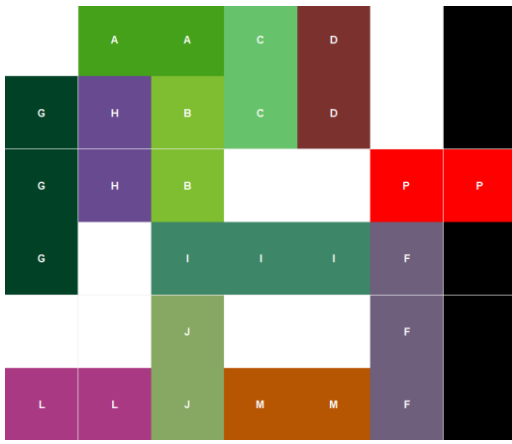
LLJMM.

Berdasarkan konfigurasi di atas, akan terbentuk papan *puzzle* yang terlihat seperti ini:



Gambar 6. Visualisasi berdasarkan konfigurasi contoh. Sumber: Dokumen Pribadi.

Ini adalah *state* (keadaan) awal dari *puzzle*. Setelah *puzzle* diselesaikan, *state* akhirnya akan menjadi seperti ini:



Gambar 7. Solusi berdasarkan konfigurasi contoh. Sumber: Dokumen Pribadi.

Atau dalam representasi teks:

.AACD.

GHBCD.

GHB...PP

G.IIIIF

...J...F

LLJMMF

Namun, sebelum bisa mendapat *state* akhir dari *puzzle*, fungsi heuristik harus ditentukan terlebih dahulu, lalu algoritma penyelesaian *puzzle* diimplementasikan.

C. Penentuan Fungsi Heuristik yang Digunakan

Ada tiga buah fungsi heuristik yang telah dibahas sebelumnya dan akan dipertimbangkan, yaitu *Manhattan distance*, *Euclidean distance*, dan *Chebyshev distance*.

Jika merujuk pada bagian A. Aturan Permainan Rush Hour, salah satu aturan dari *puzzle* ini adalah bahwa *piece* hanya dapat bergerak secara horizontal—jika *piece* diletakkan secara horizontal—atau vertikal—jika diletakkan vertikal—.

Untuk *Manhattan distance*, jarak yang ditempuh selalu sama dengan jarak yang ditempuh secara vertikal, lalu jarak yang ditempuh secara horizontal. Jika merujuk pada aturan permainan tersebut, maka rumus

$$d = |x_{node} - x_{tujuan}| + |y_{node} - y_{tujuan}|$$

akan sama saja dengan

$$d_{horizontal} = |x_{node} - x_{tujuan}| + 0$$

$$= |x_{node} - x_{tujuan}| \text{ dan}$$

$$d_{vertikal} = 0 + |y_{node} - y_{tujuan}|$$

$$= |y_{node} - y_{tujuan}|.$$

Sedangkan pada *Euclidean distance*, jarak yang ditempuh merupakan rumus dari Teorema Pythagoras untuk mencari sisi miring, atau jarak yang ditempuh langsung dengan

menarik garis lurus dari titik asal ke titik tujuan. Jika merujuk pada aturan permainan tersebut, maka rumus

$$d = \sqrt{(|x_{node} - x_{tujuan}|)^2 + (|y_{node} - y_{tujuan}|)^2}$$

akan sama saja dengan

$$d_{horizontal} = \sqrt{(|x_{node} - x_{tujuan}|)^2 + (0)^2}$$

$$= |x_{node} - x_{tujuan}| \text{ dan}$$

$$d_{vertikal} = \sqrt{(0)^2 + (|y_{node} - y_{tujuan}|)^2}$$

$$= |y_{node} - y_{tujuan}|.$$

Untuk *Chebyshev distance*, jika merujuk pada aturan yang sama, maka rumus

$$d = \max(|x_{node} - x_{tujuan}|, |y_{node} - y_{tujuan}|)$$

akan sama saja dengan

$$d_{horizontal} = \max(|x_{node} - x_{tujuan}|, 0)$$

$$= |x_{node} - x_{tujuan}| \text{ dan}$$

$$d_{vertikal} = \max(0, |y_{node} - y_{tujuan}|)$$

$$= |y_{node} - y_{tujuan}|.$$

Karena semua jarak yang ditempuh $d_{horizontal}$ dan $d_{vertikal}$ memiliki rumus yang sama di antara ketiga fungsi heuristik, maka untuk kasus ini, *Manhattan distance*, *Euclidean distance*, dan *Chebyshev distance* pada dasarnya adalah sama, sehingga dapat memilih fungsi heuristik yang mana saja. Pada program yang dibuat, fungsi heuristik yang dipilih adalah *Manhattan distance* karena memiliki operasi matematika yang lebih sederhana.

Manhattan distance yang diukur dalam program ini adalah jarak *piece* target (*piece* merah)—dari *tile/grid* terdekat ke pintu keluar—ke pintu keluar. Contohnya, berdasarkan Gambar 6., maka *Manhattan distance* dari *piece* merah ke pintu keluar adalah $6 - 2 = 4$. Enam merupakan indeks dari 'K'. Sedangkan dua merupakan indeks dari *grid* pada *piece* merah yang paling dekat ke pintu keluar.

D. Algoritma Penyelesaian Puzzle

Setelah fungsi heuristik ditentukan dan diimplementasikan, akan dibuat algoritma penyelesaian *puzzle*, yaitu algoritma *Greedy Best-First Search*. Berikut adalah cara kerja dari algoritma penyelesaian *puzzle* Rush Hour:

1. Program membaca konfigurasi yang berupa *file* teks.
 - *Piece* akan dikelompokkan berdasarkan huruf, kecuali huruf 'K' yang melambangkan pintu keluar.
2. Program membuat struktur data *openSet* dan *closedSet*

- openSet: *Priority queue* yang mengurutkan *state* berdasarkan *heuristic cost* (*Manhattan distance* terdekat dari blok 'P' ke 'K').
 - closedSet: *hash set* untuk mencatat *board* yang telah dieksplor. *Hash set* dipilih dibandingkan *list*, karena dapat otomatis menolak duplikat (*state* yang sama dengan *state* lainnya).
3. Inisialisasi *state* awal, masukkan *state* tersebut ke openSet.
 4. Lakukan *loop* sampai openSet kosong atau sampai batas jumlah iterasi tertentu (diset dengan angka besar seperti 500000, untuk mengantisipasi jika program *loop* selamanya).
 - i. Ambil *state*—*child* dari *state* sebelumnya—dari openSet dengan *heuristic cost* terendah. Hapus/pop *state* tersebut dari openSet.
 - ii. Jika *state* ada di closedSet—lokasi semua *piece* sekarang sama dengan yang pernah terjadi sebelumnya—, *skip* iterasi sekarang (kembali ke tahap i.).
 - iii. Jika *state* tidak ada di closedSet, tandai *state* sekarang sebagai dikunjungi, masukkan ke closeSet.
 - iv. Jika pada *state* sekarang, *piece* 'P' sudah mencapai pintu keluar, maka *return path*—urutan *piece* minimal yang digerakkan agar 'P' sampai ke pintu keluar—, yang bisa didapatkan dari *backtracking* ke *state* awal; beserta dengan arah pergerakan *piece-piece* tersebut.
 - v. Jika belum, masukkan semua *state* hasil pergerakan satu langkah semua *piece* yang mungkin terjadi, dan tidak ada di dalam closeSet, ke dalam openSet.
 5. Program menampilkan hasil solusi pada GUI dan terminal.

IV. PENGUJIAN

Untuk menguji penerapan dari algoritma *Greedy Best-First Search* pada *puzzle* Rush Hour dan membuktikan pernyataan II. D. *Greedy Best First Search*, akan digunakan tiga buah kasus percobaan.

Kasus pertama menggunakan Gambar 6.. Berikut merupakan teks konfigurasi kasus pertama dengan hasil percobaannya:

6 6

11

AAB...F

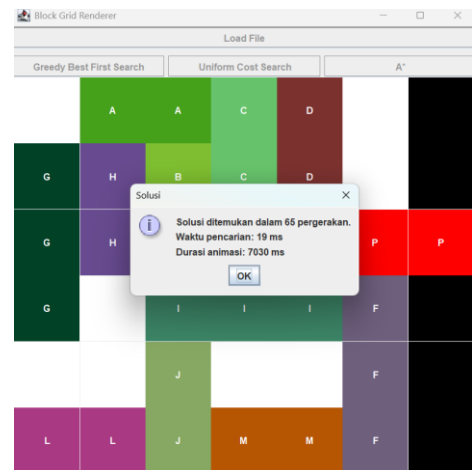
..BCDF

GPPCDFK

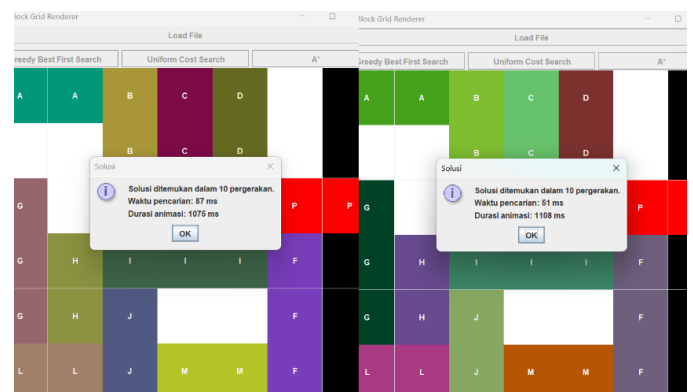
GH.III

GHJ...

LLJMM.



Gambar 8. Hasil pengujian kasus pertama dengan *Greedy Best-First Search*. Sumber: Dokumen Pribadi.



Gambar 9. Hasil pengujian kasus pertama dengan A* dan UCS. Sumber: Dokumen Pribadi.

Berdasarkan hasil pengujian di atas, waktu yang dibutuhkan untuk mendapatkan hasil menggunakan *Greedy Best First Search* adalah 19 ms, 65 langkah. Sedangkan dengan A* dan *Uniform Cost Search*, waktu yang dibutuhkan berturut-turut adalah 87 ms dan 51 ms, keduanya 10 langkah. Dalam kasus ini, algoritma A* dan UCS membutuhkan waktu yang lebih lama daripada *Greedy Best-First Search*, meskipun memiliki jumlah pergerakan atau langkah-langkah yang lebih optimal.

Perhatikan bahwa durasi pergerakan pada GUI berbeda dengan proses algoritma sebenarnya, karena pergerakan pada GUI mereplikasi pergerakan yang sudah terbuat oleh program dan interval 100 ms tiap langkah.

Langkah-langkahnya dengan *Greedy Best-First Search* adalah sebagai berikut:

```

1  Algoritma: Greedy Best First Search
2  Waktu Eksekusi Algoritma: 19 ms
3  Jumlah Langkah: 65
4  =====
5  Langkah 0 (Initial State):
6  AAB..F
7  ..BCDF
8  GPPCDF
9  GH.III
10 GHJ...
11 LLJMM.
12 -----
13 Langkah 1: C_UP
14 AABC.F
15 ..BCDF
16 GPP.DF
17 GH.III
18 GHJ...
19 LLJMM.
20 -----
21 Langkah 2: P_RIGHT
22 AABC.F
23 ..BCDF
24 G.PPDF
25 GH.III
26 GHJ...
27 LLJMM.
28 -----
29 Langkah 3: D_UP
30 AABCFD
31 ..BCDF
32 G.PPF.
33 GH.III
34 GHJ...
35 LLJMM.
36 -----
37 Langkah 4: P_RIGHT
38 AABCFD
39 ..BCDF
40 G..PPF
41 GH.III
42 GHJ...
43 LLJMM.
44 -----
45 Langkah 5: G_UP
46 AABCFD
47 ..BCDF
48 G..PPF
49 GH.III
50 GHJ...
51 LLJMM.
52 -----
53 Langkah 6: M_RIGHT
54 AABCFD
55 ..BCDF
56 G..PPF
57 GH.III
58 GHJ...
59 LLJMM.
60 -----
59 Langkah 6: M_RIGHT
60 AABCFD
61 G..BCDF
62 G..PPF
63 GH.III
64 GHJ...
65 LLJMM
66 -----
67 Langkah 7: J_UP
68 AABCFD
69 AABCFD
70 G..BCDF
71 G..PPF
72 GHJIII
73 .HJ...
74 LL..MM
75 -----
76 Langkah 8: L_RIGHT
77 AABCFD
78 AABCFD
79 G..BCDF
80 G..PPF
81 GHJIII
82 .HJ...
83 LL..MM
84 -----
85 Langkah 9: M_LEFT
86 AABCFD
87 AABCFD
88 G..BCDF
89 G..PPF
90 GHJIII
91 .HJ...
92 LL..MM
93 -----

```

Gambar 10. Langkah 1-8 pengujian kasus pertama. Sumber: Dokumen Pribadi

Karena jumlah langkah setiap kasus sangat banyak dengan jumlah maksimal halaman makalah yang terbatas, maka langkah yang ditampilkan hanyalah beberapa langkah awal dan akhir.

```

517 Langkah 57: I_LEFT
518 .AACD.
519 GHBCDF
520 GHBPFF
521 GIII.F
522 ..J...
523 LLJMM
524 -----
525 Langkah 58: M_LEFT
526 .AACD.
527 GHBCDF
528 GHBPFF
529 GIII.F
530 ..J...
531 LLJMM
532 -----
533 Langkah 59: F_DOWN
534 .AACD.
535 GHBCD.
536 GHBPFF
537 GIII.F
538 ..J...
539 LLJMM
540 -----
541 Langkah 60: M_RIGHT
542 .AACD.
543 GHBCD.
544 GHBPFF
545 GIII.F
546 ..J...
547 LLJMM
548 -----
549 Langkah 61: I_RIGHT
550 .AACD.
551 GHBCD.
552 GHBPFF
553 GIII.F
554 ..J...
555 LLJMM
556 -----
557 Langkah 62: M_LEFT
558 .AACD.
559 GHBCD.
560 GHBPFF
561 GIII.F
562 ..J...
563 LLJMM
564 -----
565 Langkah 63: F_DOWN
566 .AACD.
567 GHBCD.
568 GHBPFF
569 GIII.F
570 ..J...
571 LLJMM
572 -----
573 Langkah 64: P_RIGHT
574 .AACD.
575 GHBCD.
576 GHBPFF
577 GIII.F
578 ..J...
579 LLJMMF
580 -----
581 Langkah 65: P_RIGHT
582 .AACD.
583 GHBCD.
584 GHBPFF
585 GIII.F
586 ..J...
587 LLJMMF
588 -----
589 Langkah 66: M_LEFT
590 .AACD.
591 GHBCD.
592 GHBPFF
593 GIII.F
594 ..J...
595 LLJMMF
596 -----
597 Langkah 67: F_DOWN
598 .AACD.
599 GHBCD.
600 GHBPFF
601 GIII.F
602 ..J...
603 LLJMMF
604 -----

```

Gambar 11. Langkah 57-65 pengujian kasus pertama. Sumber: Dokumen Pribadi

Kasus kedua menggunakan contoh berikut:



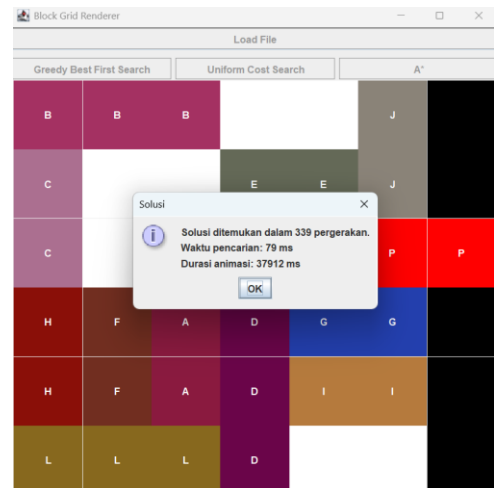
Gambar 12. Contoh/kasus kedua Rush Hour. Sumber: <https://stackoverflow.com/questions/2877724>

Berikut adalah teks konfigurasi dari kasus kedua dengan hasil dari percobaannya:

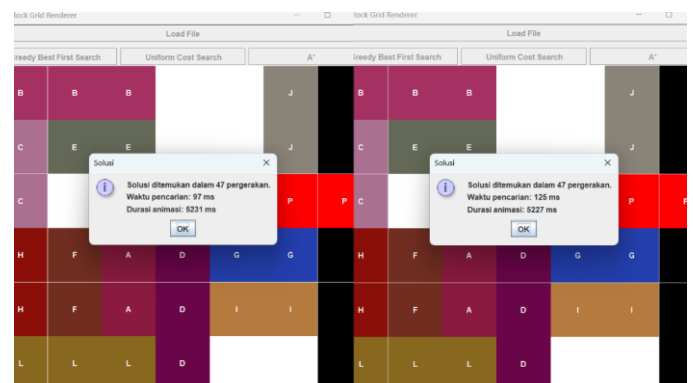
```

6 6
11
..ABBB
C.ADEE
CPPD..K
.F.DGG
HFII.J
HLLL.J

```



Gambar 13. Hasil pengujian kasus kedua dengan *Greedy Best-First Search*. Sumber: Dokumen Pribadi.



Gambar 14. Hasil pengujian kasus kedua dengan A* dan UCS. Sumber: Dokumen Pribadi.

Berdasarkan hasil pengujian di atas, waktu yang dibutuhkan untuk mendapatkan hasil menggunakan *Greedy Best First Search* adalah 79 ms, 339 langkah. Sedangkan dengan A* dan *Uniform Cost Search*, waktu yang dibutuhkan berturut-turut adalah 97 ms dan 125 ms, keduanya 47 langkah. Dalam kasus kedua, algoritma A* dan UCS juga membutuhkan waktu yang lebih lama daripada *Greedy Best-First Search*. Langkah-langkahnya dengan *Greedy Best-First Search* adalah sebagai berikut:


```

1 Algoritma: Greedy Best First Search
2 Waktu Eksekusi Algoritma: 90 ms
3 Jumlah Langkah: 421
4 =====
5 Langkah 0 (Initial State):
6 ABB.E.
7 ACD.EF
8 ACDPPF.
9 GGGH.F
10 ..IHJJ
11 LLIMM.
12 -----
13 Langkah 1: M_RIGHT
14 ABB.E.
15 ACD.EF
16 ACDPPF.
17 GGGH.F
18 ..IHJJ
19 LLIMM.
20 -----
21 Langkah 2: H_DOWN
22 ABB.E.
23 ACD.EF
24 ACDPPF.
25 GGGH.F
26 ..IHJJ
27 LLIMM.
28 -----
29 Langkah 3: G_RIGHT
30 ABB.E.
31 ACD.EF
32 ACDPPF.
33 GGGH.F
34 ..IHJJ
35 LLIMM.
36 -----
37 Langkah 4: G_RIGHT
38 ABB.E.
39 ACD.EF
40 ACDPPF.
41 GGGH.F
42 ..IHJJ
43 LLIMM.
44 -----
45 Langkah 5: F_UP
46 ABB.EF
47 ACD.EF
48 ACDPPF.
49 GGGH.F
50 ..IHJJ
51 LLIMM.
52 -----
53 Langkah 6: G_RIGHT
54 ABB.EF
55 ACD.EF
56 ACDPPF.
57 GGGH.F
58 ..IHJJ
59 LLIMM.
60 -----
61 Langkah 7: I_UP
62 ABB.EF
63 ACD.EF
64 ACDPPF.
65 GGGH.F
66 ..IHJJ
67 LLIMM.
68 -----
69 Langkah 8: L_RIGHT
70 ABB.EF
71 ACD.EF
72 ACDPPF.
73 GGGH.F
74 ..IHJJ
75 LLIMM.
76 -----
77 Langkah 9: C_DOWN
78 ABB.EF
79 ACD.EF
80 ACDPPF.
81 GGGH.F
82 ..IHJJ
83 LLIMM.
84 -----
85 Langkah 10: F_DOWN
86 ABB.EF
87 ACD.EF
88 ACDPPF.
89 GGGH.F
90 ..IHJJ
91 LLIMM.
92 -----
93 Langkah 11: M_LEFT
94 ABB.EF
95 ACD.EF
96 ACDPPF.
97 GGGH.F
98 ..IHJJ
99 LLIMM.
100 -----
101 Langkah 12: H_DOWN
102 ABB.EF
103 ACD.EF
104 ACDPPF.
105 GGGH.F
106 ..IHJJ
107 LLIMM.
108 -----
109 Langkah 13: G_RIGHT
110 ABB.EF
111 ACD.EF
112 ACDPPF.
113 GGGH.F
114 ..IHJJ
115 LLIMM.
116 -----
117 Langkah 14: G_RIGHT
118 ABB.EF
119 ACD.EF
120 ACDPPF.
121 GGGH.F
122 ..IHJJ
123 LLIMM.
124 -----
125 Langkah 15: F_UP
126 ABB.EF
127 ACD.EF
128 ACDPPF.
129 GGGH.F
130 ..IHJJ
131 LLIMM.
132 -----
133 Langkah 16: G_RIGHT
134 ABB.EF
135 ACD.EF
136 ACDPPF.
137 GGGH.F
138 ..IHJJ
139 LLIMM.
140 -----
141 Langkah 17: I_UP
142 ABB.EF
143 ACD.EF
144 ACDPPF.
145 GGGH.F
146 ..IHJJ
147 LLIMM.
148 -----
149 Langkah 18: L_RIGHT
150 ABB.EF
151 ACD.EF
152 ACDPPF.
153 GGGH.F
154 ..IHJJ
155 LLIMM.
156 -----
157 Langkah 19: C_DOWN
158 ABB.EF
159 ACD.EF
160 ACDPPF.
161 GGGH.F
162 ..IHJJ
163 LLIMM.
164 -----
165 Langkah 20: F_DOWN
166 ABB.EF
167 ACD.EF
168 ACDPPF.
169 GGGH.F
170 ..IHJJ
171 LLIMM.
172 -----
173 Langkah 21: M_LEFT
174 ABB.EF
175 ACD.EF
176 ACDPPF.
177 GGGH.F
178 ..IHJJ
179 LLIMM.
180 -----
181 Langkah 22: H_DOWN
182 ABB.EF
183 ACD.EF
184 ACDPPF.
185 GGGH.F
186 ..IHJJ
187 LLIMM.
188 -----
189 Langkah 23: G_RIGHT
190 ABB.EF
191 ACD.EF
192 ACDPPF.
193 GGGH.F
194 ..IHJJ
195 LLIMM.
196 -----
197 Langkah 24: G_RIGHT
198 ABB.EF
199 ACD.EF
200 ACDPPF.
201 GGGH.F
202 ..IHJJ
203 LLIMM.
204 -----
205 Langkah 25: F_UP
206 ABB.EF
207 ACD.EF
208 ACDPPF.
209 GGGH.F
210 ..IHJJ
211 LLIMM.
212 -----
213 Langkah 26: G_RIGHT
214 ABB.EF
215 ACD.EF
216 ACDPPF.
217 GGGH.F
218 ..IHJJ
219 LLIMM.
220 -----
221 Langkah 27: I_UP
222 ABB.EF
223 ACD.EF
224 ACDPPF.
225 GGGH.F
226 ..IHJJ
227 LLIMM.
228 -----
229 Langkah 28: L_RIGHT
230 ABB.EF
231 ACD.EF
232 ACDPPF.
233 GGGH.F
234 ..IHJJ
235 LLIMM.
236 -----
237 Langkah 29: C_DOWN
238 ABB.EF
239 ACD.EF
240 ACDPPF.
241 GGGH.F
242 ..IHJJ
243 LLIMM.
244 -----
245 Langkah 30: F_DOWN
246 ABB.EF
247 ACD.EF
248 ACDPPF.
249 GGGH.F
250 ..IHJJ
251 LLIMM.
252 -----
253 Langkah 31: M_LEFT
254 ABB.EF
255 ACD.EF
256 ACDPPF.
257 GGGH.F
258 ..IHJJ
259 LLIMM.
260 -----
261 Langkah 32: H_DOWN
262 ABB.EF
263 ACD.EF
264 ACDPPF.
265 GGGH.F
266 ..IHJJ
267 LLIMM.
268 -----
269 Langkah 33: G_RIGHT
270 ABB.EF
271 ACD.EF
272 ACDPPF.
273 GGGH.F
274 ..IHJJ
275 LLIMM.
276 -----
277 Langkah 34: G_RIGHT
278 ABB.EF
279 ACD.EF
280 ACDPPF.
281 GGGH.F
282 ..IHJJ
283 LLIMM.
284 -----
285 Langkah 35: F_UP
286 ABB.EF
287 ACD.EF
288 ACDPPF.
289 GGGH.F
290 ..IHJJ
291 LLIMM.
292 -----
293 Langkah 36: G_RIGHT
294 ABB.EF
295 ACD.EF
296 ACDPPF.
297 GGGH.F
298 ..IHJJ
299 LLIMM.
300 -----
301 Langkah 37: I_UP
302 ABB.EF
303 ACD.EF
304 ACDPPF.
305 GGGH.F
306 ..IHJJ
307 LLIMM.
308 -----
309 Langkah 38: L_RIGHT
310 ABB.EF
311 ACD.EF
312 ACDPPF.
313 GGGH.F
314 ..IHJJ
315 LLIMM.
316 -----
317 Langkah 39: C_DOWN
318 ABB.EF
319 ACD.EF
320 ACDPPF.
321 GGGH.F
322 ..IHJJ
323 LLIMM.
324 -----
325 Langkah 40: F_DOWN
326 ABB.EF
327 ACD.EF
328 ACDPPF.
329 GGGH.F
330 ..IHJJ
331 LLIMM.
332 -----
333 Langkah 41: M_LEFT
334 ABB.EF
335 ACD.EF
336 ACDPPF.
337 GGGH.F
338 ..IHJJ
339 LLIMM.
340 -----
341 Langkah 42: H_DOWN
342 ABB.EF
343 ACD.EF
344 ACDPPF.
345 GGGH.F
346 ..IHJJ
347 LLIMM.
348 -----
349 Langkah 43: G_RIGHT
350 ABB.EF
351 ACD.EF
352 ACDPPF.
353 GGGH.F
354 ..IHJJ
355 LLIMM.
356 -----
357 Langkah 44: G_RIGHT
358 ABB.EF
359 ACD.EF
360 ACDPPF.
361 GGGH.F
362 ..IHJJ
363 LLIMM.
364 -----
365 Langkah 45: F_UP
366 ABB.EF
367 ACD.EF
368 ACDPPF.
369 GGGH.F
370 ..IHJJ
371 LLIMM.
372 -----
373 Langkah 46: G_RIGHT
374 ABB.EF
375 ACD.EF
376 ACDPPF.
377 GGGH.F
378 ..IHJJ
379 LLIMM.
380 -----
381 Langkah 47: I_UP
382 ABB.EF
383 ACD.EF
384 ACDPPF.
385 GGGH.F
386 ..IHJJ
387 LLIMM.
388 -----
389 Langkah 48: L_RIGHT
390 ABB.EF
391 ACD.EF
392 ACDPPF.
393 GGGH.F
394 ..IHJJ
395 LLIMM.
396 -----
397 Langkah 49: C_DOWN
398 ABB.EF
399 ACD.EF
400 ACDPPF.
401 GGGH.F
402 ..IHJJ
403 LLIMM.
404 -----
405 Langkah 50: F_DOWN
406 ABB.EF
407 ACD.EF
408 ACDPPF.
409 GGGH.F
410 ..IHJJ
411 LLIMM.
412 -----
413 Langkah 51: M_LEFT
414 ABB.EF
415 ACD.EF
416 ACDPPF.
417 GGGH.F
418 ..IHJJ
419 LLIMM.
420 -----
421 Langkah 52: H_DOWN
422 ABB.EF
423 ACD.EF
424 ACDPPF.
425 GGGH.F
426 ..IHJJ
427 LLIMM.
428 -----
429 Langkah 53: G_RIGHT
430 ABB.EF
431 ACD.EF
432 ACDPPF.
433 GGGH.F
434 ..IHJJ
435 LLIMM.
436 -----
437 Langkah 54: G_RIGHT
438 ABB.EF
439 ACD.EF
440 ACDPPF.
441 GGGH.F
442 ..IHJJ
443 LLIMM.
444 -----
445 Langkah 55: F_UP
446 ABB.EF
447 ACD.EF
448 ACDPPF.
449 GGGH.F
450 ..IHJJ
451 LLIMM.
452 -----
453 Langkah 56: G_RIGHT
454 ABB.EF
455 ACD.EF
456 ACDPPF.
457 GGGH.F
458 ..IHJJ
459 LLIMM.
460 -----
461 Langkah 57: I_UP
462 ABB.EF
463 ACD.EF
464 ACDPPF.
465 GGGH.F
466 ..IHJJ
467 LLIMM.
468 -----
469 Langkah 58: L_RIGHT
470 ABB.EF
471 ACD.EF
472 ACDPPF.
473 GGGH.F
474 ..IHJJ
475 LLIMM.
476 -----
477 Langkah 59: C_DOWN
478 ABB.EF
479 ACD.EF
480 ACDPPF.
481 GGGH.F
482 ..IHJJ
483 LLIMM.
484 -----
485 Langkah 60: F_DOWN
486 ABB.EF
487 ACD.EF
488 ACDPPF.
489 GGGH.F
490 ..IHJJ
491 LLIMM.
492 -----
493 Langkah 61: M_LEFT
494 ABB.EF
495 ACD.EF
496 ACDPPF.
497 GGGH.F
498 ..IHJJ
499 LLIMM.
500 -----
501 Langkah 62: H_DOWN
502 ABB.EF
503 ACD.EF
504 ACDPPF.
505 GGGH.F
506 ..IHJJ
507 LLIMM.
508 -----
509 Langkah 63: G_RIGHT
510 ABB.EF
511 ACD.EF
512 ACDPPF.
513 GGGH.F
514 ..IHJJ
515 LLIMM.
516 -----
517 Langkah 64: G_RIGHT
518 ABB.EF
519 ACD.EF
520 ACDPPF.
521 GGGH.F
522 ..IHJJ
523 LLIMM.
524 -----
525 Langkah 65: F_UP
526 ABB.EF
527 ACD.EF
528 ACDPPF.
529 GGGH.F
530 ..IHJJ
531 LLIMM.
532 -----
533 Langkah 66: G_RIGHT
534 ABB.EF
535 ACD.EF
536 ACDPPF.
537 GGGH.F
538 ..IHJJ
539 LLIMM.
540 -----
541 Langkah 67: I_UP
542 ABB.EF
543 ACD.EF
544 ACDPPF.
545 GGGH.F
546 ..IHJJ
547 LLIMM.
548 -----
549 Langkah 68: L_RIGHT
550 ABB.EF
551 ACD.EF
552 ACDPPF.
553 GGGH.F
554 ..IHJJ
555 LLIMM.
556 -----
557 Langkah 69: C_DOWN
558 ABB.EF
559 ACD.EF
560 ACDPPF.
561 GGGH.F
562 ..IHJJ
563 LLIMM.
564 -----
565 Langkah 70: F_DOWN
566 ABB.EF
567 ACD.EF
568 ACDPPF.
569 GGGH.F
570 ..IHJJ
571 LLIMM.
572 -----
573 Langkah 71: M_LEFT
574 ABB.EF
575 ACD.EF
576 ACDPPF.
577 GGGH.F
578 ..IHJJ
579 LLIMM.
580 -----
581 Langkah 72: H_DOWN
582 ABB.EF
583 ACD.EF
584 ACDPPF.
585 GGGH.F
586 ..IHJJ
587 LLIMM.
588 -----
589 Langkah 73: G_RIGHT
590 ABB.EF
591 ACD.EF
592 ACDPPF.
593 GGGH.F
594 ..IHJJ
595 LLIMM.
596 -----
597 Langkah 74: G_RIGHT
598 ABB.EF
599 ACD.EF
600 ACDPPF.
601 GGGH.F
602 ..IHJJ
603 LLIMM.
604 -----
605 Langkah 75: F_UP
606 ABB.EF
607 ACD.EF
608 ACDPPF.
609 GGGH.F
610 ..IHJJ
611 LLIMM.
612 -----
613 Langkah 76: G_RIGHT
614 ABB.EF
615 ACD.EF
616 ACDPPF.
617 GGGH.F
618 ..IHJJ
619 LLIMM.
620 -----
621 Langkah 77: I_UP
622 ABB.EF
623 ACD.EF
624 ACDPPF.
625 GGGH.F
626 ..IHJJ
627 LLIMM.
628 -----
629 Langkah 78: L_RIGHT
630 ABB.EF
631 ACD.EF
632 ACDPPF.
633 GGGH.F
634 ..IHJJ
635 LLIMM.
636 -----
637 Langkah 79: C_DOWN
638 ABB.EF
639 ACD.EF
640 ACDPPF.
641 GGGH.F
642 ..IHJJ
643 LLIMM.
644 -----
645 Langkah 80: F_DOWN
646 ABB.EF
647 ACD.EF
648 ACDPPF.
649 GGGH.F
650 ..IHJJ
651 LLIMM.
652 -----
653 Langkah 81: M_LEFT
654 ABB.EF
655 ACD.EF
656 ACDPPF.
657 GGGH.F
658 ..IHJJ
659 LLIMM.
660 -----
661 Langkah 82: H_DOWN
662 ABB.EF
663 ACD.EF
664 ACDPPF.
665 GGGH.F
666 ..IHJJ
667 LLIMM.
668 -----
669 Langkah 83: G_RIGHT
670 ABB.EF
671 ACD.EF
672 ACDPPF.
673 GGGH.F
674 ..IHJJ
675 LLIMM.
676 -----
677 Langkah 84: G_RIGHT
678 ABB.EF
679 ACD.EF
680 ACDPPF.
681 GGGH.F
682 ..IHJJ
683 LLIMM.
684 -----
685 Langkah 85: F_UP
686 ABB.EF
687 ACD.EF
688 ACDPPF.
689 GGGH.F
690 ..IHJJ
691 LLIMM.
692 -----
693 Langkah 86: G_RIGHT
694 ABB.EF
695 ACD.EF
696 ACDPPF.
697 GGGH.F
698 ..IHJJ
699 LLIMM.
700 -----
701 Langkah 87: I_UP
702 ABB.EF
703 ACD.EF
704 ACDPPF.
705 GGGH.F
706 ..IHJJ
707 LLIMM.
708 -----
709 Langkah 88: L_RIGHT
710 ABB.EF
711 ACD.EF
712 ACDPPF.
713 GGGH.F
714 ..IHJJ
715 LLIMM.
716 -----
717 Langkah 89: C_DOWN
718 ABB.EF
719 ACD.EF
720 ACDPPF.
721 GGGH.F
722 ..IHJJ
723 LLIMM.
724 -----
725 Langkah 90: F_DOWN
726 ABB.EF
727 ACD.EF
728 ACDPPF.
729 GGGH.F
730 ..IHJJ
731 LLIMM.
732 -----
733 Langkah 91: M_LEFT
734 ABB.EF
735 ACD.EF
736 ACDPPF.
737 GGGH.F
738 ..IHJJ
739 LLIMM.
740 -----
741 Langkah 92: H_DOWN
742 ABB.EF
743 ACD.EF
744 ACDPPF.
745 GGGH.F
746 ..IHJJ
747 LLIMM.
748 -----
749 Langkah 93: G_RIGHT
750 ABB.EF
751 ACD.EF
752 ACDPPF.
753 GGGH.F
754 ..IHJJ
755 LLIMM.
756 -----
757 Langkah 94: G_RIGHT
758 ABB.EF
759 ACD.EF
760 ACDPPF.
761 GGGH.F
762 ..IHJJ
763 LLIMM.
764 -----
765 Langkah 95: F_UP
766 ABB.EF
767 ACD.EF
768 ACDPPF.
769 GGGH.F
770 ..IHJJ
771 LLIMM.
772 -----
773 Langkah 96: G_RIGHT
774 ABB.EF
775 ACD.EF
776 ACDPPF.
777 GGGH.F
778 ..IHJJ
779 LLIMM.
780 -----
781 Langkah 97: I_UP
782 ABB.EF
783 ACD.EF
784 ACDPPF.
785 GGGH.F
786 ..IHJJ
787 LLIMM.
788 -----
789 Langkah 98: L_RIGHT
790 ABB.EF
791 ACD.EF
792 ACDPPF.
793 GGGH.F
794 ..IHJJ
795 LLIMM.
796 -----
797 Langkah 99: C_DOWN
800 ABB.EF
801 ACD.EF
802 ACDPPF.
803 GGGH.F
804 ..IHJJ
805 LLIMM.
806 -----
807 Langkah 100: F_DOWN
808 ABB.EF
809 ACD.EF
810 ACDPPF.
811 GGGH.F
812 ..IHJJ
813 LLIMM.
814 -----
815 Langkah 101: M_LEFT
816 ABB.EF
817 ACD.EF
818 ACDPPF.
819 GGGH.F
820 ..IHJJ
821 LLIMM.
822 -----
823 Langkah 102: H_DOWN
824 ABB.EF
825 ACD.EF
826 ACDPPF.
827 GGGH.F
828 ..IHJJ
829 LLIMM.
830 -----
831 Langkah 103: G_RIGHT
832 ABB.EF
833 ACD.EF
834 ACDPPF.
835 GGGH.F
836 ..IHJJ
837 LLIMM.
838 -----
839 Langkah 104: G_RIGHT
840 ABB.EF
841 ACD.EF
842 ACDPPF.
843 GGGH.F
844 ..IHJJ
845 LLIMM.
846 -----
847 Langkah 105: F_UP
848 ABB.EF
849 ACD.EF
850 ACDPPF.
851 GGGH.F
852 ..IHJJ
853 LLIMM.
854 -----
855 Langkah 106: G_RIGHT
856 ABB.EF
857 ACD.EF
858 ACDPPF.
859 GGGH.F
860 ..IHJJ
861 LLIMM.
862 -----
863 Langkah 107: I_UP
864 ABB.EF
865 ACD.EF
866 ACDPPF.
867 GGGH.F
868 ..IHJJ
869 LLIMM.
870 -----
871 Langkah 108: L_RIGHT
872 ABB.EF
873 ACD.EF
874 ACDPPF.
875 GGGH.F
876 ..IHJJ
877 LLIMM.
878 -----
879 Langkah 109: C_DOWN
880 ABB.EF
881 ACD.EF
882 ACDPPF.
883 GGGH.F
884 ..IHJJ
885 LLIMM.
886 -----
887 Langkah 110: F_DOWN
888 ABB.EF
889 ACD.EF
890 ACDPPF.
891 GGGH.F
892 ..IHJJ
893 LLIMM.
894 -----
895 Langkah 111: M_LEFT
896 ABB.EF
897 ACD.EF
898 ACDPPF.
899 GGGH.F
900 ..IHJJ
901 LLIMM.
902 -----
903 Langkah 112: H_DOWN
904 ABB.EF
905 ACD.EF
906 ACDPPF.
907 GGGH.F
908 ..IHJJ
909 LLIMM.
910 -----
911 Langkah 113: G_RIGHT
912 ABB.EF
913 ACD.EF
914 ACDPPF.
915 GGGH.F
916 ..IHJJ
917 LLIMM.
918 -----
919 Langkah 114: G_RIGHT
920 ABB.EF
921 ACD.EF
922 ACDPPF.
923 GGGH.F
924 ..IHJJ
925 LLIMM.
926 -----
927 Langkah 115: F_UP
928 ABB.EF
929 ACD.EF
930 ACDPPF.
931 GGGH.F
932 ..IHJJ
933 LLIMM.
934 -----
935 Langkah 116: G_RIGHT
936 ABB.EF
937 ACD.EF
938 ACDPPF.
939 GGGH.F
940 ..IHJJ
941 LLIMM.
942 -----
943 Langkah 117: I_UP
944 ABB.EF
945 ACD.EF
946 ACDPPF.
947 GGGH.F
948 ..IHJJ
949 LLIMM.
950 -----
951 Langkah 118: L_RIGHT
952 ABB.EF
953 ACD.EF
954 ACDPPF.
955 GGGH.F
956 ..IHJJ
957 LLIMM.
958 -----
959 Langkah 119: C_DOWN
960 ABB.EF
961 ACD.EF
962 ACDPPF.
963 GGGH.F
964 ..IHJJ
965 LLIMM.
966 -----
967 Langkah 120: F_DOWN
968 ABB.EF
969 ACD.EF
970 ACDPPF.
971 GGGH.F
972 ..IHJJ
973 LLIMM.
974 -----
975 Langkah 121: M_LEFT
976 ABB.EF
977 ACD.EF
978 ACDPPF.
979 GGGH.F
980 ..IHJJ
981 LLIMM.
982 -----
983 Langkah 122: H_DOWN
984 ABB.EF
985 ACD.EF
986 ACDPPF.
987 GGGH.F
988 ..IHJJ
989 LLIMM.
990 -----
991 Langkah 123: G_RIGHT
992 ABB.EF
993 ACD.EF
994 ACDPPF.
995 GGGH.F
996 ..IHJJ
997 LLIMM.
998 -----
999 Langkah 124: G_RIGHT
1000 ABB.EF
1001 ACD.EF
1002 ACDPPF.
1003 GGGH.F
1004 ..IHJJ
1005 LLIMM.
1006 -----
1007 Langkah 125: F_UP
1008 ABB.EF
1009 ACD.EF
1010 ACDPPF.
1011 GGGH.F
1012 ..IHJJ
1013 LLIMM.
1014 -----
1015 Langkah 126: G_RIGHT
1016 ABB.EF
1017 ACD.EF
1018 ACDPPF.
1019 GGGH.F
1020 ..IHJJ
1021 LLIMM.
1022 -----
1023 Langkah 127: I_UP
1024 ABB.EF
1025 ACD.EF
1026 ACDPPF.
1027 GGGH.F
1028 ..IHJJ
1029 LLIMM.
1030 -----
1031 Langkah 128: L_RIGHT
1032 ABB.EF
1033 ACD.EF
1034 ACDPPF.
1035 GGGH.F
1036 ..IHJJ
1037 LLIMM.
1038 -----
1039 Langkah 129: C_DOWN
1040 ABB.EF
1041 ACD.EF
1042 ACDPPF.
1043 GGGH.F
1044 ..IHJJ
1045 LLIMM.
1046 -----
1047 Langkah 130: F_DOWN
1048 ABB.EF
1049 ACD.EF
1050 ACDPPF.
1051 GGGH.F
1052 ..IHJJ
1053 LLIMM.
1054 -----
1055 Langkah 131: M_LEFT
1056 ABB.EF
1057 ACD.EF
1058 ACDPPF.
1059 GGGH.F
1060 ..IHJJ
1061 LLIMM.
1062 -----
1063 Langkah 132: H_DOWN
1064 ABB.EF
1065 ACD.EF
1066 ACDPPF.
1067 GGGH.F
1068 ..IHJJ
1069 LLIMM.
1070 -----
1071 Langkah 133: G_RIGHT
1072 ABB.EF
1073 ACD.EF
1074 ACDPPF.
1075 GGGH.F
1076 ..IHJJ
1077 LLIMM.
1078 -----
1079 Langkah 134: G_RIGHT
1080 ABB.EF
1081 ACD.EF
1082 ACDPPF.
1083 GGGH.F
1084 ..IHJJ
1085 LLIMM.
1086 -----
1087 Langkah 135: F_UP
1088 ABB.EF
1089 ACD.EF
1090 ACDPPF.
1091 GGGH.F
1092 ..IHJJ
1093 LLIMM.
1094 -----
1095 Langkah 136: G_RIGHT
1096 ABB.EF
1097 ACD.EF
1098 ACDPPF.
1099 GGGH.F
1100 ..IHJJ
1101 LLIMM.
1102 -----
1103 Langkah 137: I_UP
1104 ABB.EF
1105 ACD.EF
1106 ACDPPF.
1107 GGGH.F
1108 ..IHJJ
1109 LLIMM.
1110 -----
1111 Langkah 138: L_RIGHT
1112 ABB.EF
1113 ACD.EF
1114 ACDPPF.
1115 GGGH.F
1116 ..IHJJ
1117 LLIMM.
1118 -----
1119 Langkah 139: C_DOWN
1120 ABB.EF
1121 ACD.EF
1122 ACDPPF.
1123 GGGH.F
1124 ..IHJJ
1125 LLIMM.
1126 -----
1127 Langkah 140: F_DOWN
1128 ABB.EF
1129 ACD.EF
1130 ACDPPF.
1131 GGGH.F
1132 ..IHJJ
1133 LLIMM.
1134 -----
1135 Langkah 141: M_LEFT
1136 ABB.EF
1137 ACD.EF
1138 ACDPPF.
1139 GGGH.F
1140 ..IHJJ
1141 LLIMM.
1142 -----
1143 Langkah 142: H_DOWN
1144 ABB.EF
1145 ACD.EF
1146 ACDPPF.
1147 GGGH.F
1148 ..IHJJ
1149 LLIMM.
1150 -----
1151 Langkah 143: G_RIGHT
1152 ABB.EF
1153 ACD.EF
1154 ACDPPF.
1155 GGGH.F
1156 ..IHJJ
1157 LLIMM.
1158 -----
1159 Langkah 144: G_RIGHT
1160 ABB.EF
1161 ACD.EF
1162 ACDPPF.
1163 GGGH.F
1164 ..IHJJ
1165 LLIMM.
1166 -----
1167 Langkah 145: F_UP
1168 ABB.EF
1169 ACD.EF
1170 ACDPPF.
1171 GGGH.F
1172 ..IHJJ
1173 LLIMM.
1174 -----
1175 Langkah 146: G_RIGHT
1176 ABB.EF
1177 ACD.EF
1178 ACDPPF.
1179 GGGH.F
1180 ..IHJJ
1181 LLIMM.
1182 -----
1183 Langkah 147: I_UP
1184 ABB.EF
1185 ACD.EF
1186 ACDPPF.
1187 GGGH.F
1188 ..IHJJ
1189 LLIMM.
1190 -----
1191 Langkah 148: L_RIGHT
1192 ABB.EF
1193 ACD.EF
1194 ACDPPF.
1195 GGGH.F
1196 ..IHJJ
1197 LLIMM.
1198 -----
1199 Langkah 149: C_DOWN
1200 ABB.EF
1201 ACD.EF
1202 ACDPPF.
1203 GGGH.F
1204 ..IHJJ
1205 LLIMM.
1206 -----
1207 Langkah 150: F_DOWN
1208 ABB.EF
1209 ACD.EF
1210 ACDPPF.
1211 GGGH.F
1212 ..IHJJ
1213 LLIMM.
1214 -----
1215 Langkah 151: M_LEFT
1216 ABB.EF
1217 ACD.EF
1218 ACDPPF.
1219 GGGH.F
1220 ..IHJJ
1221 LLIMM.
1222 -----
1223 Langkah 152: H_DOWN
1224 ABB.EF
1225 ACD.EF
1226 ACDPPF.
1227 GGGH.F
1228 ..IHJJ
1229 LLIMM.
1230 -----
1231 Langkah 153: G_RIGHT
1232 ABB.EF
1233 ACD.EF
1234 ACDPPF.
1235 GGGH.F
1236 ..IHJJ
1237 LLIMM.
1238 -----
1239 Langkah 154: G_RIGHT
1240 ABB.EF
1241 ACD.EF
1242 ACDPPF.
1243 GGGH.F
1244 ..IHJJ
1245 LLIMM.
1246 -----
1247 Langkah 155: F_UP
1248 ABB.EF
1249 ACD.EF
1250 ACDPPF.
1251 GGGH.F
1252 ..IHJJ
1253 LLIMM.
1254 -----
1255 Langkah 156: G_RIGHT
1256 ABB.EF
1257 ACD.EF
1258 ACDPPF.
1259 GGGH.F
1260 ..IHJJ
1261 LLIMM.
1262 -----
1263 Langkah 157: I_UP
1264 ABB.EF
1265 ACD.EF
1266 ACDPPF.
1267 GGGH.F
1268 ..IHJJ
1269 LLIMM.
1270 -----
1271 Langkah 158: L_RIGHT
1272 ABB.EF
1273 ACD.EF
1274 ACDPPF.
1275 GGGH.F
1276 ..IHJJ
1277 LLIMM.
1278 -----
1279 Langkah 159: C_DOWN
1280 ABB.EF
1281 ACD.EF
1282 ACDPPF.
1283 GGGH.F
1284 ..IHJJ
1285 LLIMM.
1286 -----
1287 Langkah 160: F_DOWN
1288 ABB.EF
1289 ACD.EF
1290 ACDPPF.
1291 GGGH.F
1292 ..IHJJ
1293 LLIMM.
1294 -----
1295 Langkah 161: M_LEFT
1296 ABB.EF
1297 ACD.EF
1298 ACDPPF.
1299 GGGH.F
1300 ..IHJJ
1301 LLIMM.
1302 -----
1303 Langkah 162: H_DOWN
1304 ABB.EF
1305 ACD.EF
1306 ACDPPF.
1307 GGGH.F
1308 ..IHJJ
1309 LLIMM.
1310 -----
1311 Langkah 163: G_RIGHT
1312 ABB.EF
1313 ACD.EF
1314 ACDPPF.
1315 GGGH.F
1316 ..IHJJ
1317 LLIMM.
1318 -----
1319 Langkah 164: G_RIGHT
1320 ABB.EF
1321 ACD.EF
1322 ACDPPF.
1323 GGGH.F
1324 ..IHJJ
1325 LLIMM.
1326 -----
1327 Langkah 165: F_UP
1328 ABB.EF
1329 ACD.EF
1330 ACDPPF.
1331 GGGH.F
1332 ..IHJJ
1333 LLIMM.
1334 -----
1335 Langkah 166: G_RIGHT
1336 ABB.EF
1337 ACD.EF
1338 ACDPPF.
1339 GGGH.F
1340 ..IHJJ
1341 LLIMM.
1342 -----
1343 Langkah 167: I_UP
1344 ABB.EF
1345 ACD.EF
1346 ACDPPF.
1347 GGGH.F
1348 ..IHJJ
1349 LLIMM.
1350 -----
1351 Langkah 168: L_RIGHT
1352 ABB.EF
1353 ACD.EF
1354 ACDPPF.
1355 GGGH.F
1356 ..IHJJ
1357 LLIMM.
1358 -----
1359 Langkah 169: C_DOWN
1360 ABB.EF
1361 ACD.EF
1362 ACDPPF.
1363 GGGH.F
1364 ..IHJJ
1365 LLIMM.
1366 -----
1367 Langkah 170: F_DOWN
1368 ABB.EF
1369 ACD.EF
1370 ACDPPF.
1371 GGGH.F
1372 ..IHJJ
1373 LLIMM.
1374 -----
1375 Langkah 171: M_LEFT
1376 ABB.EF
1377 ACD.EF
1378 ACDPPF.
1379 GGGH.F
1380 ..IHJJ
1381 LLIMM.
1382 -----
1383 Langkah 172: H_DOWN
1384 ABB.EF
1385 ACD.EF
1386 ACDPPF.
1387 GGGH.F
1388 ..IHJJ
1389 LLIMM.
1390 -----
1391 Langkah 173: G_RIGHT
1392 ABB.EF
1393 ACD.EF
1394 ACDPPF.
1395 GGGH.F
1396 ..IHJJ
1397 LLIMM.
1398 -----
1399 Langkah 174: G_RIGHT
1400 ABB.EF
1401 ACD.EF
1402 ACDPPF.
1403 GGGH.F
1404 ..IHJJ
1405 LLIMM.
1406 -----
1407 Langkah 175: F_UP
1408 ABB.EF
1409 ACD.EF
1410 ACDPPF.
1411 GGGH.F
1412 ..IHJJ
1413 LLIMM.
1414 -----
1415 Langkah 176: G_RIGHT
1416 ABB.EF
1417 ACD.EF
1418 ACDPPF.
1419 GGGH.F
1420 ..IHJJ
1421 LLIMM.
1422 -----
1423 Langkah 177: I_UP
1424 ABB.EF
1425 ACD.EF
1426 ACDPPF.
1427 GGGH.F
1428 ..IHJJ
1429 LLIMM.
1430 -----
1431 Langkah 178: L_RIGHT
1432 ABB.EF
1433 ACD.EF
1434 ACDPPF.
1435 GGGH.F
1436 ..IHJJ
1437 LLIMM.
1438 -----
1439 Langkah 179: C_DOWN
1440 ABB.EF
1441 ACD.EF
1442 ACDPPF.
1443 GGGH.F
1444 ..IHJJ
1445 LLIMM.
1446 -----
1447 Langkah 180: F_DOWN
1448 ABB.EF
1449 ACD.EF
1450 ACDPPF.
1451 GGGH.F
1452 ..IHJJ
1453 LLIMM.
1454 -----
1455 Langkah 181: M_LEFT
1456 ABB.EF
1457 ACD.EF
1458 ACDPPF.
1459 GGGH.F
1460 ..IHJJ
1461 LLIMM.
1462 -----
1463 Langkah 182: H_DOWN
1464 ABB.EF
1465 ACD.EF
1466 ACDPPF.
1467 GGGH.F
1468 ..IHJJ
1469 LLIMM.
1470 -----
1471 Langkah 183: G_RIGHT
1472 ABB.EF
1473 ACD.EF
1474 ACDPPF.
1475 GGGH.F
1476 ..IHJJ
1477 LLIMM.
1478 -----
1479 Langkah 184: G_RIGHT
1480 ABB.EF
1481 ACD.EF
1482 ACDPPF.
1483 GGGH.F
1484 ..IHJJ
1485 LLIMM.
1486 -----
1487 Langkah 185: F_UP
148
```