

Implementasi Algoritma Backtracking dalam Penyelesaian Family of Five Within Thirty Seconds Problem

Fernaldy - 13520112

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13520112@std.stei.itb.ac.id

Abstract—Algoritma *backtracking* merupakan salah satu metode pemecahan masalah yang cukup efisien. Algoritma *backtracking* telah banyak digunakan untuk menyelesaikan berbagai permasalahan yang ada. Dalam makalah ini, algoritma *backtracking* akan dimanfaatkan untuk menyelesaikan permasalahan *family of five within thirty seconds*. Masalah ini merupakan salah satu varian dari *river crossing puzzle*. Dengan menggunakan algoritma *backtracking*, akan ditentukan semua konfigurasi solusi agar lima orang dalam permasalahan tersebut dapat menyeberangi sungai menggunakan kapal yang hanya mampu menampung dua orang penumpang dalam maksimal 30 detik.

Keywords—*backtrack*; *family of five*; *pemangkas*; *river crossing puzzle*; *simpul*

I. PENDAHULUAN

A. Latar Belakang

Puzzle merupakan salah satu permainan yang digemari banyak orang di dunia. Baik anak-anak, orang dewasa, maupun lanjut usia, banyak yang senang bermain *puzzle*. *Puzzle* dianggap sebagai permainan yang mempertajam ingatan, mengasah kemampuan penyelesaian masalah, atau sekadar sarana untuk menghabiskan waktu luang.

Salah satu permainan *puzzle* yang cukup terkenal adalah *river crossing puzzle* yaitu permainan untuk memindahkan sejumlah objek dari satu tepi sungai ke tepi lainnya menggunakan sebuah perahu. Jumlah objek serta batasan-batasan dalam permainan cukup beragam tergantung variasi *river crossing puzzle* yang dimainkan. Tiga variasi awal dari *river crossing puzzle* ini adalah *the fox, goose and bag of beans puzzle*, *the jealous husband problem*, dan *the bridge and torch problem*. *River crossing puzzle* kemudian terus berkembang dan berbagai variasi baru bermunculan. Salah satu variasi tersebut adalah *family of five within thirty seconds*. Dalam makalah ini, penulis akan membahas penerapan algoritma *backtracking* dalam penentuan solusi permasalahan *family of five within thirty seconds*.

B. Tujuan

Berdasarkan latar belakang yang telah dijabarkan di atas, tujuan dari penulisan makalah ini adalah menentukan solusi permasalahan *family of five within thirty seconds* dengan memanfaatkan algoritma *backtracking*.

II. TEORI DASAR

A. Exhaustive Search

Exhaustive search merupakan teknik pencarian solusi atas suatu permasalahan yang bersifat kombinatorik dengan menggunakan pendekatan *brute force*. *Brute force* adalah pendekatan yang *straight forward* dalam pemecahan suatu permasalahan dan didasarkan pada karakteristik persoalan yang dinyatakan. Secara teoritis, *exhaustive search* akan selalu menemukan solusi karena melakukan pencarian terhadap semua kemungkinan solusi, tetapi kelemahan dari *exhaustive search* ini adalah memerlukan waktu dan sumber daya yang sangat besar [1].

Berikut adalah langkah-langkah di dalam *exhaustive search*:

1. Enumerasi semua kemungkinan solusi
2. Evaluasi kemungkinan solusi yang ada satu per satu dan simpan solusi terbaik yang ditemukan sejauh ini
3. Umumkan solusi terbaik

B. Traversal Graf

Traversal graf adalah teknik pencarian solusi dengan mengunjungi simpul secara sistematis. Teknik traversal graf dapat dibagi menjadi dua berdasarkan penentuan urutan simpul yang dikunjungi yaitu pencarian melebar (*breadth first search / BFS*) dan pencarian mendalam (*depth first search / DFS*). Selain itu, dalam proses pencarian, juga terdapat dua representasi graf yang mungkin yaitu graf statis dan graf dinamis. Graf statis adalah graf yang sudah terbentuk sebelum proses traversal dilakukan, sedangkan graf dinamis adalah graf yang terbentuk selama proses traversal dilakukan [2].

Algoritma *breadth first search* (BFS) adalah teknik traversal graf yang mengunjungi simpul-simpul berdasarkan urutan jarak simpul tersebut dengan simpul awal. Algoritma BFS memanfaatkan struktur data *queue* atau antrian untuk menentukan simpul selanjutnya yang akan dikunjungi. Berikut adalah langkah-langkah di dalam *breadth first search* (BFS):

1. Buat suatu antrian
2. Masukkan simpul awal ke antrian
3. Kunjungi simpul yang ada di awal antrian
4. Masukkan semua simpul yang bertetangga dengan simpul yang sedang dikunjungi dan belum pernah dikunjungi sebelumnya ke antrian
5. Ulangi langkah 3 dan 4 sampai antrian kosong [2]

Algoritma *depth first search* (DFS) adalah teknik traversal graf yang mengunjungi simpul-simpul dengan mengutamakan kedalaman terlebih dahulu. Berbeda dari algoritma BFS yang memanfaatkan struktur data *queue* untuk menentukan simpul selanjutnya yang akan dikunjungi, algoritma DFS akan memanfaatkan struktur data *stack* untuk melakukan *backtracking* ketika ada simpul yang belum dikunjungi dan semua tetangga dari simpul yang sedang dievaluasi telah dikunjungi. Berikut adalah langkah-langkah di dalam *depth first search* (DFS):

1. Kunjungi simpul awal
2. Kunjungi simpul yang bertetangga dengan simpul yang sedang dikunjungi dan belum pernah dikunjungi sebelumnya
3. Ulangi langkah 2 hingga tidak ada simpul yang belum dikunjungi [2]

Di dalam *depth first search* (DFS), apabila ada simpul yang belum dikunjungi, tetapi semua simpul tetangga dari simpul yang sedang dikunjungi sudah dikunjungi seluruhnya, maka akan dilakukan *backtrack* ke simpul terakhir yang dikunjungi sebelum simpul saat ini.

Dalam pencarian solusi dengan pendekatan graf dinamis, pencarian solusi akan dimulai dari suatu simpul. Kemudian, setiap simpul akan diperiksa apakah merupakan simpul solusi. Apabila bukan simpul solusi, maka akan dilakukan pembangkitan simpul lain dari simpul saat ini. Pembangkitan simpul dapat menggunakan pendekatan BFS ataupun DFS.

Pencarian solusi dengan algoritma *breadth first search* (BFS) selalu menemukan solusi jika solusi memang ada dan solusi yang ditemukan pasti optimal. Kompleksitas waktu dan kompleksitas ruang *breadth first search* (BFS) adalah $O(b^d)$ dengan b adalah jumlah maksimum cabang dari suatu simpul dan d adalah kedalaman dari solusi terbaik. Sementara itu, pencarian solusi dengan algoritma *depth first search* (DFS) juga selalu menemukan solusi jika solusi memang ada. Namun, solusi yang ditemukan tidak pasti optimal. Kompleksitas waktu dari algoritma *depth first search* (DFS) adalah $O(b^m)$ dan kompleksitas ruangnya adalah $O(bm)$ dengan b adalah jumlah maksimum cabang dari suatu simpul dan m adalah maksimum kedalaman dari ruang status [3].

C. Backtracking

Algoritma *backtracking* merupakan variasi dari algoritma traversal graf dengan pendekatan *depth first search* (DFS) yang lebih baik. Selain itu, algoritma *backtracking* juga dapat dipandang sebagai perbaikan dari algoritma *exhaustive search* karena semua kemungkinan solusi akan dievaluasi pada *exhaustive search*, sedangkan pada algoritma *backtracking*, hanya simpul-simpul yang mengarah ke solusi yang akan dievaluasi. Tiga elemen algoritma *backtracking* adalah sebagai berikut:

1. Solusi persoalan yang dinyatakan sebagai vektor n -tuple
2. Fungsi pembangkit T
3. Fungsi pembatas B yang bernilai *true* jika simpul mengarah ke solusi dan bernilai *false* jika simpul melanggar *constraint* atau batasan [4]

Pembangkitan simpul akan menggunakan pendekatan DFS dan simpul-simpul pada pohon ruang status dapat dibedakan menjadi 3 yaitu simpul hidup (*live node*), simpul-E (*expand node*), dan simpul mati (*dead node*). Simpul hidup (*live node*) adalah simpul-simpul yang sudah dibangkitkan, sedangkan simpul-E (*expand node*) adalah simpul yang sedang diperluas, dan simpul mati (*dead node*) adalah simpul yang tidak mengarah ke solusi.

Dalam penyelesaian masalah, konfigurasi awal persoalan akan direpresentasikan sebagai simpul akar pada pohon ruang status. Anak dari suatu simpul pada pohon ruang status merupakan hasil aplikasi fungsi pembangkit T pada simpul tersebut dan adalah semua konfigurasi persoalan yang mungkin dicapai dari simpul tersebut. Pengecekan kepada setiap simpul dilakukan dengan menerapkan fungsi pembatas B . Apabila fungsi pembatas B menghasilkan nilai *false* pada suatu simpul, maka simpul tersebut akan dijadikan simpul mati dan tidak akan di-*expand* lebih lanjut karena tidak akan mengarah ke solusi. Sementara itu, simpul daun pada pohon ruang status disebut sebagai simpul solusi dan lintasan dari simpul akar ke simpul daun membentuk solusi yang mungkin atas persoalan yang dihadapi.

Berikut adalah langkah-langkah pencarian solusi dengan algoritma *backtracking*:

1. Terapkan fungsi pembatas B pada simpul saat ini. Apabila bernilai *true*, maka lanjut ke langkah ke-2, sedangkan apabila bernilai *false*, simpul tersebut akan dijadikan simpul mati dan simpul-E kembali ke simpul orang tua dan lanjut ke langkah ke-2
2. Periksa simpul apakah adalah solusi persoalan. Apabila bukan solusi persoalan maka bangkitkan salah satu anak dengan menggunakan fungsi pembangkit T dan kembali ke langkah ke-1
3. Proses dilakukan hingga semua solusi telah ditemukan

D. Family of Five within Thirty Seconds Problem

Family of Five within Thirty Seconds Problem adalah salah satu variasi dari *river crossing puzzle*. *River crossing puzzle* pertama kali diperkenalkan oleh Alcuin, seorang lulusan Inggris yang menjadi penasihat dari Charlemagne, seorang Kaisar Romawi Suci. Alcuin mengumpulkan kumpulan teka-tekinya pada sebuah buku berjudul *Propositiones ad Acuendos Juvenes* atau *Problems to Sharpen the Young*.

River crossing puzzle kemudian terus berkembang dan muncul berbagai variasi yang berkaitan dengan *puzzle* tersebut. Dari variasi-variasi tersebut, ada yang memiliki tingkat kesulitan rendah dan ada yang memiliki tingkat kesulitan tinggi sesuai jumlah objek yang perlu dipindahkan dan batasan-batasan yang berlaku dalam pemindahan objek tersebut.

Salah satu variasi yang muncul adalah *family of five within thirty seconds problem*.



Gambar 1. Contoh tampilan permainan *river crossing puzzle a family of five within thirty seconds problem* yang dapat diunduh dari <https://play.google.com/store/apps/details?id=zl.puzzle.riveriq>

Di dalam persoalan tersebut, terdapat sebuah keluarga yang terdiri dari lima orang. Kelima orang tersebut harus dipindahkan dari satu tepi sungai ke tepi sungai lainnya. Pemindahan orang antar tepi sungai dilakukan dengan menggunakan sebuah perahu. Namun, perahu memiliki kapasitas maksimal dua orang dan untuk menggerakkan perahu, dibutuhkan satu orang untuk menakhodainya. Masing-masing orang dari kelima orang tersebut membutuhkan waktu yang berbeda-beda untuk menyeberangi sungai menggunakan perahu. Waktu tersebut diurutkan dari yang terkecil adalah 1 detik, 3 detik, 6 detik, 8 detik, dan 12 detik. Apabila terdapat dua orang di dalam perahu, maka perahu akan sampai di tepi sungai lainnya dalam waktu terbesar antara waktu yang diperlukan dua orang di dalam perahu untuk sampai ke tepi sungai lainnya. Kelima orang tersebut harus mampu dipindahkan dari satu tepi sungai ke tepi sungai lainnya dalam waktu maksimal 30 detik.

III. ANALISIS DAN PEMBAHASAN

Algoritma *backtracking* akan diterapkan untuk menentukan semua solusi atas permasalahan *a family of five within thirty seconds problem*.

A. Deskripsi Persoalan

Berdasarkan deskripsi persoalan yang ada, dapat disimpulkan bahwa tujuan utama persoalan *a family of five*

within thirty seconds problem adalah untuk memindahkan kelima orang yang terdapat di satu tepi sungai ke tepi sungai lainnya. Selain itu, terdapat juga batasan-batasan dalam penyelesaian persoalan *a family of five within thirty seconds problem*. Berikut adalah batasan-batasan (*constraints*) yang terdefinisi dalam permasalahan *a family of five within thirty seconds problem*:

1. Pemindahan orang dari satu tepi sungai ke tepi sungai lainnya hanya dapat menggunakan sebuah perahu.
2. Perahu hanya mampu menampung maksimum dua orang dan perlu ada satu orang yang menakhodainya.
3. Setiap orang memiliki waktu tempuh yang berbeda-beda untuk menyeberangi sungai, yaitu 1 detik, 3 detik, 6 detik, 8 detik, dan 12 detik.
4. Jika terdapat dua orang yang menaiki perahu untuk menyeberangi sungai, maka perahu memerlukan waktu tempuh maksimum antara kedua orang tersebut untuk menyeberangi sungai.
5. Pemindahan kelima orang tersebut harus dilakukan dalam waktu maksimal 30 detik.

B. Pemodelan Persoalan

Layaknya suatu permasalahan yang diselesaikan dengan algoritma *backtracking*, persoalan *a family of five within thirty seconds problem* akan memiliki sekumpulan kondisi yang dimodelkan sebagai simpul pada pohon ruang status. Kondisi yang digambarkan oleh simpul dalam dan daun pada pohon ruang status hanya kondisi ketika perahu sedang berada di tepi sungai tujuan, sedangkan simpul akar akan memodelkan kondisi awal ketika lima orang sedang di tepi sungai awal dengan perahu di tepi sungai tersebut. Simpul pada pohon ruang status untuk persoalan *a family of five within thirty seconds problem* akan berupa *tuple* dengan struktur sebagai berikut:

$$X = ((x_1, x_2, x_3, x_4, x_5), t)$$

Berikut adalah properti terkait struktur *tuple* tersebut:

1. x_1 menyatakan posisi dari orang dengan waktu tempuh 1 detik. Nilai 0 menandakan orang tersebut ada di tepi sungai awal dan nilai 1 menandakan orang tersebut ada di tepi sungai tujuan.
2. x_2 menyatakan posisi dari orang dengan waktu tempuh 3 detik. Definisi nilai sama seperti x_1 .
3. x_3 menyatakan posisi dari orang dengan waktu tempuh 6 detik. Definisi nilai sama seperti x_1 .
4. x_4 menyatakan posisi dari orang dengan waktu tempuh 8 detik. Definisi nilai sama seperti x_1 .
5. x_5 menyatakan posisi dari orang dengan waktu tempuh 12 detik. Definisi nilai sama seperti x_1 .

6. t menyatakan waktu yang telah dihabiskan sejak permainan dimulai.

Berikut adalah elemen-elemen algoritma *backtracking* untuk persoalan *a family of five within thirty seconds problem*:

1. Solusi persoalan

Sesuai dengan tujuan dari permainan *river crossing puzzle* dan struktur simpul yang telah didefinisikan sebelumnya, solusi persoalan *a family of five within thirty seconds problem* adalah simpul dengan struktur:

$$X = ((1,1,1,1,1), t)$$

Struktur solusi tersebut menandakan bahwa kelima orang tersebut harus berada pada tepi sungai tujuan dengan total waktu t sesuai deskripsi persoalan. Selain itu, perlu diperhatikan bahwa nilai t harus lebih kecil atau sama dengan 30 sesuai batasan yang terdefinisi untuk permasalahan *a family of five within thirty seconds problem*.

2. Fungsi pembangkit

Karena hanya kondisi ketika perahu berada di tepi sungai tujuan yang dimodelkan sebagai simpul pada pohon ruang status, harus dilakukan sedikit modifikasi pada fungsi pembangkit.

Aplikasi fungsi pembangkit pada suatu simpul dalam akan terlebih dahulu mengubah nilai salah satu elemen yang telah bernilai 1 menjadi 0. Kondisi ini menandakan bahwa satu orang dari tepi sungai tujuan akan kembali ke tepi sungai awal karena perahu memerlukan seseorang untuk menaikhodainya. Karena simpul akar memodelkan kondisi awal dengan perahu di tepi sungai awal, maka pengubahan nilai 1 menjadi 0 ini tidak perlu dilakukan saat pembangkitan anak dari simpul akar. Selanjutnya, fungsi pembangkit akan mengubah dua buah nilai 0 pada simpul tersebut menjadi 1 yang menggambarkan keadaan bahwa dua orang menyeberangi sungai bersama-sama menggunakan perahu.

Elemen 1 yang dipilih untuk diubah menjadi 0 adalah elemen 1 dengan waktu tempuh terkecil agar waktu yang dibutuhkan untuk kembali ke tepi sungai awal seminimum mungkin.

Pemilihan 2 buah elemen 0 yang diubah menjadi 1 dilakukan dengan mengambil subset berukuran 2 dari himpunan elemen 0 dari simpul yang elemen 1 terkecilnya telah diubah menjadi 0. Sesuai teori kombinatorial, maka kemungkinan jumlah anak dari suatu simpul adalah

$$C(n, 2)$$

dengan n adalah jumlah elemen bernilai 0 setelah perubahan salah satu elemen 1 menjadi 0 jika simpul tersebut adalah simpul dalam.

Berikut adalah contoh penerapan fungsi pembangkit pada salah satu simpul:

$$T((1, 0, 0, 0, 1), 12)$$

Kondisi di atas menggambarkan perahu ada di tepi sungai tujuan dengan orang pertama dan orang kelima telah berada di tepi sungai tujuan. Karena simpul tersebut bukan akar maka perlu mengubah elemen 1 yang waktu tempuhnya terkecil menjadi 0 sehingga kondisi menjadi

$$T((0, 0, 0, 0, 1), 13)$$

Simpul tersebut tidak akan dibuat sebagai simpul pada pohon ruang status karena menggambarkan kondisi ketika perahu ada di tepi sungai awal. Selanjutnya, fungsi pembangkit akan memilih dua nilai 0 acak untuk diubah menjadi 1. Karena nilai 0 ada 4, berarti untuk simpul $((1,0,0,0,1),12)$ mungkin memiliki $C(4,2)$ anak yaitu 6 simpul. Misalnya dipilih elemen kedua dan ketiga untuk diubah nilainya menjadi 1, maka akan dibangkitkan suatu simpul anak dengan konfigurasi

$$X = ((0, 1, 1, 0, 1), 19)$$

3. Fungsi pembatas

Sesuai batasan persoalan *a family of five within thirty seconds problem* yang telah didefinisikan, fungsi pembatas untuk persoalan ini adalah total waktu tempuh lebih kecil atau sama dengan 30. Ketika suatu simpul menghasilkan *false* dari fungsi pembatas, maka simpul tersebut akan dimatikan karena melanggar *constraint* dan tidak mengarah ke solusi.

C. Implementasi Solusi

Implementasi algoritma *backtracking* untuk persoalan *a family of five within thirty seconds problem* menggunakan pendekatan rekursif. Berikut adalah pseudocode yang diimplementasikan:

```

procedure Backtrack(input x: Simpul)
{Mencari semua solusi persoalan dengan metode backtracking
menggunakan skema rekursif}
{Masukan: x sebuah simpul}
{Luaran: semua solusi persoalan}

Algoritma:

if total waktu tempuh <= 30 then
  if node adalah solusi then
    Cetak lintasan dari akar ke simpul x
  else
    Mengubah elemen 1 minimum menjadi 0
    for each subset dengan panjang 2 dari himpunan elemen 0 do
      x1 <- x yang diubah kedua nilai 0 yang terpilih menjadi 1
      Backtrack(x1)

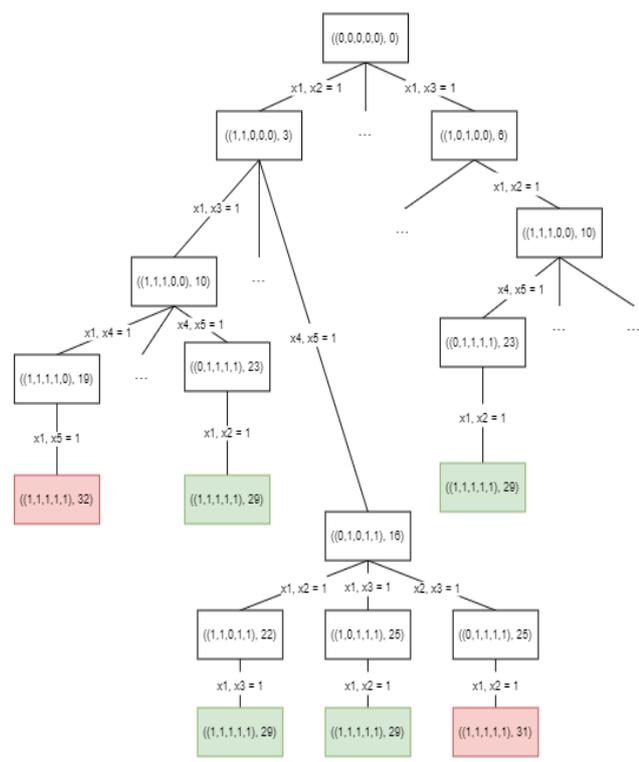
```

Untuk memulai pencarian solusi, diberikan suatu simpul akar yaitu

$$X = ((0,0,0,0,0), 0)$$

yaitu semua orang masih berada di tepi sungai awal dan total waktu tempuh masih 0.

Berikut adalah cuplikan pohon ruang status yang dibangun berdasarkan hasil penelusuran menggunakan algoritma *backtracking*:



Gambar 2. Cuplikan pohon ruang status

Pohon ruang status yang dibuat hanya menggambarkan lintasan ke semua kemungkinan solusi dan beberapa contoh lintasan ke simpul yang dimatikan karena melanggar batasan, bukan untuk menunjukkan keseluruhan pohon ruang status.

Berdasarkan hasil percobaan, semua bagian pohon ruang status yang tidak dimunculkan dalam gambar tidak mengarah ke solusi. Simpul yang diberi warna hijau menunjukkan simpul solusi dan simpul yang diwarnai merah adalah contoh simpul yang melanggar batasan.

Berikut adalah daftar lintasan solusi permasalahan *a family of five within thirty seconds problem*:

1. $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((1,1,1,0,0), 10) \rightarrow ((0,1,1,1,1), 23) \rightarrow ((1,1,1,1,1), 29)$
2. $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((0,1,0,1,1), 16) \rightarrow ((1,1,0,1,1), 22) \rightarrow ((1,1,1,1,1), 29)$
3. $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((0,1,0,1,1), 16) \rightarrow ((1,0,1,1,1), 25) \rightarrow ((1,1,1,1,1), 29)$
4. $((0,0,0,0,0), 0) \rightarrow ((1,0,1,0,0), 6) \rightarrow ((1,1,1,0,0), 10) \rightarrow ((0,1,1,1,1), 23) \rightarrow ((1,1,1,1,1), 29)$

Pada solusi pertama, orang pertama dan kedua akan terlebih dahulu menyeberangi sungai selama 3 detik. Karena belum semua orang menyeberangi sungai, maka orang pertama sebagai orang dengan waktu tempuh terkecil kembali ke tepi sungai awal. Total waktu tempuh saat ini adalah 4 detik. Selanjutnya, orang pertama dan ketiga akan menyeberangi sungai selama 6 detik sehingga total waktu tempuh menjadi 10 detik. Karena belum semua orang menyeberangi sungai, maka orang pertama sebagai orang dengan waktu tempuh terkecil kembali ke tepi sungai awal dan total waktu tempuh menjadi 11 detik. Lalu, orang keempat dan kelima menyeberangi sungai sehingga waktu bertambah 12 detik dan total waktu tempuh menjadi 23 detik. Pada kondisi ini, tersisa orang pertama yang belum menyeberangi sungai sehingga orang kedua sebagai orang dengan waktu tempuh terkecil di tepi sungai tujuan kembali ke tepi sungai awal dan total waktu tempuh bertambah 3 detik menjadi 26 detik. Pada akhirnya, orang pertama dan kedua menyeberangi sungai selama 3 detik dan total waktu tempuh menjadi 29 detik. Karena semua orang telah berada di tepi sungai tujuan dan total waktu tempuh kurang dari atau sama dengan 30, maka permainan berhasil diselesaikan.

Pada solusi kedua, orang pertama dan kedua menyeberangi sungai (total waktu = 3 detik). Lalu, orang pertama kembali diikuti orang keempat dan kelima menyeberangi sungai (total waktu = 16 detik). Selanjutnya, orang kedua kembali diikuti orang pertama dan kedua menyeberangi sungai (total waktu = 22 detik). Kemudian, orang pertama kembali serta orang pertama dan orang ketiga menyeberangi sungai (total waktu = 29 detik).

Pada solusi ketiga, orang pertama dan kedua menyeberangi sungai (total waktu = 3 detik). Lalu, orang pertama kembali diikuti orang keempat dan kelima menyeberangi sungai (total waktu = 16 detik). Selanjutnya orang kedua kembali diikuti orang pertama dan ketiga menyeberangi sungai (total waktu = 25 detik). Kemudian, orang pertama kembali serta orang pertama dan kedua menyeberangi sungai (total waktu = 29 detik).

Pada kemungkinan solusi terakhir, orang pertama dan ketiga menyeberangi sungai (total waktu = 6 detik). Lalu, orang pertama kembali diikuti orang pertama dan orang kedua menyeberangi sungai (total waktu = 10 detik). Selanjutnya, orang pertama kembali diikuti orang keempat dan kelima menyeberangi sungai (total waktu = 23 detik). Kemudian, orang kedua kembali serta orang pertama dan kedua menyeberangi sungai (total waktu = 29 detik).

Salah satu contoh lintasan yang tidak mengarah ke solusi adalah

$$((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((1,1,1,0,0), 10) \rightarrow ((1,1,1,1,0), 19) \rightarrow ((1,1,1,1,1), 32)$$

Pada lintasan tersebut, terlebih dahulu orang pertama dan kedua menyeberangi sungai (total waktu = 3 detik). Lalu, orang pertama kembali diikuti orang pertama dan ketiga menyeberangi sungai (total waktu = 10 detik). Selanjutnya, orang pertama kembali diikuti orang pertama dan keempat menyeberangi sungai (total waktu = 19 detik). Kemudian, orang pertama kembali serta orang pertama dan kelima menyeberangi sungai (total waktu = 32 detik). Meskipun kelima orang telah menyeberangi sungai, total waktu tempuh telah melebihi 30 detik sehingga lintasan tersebut tidak membentuk solusi.

IV. KESIMPULAN

A *family of five within thirty seconds problem* adalah salah satu variasi dari *river crossing puzzle*. Dengan menggunakan algoritma *backtracking*, dapat diperoleh seluruh alternatif solusi untuk menyelesaikan persoalan ini. Secara keseluruhan, terdapat 4 buah solusi untuk permasalahan ini, yaitu lintasan $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((1,1,1,0,0), 10) \rightarrow ((0,1,1,1,1), 23) \rightarrow ((1,1,1,1,1), 29)$, $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((0,1,0,1,1), 16) \rightarrow ((1,1,0,1,1), 22) \rightarrow ((1,1,1,1,1), 29)$, $((0,0,0,0,0), 0) \rightarrow ((1,1,0,0,0), 3) \rightarrow ((0,1,0,1,1), 16) \rightarrow ((1,0,1,1,1), 25) \rightarrow ((1,1,1,1,1), 29)$, dan $((0,0,0,0,0), 0) \rightarrow ((1,0,1,0,0), 6) \rightarrow ((1,1,1,0,0), 10) \rightarrow ((0,1,1,1,1), 23) \rightarrow ((1,1,1,1,1), 29)$.

VIDEO LINK AT YOUTUBE

<https://youtu.be/JyZjwaGqR-M>

REPOSITORY GITHUB

<https://github.com/fernaldy112/AFamilyOfFive>

UCAPAN TERIMA KASIH

Penulis mengucapkan puji syukur kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya, penulis dapat menyelesaikan makalah ini. Penulis juga mengucapkan terima kasih kepada Ibu Masayu Leylia Khodra sebagai dosen kelas 1 mata kuliah IF2211 Strategi Algoritma atas bimbingan selama satu semester. Penulis juga berterima kasih kepada orang tua dan pihak-pihak lain yang tidak disebutkan yang senantiasa memberi dukungan kepada penulis dalam penyelesaian makalah ini. Penulis sadar bahwa makalah ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari pembaca.

REFERENCES

- [1] Munir, Rinaldi. 2022. Algoritma Brute Force Bagian 1. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-\(2022\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2021-2022/Algoritma-Brute-Force-(2022)-Bag1.pdf)
- [2] Munir, Rinaldi. 2022. Breadth/Depth First Search (BFS/DFS) Bagian 1. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/BFS-DFS-2021-Bag1.pdf>
- [3] Munir, Rinaldi. 2022. Breadth/Depth First Search (BFS/DFS) Bagian 2. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/BFS-DFS-2021-Bag2.pdf>
- [4] Munir, Rinaldi. 2022. Algoritma Runut-balik (Backtracking) Bagian 1. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-backtracking-2021-Bagian1.pdf>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 20 Mei 2022



Fernaldy
13520112