

PENERAPAN ALGORITMA *DEPTH-FIRST SEARCH* (DFS) PADA PENCARIAN SOLUSI PERMAINAN “PAIRS”

Lovinta Happy Atrinawati (13505032)

Program Studi Teknik Informatika Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
Jl. Ganesha No. 10, Bandung
e-mail: if15032@students.if.itb.ac.id

ABSTRAK

Permainan komputer merupakan aplikasi perangkat lunak yang sering digunakan oleh para pengguna komputer. Dengan jenis yang bermacam-macam dan tampilan yang menarik, permainan komputer termasuk perangkat lunak yang diminati oleh berbagai kalangan. Selain karena tampilan dan aplikasinya relatif menarik, permainan komputer dapat menjadi salah satu perangkat lunak yang cukup menyenangkan terutama bagi orang yang telah terbiasa menggunakan komputer.

Salah satu permainan yang sering dimainkan para pengguna komputer adalah permainan *Pairs*. Permainan *Pairs* ini mempunyai peraturan yang cukup sederhana. Pada permainan ini, pemain mencari sepasang kartu yang mempunyai gambar yang sama pada kartu-kartu yang ada. Permainan akan berakhir ketika pemain telah menemukan seluruh pasang gambar yang sama.

Untuk menyelesaikan suatu persoalan, dapat digunakan berbagai macam algoritma. Algoritma *Depth-First Search* (DFS), adalah salah satu jenis algoritma pencarian solusi. Algoritma ini dijalankan dengan cara membangkitkan pohon pencarian secara dinamis. Pencarian solusi dilakukan secara mendalam.

Pada makalah ini, penulis mencoba membahas permainan “*Pairs*” dan menganalisis penerapan algoritma *Depth-First Search* (DFS) sebagai salah satu cara penyelesaian persoalan pada permainan ini sekaligus untuk mengaplikasikan ilmu yang telah didapatkan dari mata kuliah IF2251 Strategi Algoritmik dalam persoalan-persoalan sederhana yang sering ditemui dalam kehidupan sehari-hari.

Kata Kunci : pencarian solusi, *Depth-First Search* (DFS), *Pairs*

1. PENDAHULUAN

Semenjak dahulu, permainan merupakan hal yang sangat menarik bagi sebagian besar masyarakat dunia. Dengan berkembangnya berbagai bentuk permainan, baik berupa permainan yang masih sederhana maupun yang sudah menggunakan teknologi yang canggih, permainan menjadi sangat populer di berbagai tempat.

Permainan *Pairs* merupakan salah satu permainan dari berribu jenis permainan yang ada. Permainan ini adalah permainan yang sederhana dan mudah dimainkan, sehingga menjadi permainan favorit orang banyak. Objektif dari permainan ini adalah pemain harus mencari gambar-gambar yang sepasang. Permainan ini mempunyai tepat satu solusi untuk setiap permainan, namun solusi ini dapat dicari dengan berbagai cara.

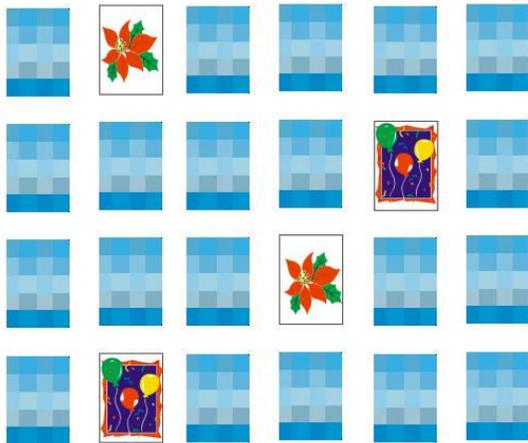
Jenis algoritma yang akan digunakan untuk mencari solusi permainan *pairs* adalah algoritma DFS. Algoritma DFS adalah salah satu jenis algoritma pencarian solusi yang dipelajari pada mata kuliah IF2251 Strategi Algoritmik. Algoritma DFS mencari solusi dari suatu persoalan dengan membangkitkan pohon solusi secara dinamis. Pencarian solusi pada pohon solusi dilakukan secara mendalam.

2. Permainan Pairs

Permainan *Pairs* merupakan permainan yang terdiri dari beberapa kartu bergambar. Gambar pada kartu-kartu tersebut berlainan jenisnya. Satu jenis gambar terdapat pada dua buah kartu, sehingga kartu-kartu tersebut terdiri dari beberapa pasang gambar. Pada permainan *Pairs* ini, kartu-kartu yang ada berjumlah genap karena gambar-gambar yang ada pasti mempunyai pasangan.

Pada awal permainan kartu-kartu tersebut disusun membentuk suatu matriks. Kartu-kartu itu diletakan dalam keadaan tertutup. Pemain memilih dua buah kartu yang ingin dibuka. Jika gambar pada kedua kartu

tersebut tidak sama, maka kartu itu ditutup kembali dan diletakkan di tempat semula. Jika gambar pada kedua kartu tersebut sama, maka kedua kartu tersebut dibiarkan terbuka, Hal ini dilakukan terus oleh pemain hingga pemain telah berhasil menemukan kartu-kartu yang sepasang dan semua kartu telah terbuka.



Gambar 1. Contoh Permainan Pairs

Permainan Pairs hanya mempunyai tepat satu solusi.

3. ALGORITMA DEPTH-FIRST SEARCH (DFS)

Suatu persoalan dapat memiliki lebih dari satu kemungkinan solusi. Sebagian persoalan direpresentasikan dengan mengorganisasikan semua kemungkinan solusi dari persoalan tersebut ke dalam struktur pohon berakar untuk memudahkan pencarian solusinya. Pencarian solusi persoalan dilakukan dengan cara mengunjungi atau menelusuri simpul-simpul dalam pohon solusi. Setiap simpul diperiksa untuk memastikan apakah solusi sudah berhasil ditemukan. Jika sudah ditemukan, maka pencarian solusi dihentikan, tetapi jika belum ditemukan, pencarian dilanjutkan ke simpul berikutnya. Pohon solusi ini dibangkitkan secara dinamis selama pencarian solusi berlangsung. Bila sebuah simpul yang dibentuk tidak mengarah ke solusi, maka pencarian solusi dilanjutkan dengan membentuk simpul berikutnya. Begitu seterusnya sampai solusi ditemukan.

Simpul-simpul di dalam pohon dinamis yang memenuhi kendala (*constraints*) menyatakan status persoalan (*problem state*). Suatu operator mentransformasikan persoalan dari sebuah status ke status yang lain. Solusi persoalan dinyatakan dengan satu atau lebih status yang disebut status solusi (*solution state*). Status solusi yang

merupakan simpul daun disebut status tujuan (*goal state*). Himpunan semua status solusi disebut ruang solusi (*solution space*). Seluruh simpul di dalam pohon dinamis disebut ruang status (*state space*) dan pohonnya dinamakan juga pohon ruang status. Akar pada pohon ruang status menyatakan status awal, sedangkan daun menyatakan status solusi.

Pencarian solusi secara DFS dilakukan dengan ekspansi simpul-simpul terdalam lebih dulu. Pencarian solusi dengan metode DFS pada pohon ruang status digambarkan dengan algoritma berikut ini.

1. Masukkan simpul akar ke dalam antrian Q. Jika simpul akar adalah simpul tujuan, maka solusi telah ditemukan. Pencarian dihentikan.
2. Jika antrian kosong, maka tidak ada solusi. Pencarian dihentikan.
3. Ambil simpul v dari kepala (*head*) antrian. Jika kedalaman simpul sama dengan kedalaman maksimum, kembali ke langkah 2.
4. Bangkitkan semua anak dari simpul v. Jika v tidak mempunyai anak lagi, kembali ke langkah 2. Tempatkan semua anak dari v di awal antrian Q (*addFirst*).
5. Jika anak dari simpul v adalah simpul tujuan, berarti solusi telah ditemukan, kalau tidak kembali lagi ke langkah 2.

Pada kasus terburuk, DFS harus membangkitkan semua simpul pada pohon ruang status untuk menemukan solusi. Secara umum, jika setiap simpul membangkitkan b buah simpul baru, dan batas maksimum kedalaman pohon ruang status adalah $O(b^m)$.

3. PENERAPAN ALGORITMA DFS UNTUK PENCARIAN SOLUSI PERMAINAN PAIRS

Algoritma DFS dapat digunakan untuk mencari solusi pada permainan Pairs. Penggunaan algoritma ini terlihat pada pencarian pasangan dari kartu sampai seluruh kartu telah diperiksa kesamaannya, kemudian baru dilanjutkan dengan kartu berikutnya.

Batasan pada permainan Pairs ini adalah permainan ini selalu mempunyai solusi dan hanya ada satu solusi. Setiap gambar yang ada pada kartu pasti mempunyai pasangan pada kartu yang lain. Dan asumsi penulis, yang menjadi pemain adalah komputer.

Algoritma berikut ini adalah implementasi pencarian solusi untuk permainan pairs dengan bahasa pemrograman pascal.

```

{tipe bentukan dari kartu}
type
  Kartu = record
    gambar : integer;
    open : boolean;
  end;
{integer gambar merepresentasikan
jenis gambar. boolean open bernilai
false jika pasangannya belum
ditemukan, dan true jika sebaliknya}

MKartu : array [1..brs][1..kol] of
Kartu;
{matriks kartu}

procedure cariSolusi(MKartu M)
{mencari solusi dari permainan Pairs}
var
  done : boolean; {true jika selesai}
  i,j,row,col : integer;
begin
  i:=1;
  j:=2;
  row :=1;
  col := 1;
  done := false;
  while (!done)
  begin
    if (!M[row][col].avail)
    begin
      if
(M[row][col].gambar==M[i][j].gambar)
      begin
        M[row][col].avail := true;
        M[i][j].avail := true;
        if (col==kol)
        begin
          row++;
          col:=1;
        end
      else
      begin
        col++;
      end;
    if (j==kol)
    begin
      i++;
      j:=1;
    end
    else
    begin
      j++;
    end;
  end;
  else {gambar tidak sama}
  begin
    if (j==kol)
    begin
      i++;
      j:=1;
    end
  end
end

```

```

else
  begin
    j++;
  end;
end;
else
{pasangan kartu sudah ditemukan}
begin
  if (row==brs and col== kol)
  begin
    done := true;
  end
  else
  begin
    if (j==kol)
    begin
      i++;
      j:=1;
    end
    else
    begin
      j++;
    end;
  end;
end;
end;
end;
end;

```

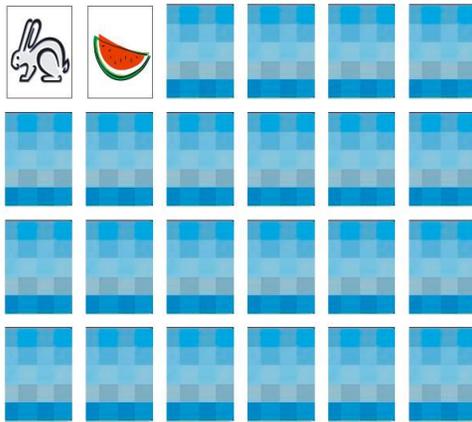
Algoritma DFS untuk permainan Pairs

Pada potongan algoritma dalam bahasa Pascal di atas, terdapat tipe bentukan yang bernama Kartu. Permainan Pairs mempunyai struktur data matriks dari Kartu (*matrix of card*). Kartu ini terdiri dari atribut gambar bertipe integer dan atribut open bertipe boolean. Atribut gambar akan bernilai integer yang merepresentasikan jenis gambar yang ada pada kartu dan bersifat unik. Sedangkan atribut open akan bernilai false jika pasangan gambarnya belum ditemukan, dan bernilai true jika pasangan gambarnya telah ditemukan.

Pemain akan membuka kartu pada baris ke-i kolom ke-j dan kartu pada baris ke-i kolom ke-j+1, kemudian memeriksa apakah gambar pada kedua kartu tersebut sama. Jika sama, maka kedua kartu dibiarkan terbuka dan pemain membuka kartu berikutnya. Jika tidak sama, maka pemain membuka kartu pada baris ke-i kolom ke-j dan kartu pada baris ke-i kolom ke-j+2. Kemudian pemain memeriksa apakah gambar pada kedua kartu tersebut sama. Begitu seterusnya hingga seluruh kartu pada permainan Pairs terbuka.

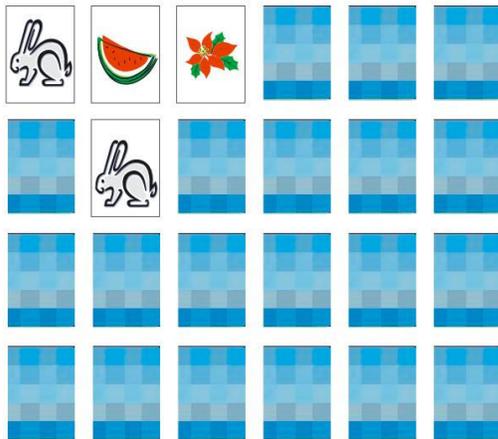
Jalan dari prosedur cariSolusi dapat digambarkan sebagai berikut.

- a. Buka dua kartu.
- b. Periksa apakah gambar kedua kartu tersebut mempunyai gambar yang sama.



Gambar 2.

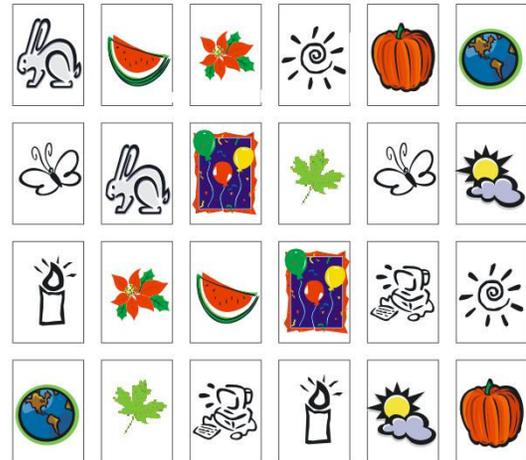
- c. Jika kedua kartu tidak sama, tutup kembali kedua kartu tersebut. Kemudian buka kartu selanjutnya. Jika sama biarkan kartu tersebut terbuka.
- d. Cek kartu berikutnya apakah sudah terbuka atau tidak. Jika masih tertutup, lakukan langkah (a). Jika sudah terbuka, lanjutkan dengan kartu berikutnya kemudian lakukan langkah (a).



Gambar 3. Permainan Pairs selesai.

Pada gambar di atas, telah ditemukan gambar yang sama dari kartu pertama. Kartu tersebut dibiarkan terbuka. Kemudian prosedur dilanjutkan dengan membuka kartu berikutnya.

- e. Prosedur cariSolusi selesai jika sudah mencapai kartu terakhir dan seluruh kartu telah terbuka.



Gambar 4. Permainan Pairs selesai.

3. KESIMPULAN

Algoritma *Depth-First Search* (DFS) merupakan algoritma yang cukup mangkus untuk menyelesaikan berbagai persoalan. Pencarian hanya mengarah pada solusi yang dipertimbangkan saja. Dan pencarian solusi pada algoritma ini dilakukan secara mendalam. Oleh karena algoritma ini cukup efektif, maka DFS banyak diterapkan dalam berbagai program permainan.

Hasil analisis kemampuan algoritma DFS dalam menyelesaikan permainan Pairs menunjukkan bahwa algoritma ini cukup efektif untuk mendapatkan solusi persoalan tersebut.

Pada makalah ini hanya membahas pencarian solusi dengan salah satu algoritma pencarian solusi yaitu DFS. Banyak algoritma-algoritma pencarian solusi lain yang dapat digunakan untuk memecahkan suatu masalah. Pencarian solusi pada permainan Pairs dengan algoritma DFS hanyalah salah satu contoh dari penerapan algoritma.

REFERENSI

- [1] Munir, Rinaldi. Strategi Algoritmik. Teknik Informatika ITB : Bandung. 2007.
- [2] Liem, Inggriani. Catatan Kuliah Algoritma dan Pemrograman. Departemen Teknik Informatika, Institut Teknologi Bandung. 2003.
- [3] Hillier, Frederick S. dan Gerald J. Lieberman. Introduction to Mathematical Programming. McGraw-Hill. 1995.