

Penyelesaian Flow Free dengan Algoritma BFS dan DFS

Yulianti Oenang 13510059
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
yulianti@students.itb.ac.id

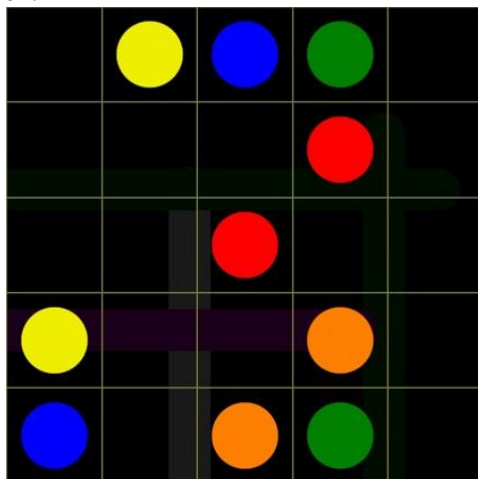
Abstrak—Flow free merupakan game yang terdiri dari sebuah papan yang terdiri dari lingkaran-lingkaran yang memiliki warna yang berpasang-pasangan. Tujuan dari permainan ini adalah menarik garis dari dari sebuah lingkaran berwarna tertentu ke lingkaran lain yang memiliki warna yang sama tanpa ada garis yang saling tumpang tindih dan tidak ada kotak pada papan yang tidak terdapat garis maupun titik. Flow free dapat diselesaikan dengan menggunakan algoritma BFS dan DFS. BFS dan DFS merupakan metode pencarian solusi dengan menggunakan pohon. Perbedaan diantara keduanya adalah BFS pencariannya solusinya dengan melebarkan pohon kemudian memperdalam pohon. Sementara DFS pencarian solusinya dilakukan dengan memperdalam pohon, kemudian melebarkan pohon tersebut.

Index Terms—DFS, BFS, Free Flow, simpul

I. PENDAHULUAN

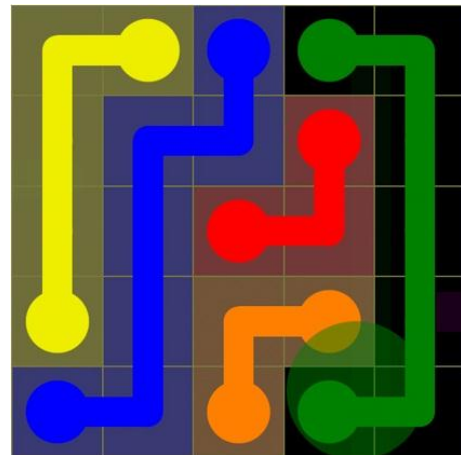
1.1 Flow Free

Flow free adalah game yang bertujuan untuk menyambungkan dua buah titik yang berwarna sama dengan sebuah garis tanpa berpotongan dengan garis lain. Selain itu, kotak kotak yang ada haruslah terisi penuh dengan garis. Berikut gambar keadaan papan sebelum permainan:



Gambar 1 Flow Free sebelum permainan

Berikut gambar keadaan papan sesudah permainan:



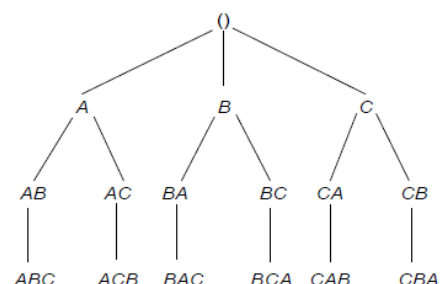
Gambar 2 Flow Free sesudah permainan

Dari titik titik yang ada pada gambar satu, kita dapat tarik garis sesuai dengan keinginan kita dengan tujuan mencapai kondisi seperti gambar kedua, yaitu setiap titik berwarna sama di sambungkan dengan sebuah garis, dan tidak ada kotak yang tidak dilewati garis.

1.2 Ruang Solusi dan Pohon Dinamis

Ruang solusi merupakan seluruh kemungkinan penyelesaian sebuah masalah, dalam hal ini penyelesaian permainan Flow Free. Ruang solusi dapat dibentuk dengan menggunakan pohon yang menyatakan tahapan-tahapan pembentukan solusi. Pohon yang dibentuk untuk mencapai solusi dari sebuah permasalahan disebut pohon dinamis.

Contoh ruang solusi untuk permasalahan permutasi huruf A,B,C adalah ABC, ACB, BCA,BAC,CAB,CBA. Solusi tersebut dapat dicapai dengan membentuk pohon sebagai berikut:



Gambar 3 Pohon Solusi

Ujung dari setiap pohon tersebut disebut daun yang merupakan solusi dari persoalan yang ada. Proses pembentukan pohon dinamis dilakukan dalam dua tahap yaitu BFS dan DFS. Namun pada BFS dan DFS terdapat daun yang mati karena tidak sesuai dengan solusi yang diinginkan.

Terminologi yang sering digunakan dalam BFS dan DFS ini adalah sebagai berikut:

- ✓ Status persoalan adalah simpul simpul yang memenuhi kendala persoalan, misalkan pada kasus flow free, tidak boleh ada garis yang saling tumpang tindih.
- ✓ Status solusi adalah satu atau lebih status yang merupakan solusi persoalan.
- ✓ Status tujuan adalah status solusi yang merupakan simpul daun.
- ✓ Ruang Solusi: Himpunan semua solusi yang mungkin.
- ✓ Ruang Status: Seluruh simpul di dalam pohon dinamis.

1.3 BFS dan DFS

BFS merupakan metode pembentukan pohon dinamis dengan membangkitkan atau menghidupkan semua simpul-simpul selevel. Dari contoh permutasi A, B, C diatas, didapatkan algoritma BFS nya sebagai berikut:

Simpul yang dibangkitkan pertama kali adalah A, B, C. Simpul A membangkitkan simpul AB, AC. Simpul B membangkitkan BA, BC. Simpul C membangkitkan CA, CB. Simpul AB membangkitkan ABC. Simpul AC membangkitkan ACB, simpul BA membangkitkan BAC. Simpul BC membangkitkan BCA. Simpul CA membangkitkan CAB. Simpul CB membangkitkan CBA. Sedangkan DFS merupakan metode pembentukan pohon dinamis dengan membangkitkan atau menghidupkan semua simpul-simpul yang dapat dihidupkan oleh sebuah simpul sampai dicapai solusi yang diinginkan, barulah pembentukan pohon dilakukan ke simpul lain. Dari persoalan permutasi diatas, algoritma DFS yang dihasilkan adalah :

- Simpul awal akan membentuk simpul A
- Simpul A akan membangkitkan simpul AB
- Simpul AB akan membangkitkan simpul ABC
- Pencarian akan mundur sampai ke simpul awal.
- Simpul awal akan membentuk simpul B dan algoritma akan terus dilakukan sampai tidak ada lagi simpul yang bisa dihidupkan.

II. PENYELESAIAN PERSOALAN

2.1 Penyelesaian dengan Menggunakan BFS

Penyelesaian permainan Flow Free dengan menggunakan BFS adalah sebagai berikut:

1. Tentukan urutan titik yang ada pada papan *Flow Free*. Titik yang diurutkan adalah salah satu dari pasangan titik.

2. Tentukan urutan pergerakan. Misalnya ke kanan ke bawah ke kiri keatas. Hal ini berguna apabila pergerakan ke kanan tidak dapat di lakukan, maka pergerakan dilakukan ke bawah. Urutan pergerakan boleh di bolak balik, namun harus konsisten. Dalam makalah ini, urutan pergerakannya adalah ke kanan, ke bawah, ke kiri, ke kanan.
3. Untuk titik pertama cek apakah titik tersebut dapat bergerak ke kanan. Pengecekan dilakukan dengan melihat apakah di kotak sebelah kanan terdapat sebuah titik atau sebuah garis. Jika terdapat maka titik tersebut tidak dapat bergerak ke sebelah kanan.
4. Jika dapat lakukan gerakan pertama (tandai titik disebelah kanan titik pertama sebagai titik yang pernah dikunjungi oleh titik pertama). Ubah posisi titik pertama menjadi posisi titik disebelah titik pertama.
5. Jika tidak dapat melakukan gerakan pertama, cek apakah dapat melakukan gerakan kedua, jika dapat lakukan hal sama seperti nomor empat, hanya yang diperhatikan adalah titik dibawah titik pertama. Jika tidak dapat melakukan gerakan kedua lakukan point ke tiga sampai lima untuk gerakan ketiga. Jika gagal lagi, lakukan point ke tiga sampai ke lima dengan gerakan keempat.
6. Lakukan langkah 3 sampai 5 untuk semua titik yang ada pada papan *Flow Free*
7. Untuk semua titik yang didapat dari langkah 6, lakukan kembali langkah 3 sampai 5 sampai semua titik bertemu dengan pasangan titik.
8. Lakukan pengecekan solusi yang didapat dari langkah 7 dengan mengecek, apakah solusi tersebut sudah membuat papan *Flow Free* terisi dengan garis atau titik.
9. Jika dapat, maka solusi tersebut merupakan solusi permainan *Flow Free* jika tidak maka bukan solusi permainan *Flow Free*.

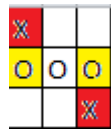
2.2 Penyelesaian dengan Menggunakan DFS

Penyelesaian permainan Flow Free dengan menggunakan DFS adalah sebagai berikut:

1. Tentukan urutan titik yang ada pada papan *Flow Free*. Titik yang diurutkan adalah salah satu dari pasangan titik.
2. Tentukan urutan pergerakan. Misalnya ke kanan ke bawah ke kiri keatas. Hal ini berguna apabila pergerakan ke kanan tidak dapat di lakukan, maka pergerakan dilakukan ke bawah. Urutan pergerakan boleh di bolak balik, namun harus konsisten. Dalam makalah ini, urutan pergerakannya adalah ke kanan, ke bawah, ke kiri, ke kanan.
3. Untuk titik pertama cek apakah titik tersebut dapat bergerak ke kanan. Pengecekan dilakukan dengan melihat apakah di kotak sebelah kanan terdapat sebuah titik atau sebuah garis. Jika

terdapat maka titik tersebut tidak dapat bergerak ke sebelah kanan.

4. Jika dapat lakukan gerakan pertama (tanda titik disebelah kanan titik pertama sebagai titik yang pernah dikunjungi oleh titik pertama). Ubah posisi titik pertama menjadi posisi titik disebelah titik pertama.
5. Jika tidak dapat melakukan gerakan pertama, cek apakah dapat melakukan gerakan kedua, jika dapat lakukan hal sama seperti nomor empat, hanya yang diperhatikan adalah titik dibawah titik pertama. Jika tidak dapat melakukan gerakan kedua lakukan point ke tiga sampai lima untuk gerakan ketiga. Jika gagal lagi, lakukan point ke tiga sampai ke lima dengan gerakan keempat.
6. Lakukan langkah 3 sampai 5 untuk titik yang didapat dari langkah 5 sampai titik tersebut bertemu dengan pasangannya.
7. Jika sebuah titik telah mencapai pasangannya maka lakukan langkah tiga sampai lima untuk semua titik selain titik pertama.
8. Pada langkah 7, jika titik selain titik pertama, tidak menemukan pasangan titiknya karena terblok oleh titik yang telah menemukan pasangannya. Maka perlu dilakukan *backtrack*, artinya titik yang telah menemukan pasangannya itu mundur satu langkah dan mengulang kembali langkah 3 dengan gerakan lain. Contoh nya sebagai berikut:



Gambar 4 Keadaan Backtrack

Titik pertama adalah titik O dengan warna kuning, titik ini melakukan langkah tiga sampai lima sehingga menemukan pasangannya. Namun ketika melakukan langkah ke enam untuk titik X. Titik X tersebut tidak dapat menemukan pasangannya karena sudah terhalang oleh titik O. Oleh sebab itu, titik O harus melakukan *backtrack*, dengan cara, mengubah gerakannya yang terakhir, gerakan terakhirnya adalah kekanan pada posisi O yang tidak berwarna kuning. Gerakan tersebut diganti dengan gerakan selanjutnya yakni bergerak ke bawah. Sehingga titik X dapat menemukan pasangannya.

9. Lakukan pengecekan solusi yang didapat dari langkah 7 dengan mengecek, apakah solusi tersebut sudah membuat papan *Flow Free* terisi dengan garis atau titik.
10. Jika dapat, maka solusi tersebut merupakan solusi permainan *Flow Free* jika tidak maka bukan solusi permainan *Flow Free*.

III KESIMPULAN

Salah satu penggunaan BFS dan DFS adalah untuk menyelesaikan sebuah permainan. BFS dan DFS dapat diterapkan untuk menyelesaikan permainan *Flow Free*.

DAFTAR PUSTAKA

- [1] <http://stackoverflow.com/questions/12926111/what-to-use-for-flow-free-like-game-random-level-creation> Tanggal akses : 21-12-2012.
- [2] Munir, Rinaldi, "Strategi Algoritma", Informatika Bandung, 2009

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, 21 Desember 2012

Yulianti Oenang
13510059