

Implementasi Pohon Keputusan untuk Membangun Jalan Cerita pada Game Engine Unity

Winarto - 13515061

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13515061@std.stei.itb.ac.id

Abstract—Membangun sebuah permainan digital bukanlah hal yang mudah, banyak aspek yang harus dipertimbangkan. Salah satu aspek paling penting adalah jalan cerita permainan. Salah satu hal yang membuat jalan cerita disukai oleh orang adalah jalan cerita yang bercabang. Makalah ini membahas tentang cara mengimplementasi pohon cerita untuk membangun jalan cerita yang bercabang dengan memanfaatkan *game engine unity*.

Keywords—Permainan, Pohon, Unity, Storyline.

I. PENDAHULUAN

Pada era modern sekarang, teknologi berkembang dengan pesat. Salah satu bidang teknologi tersebut adalah teknologi informasi. Adapun salah satu karya dari bidang tersebut yaitu *game digital*.

Game digital berkembang dengan pesat dari tahun ke tahun, salah satu bukti nyatanya yaitu perbedaan visual *game* pada tahun 2000 dan 2016 yang sangat signifikan. Tentu saja, hal itu dapat terjadi karena perkembangan perangkat keras yang mendukung seperti komputer dan konsol juga berkembang dengan pesat.

Game digital merupakan salah satu wadah untuk menuangkan kreativitas seseorang. Kreativitas tersebut tidak terbatas, bisa mencakup aspek *gameplay*, *graphic*, *art*, *music*, *storyline*, dll.

Di antara sekian banyak *game* yang dirilis di pasar, ada beberapa *game* yang dapat bertahan. Salah satu penyebabnya adalah jalan cerita *game* tersebut. Jalan cerita yang panjang dan bercabang akan meningkatkan minat seseorang memainkan kembali *game* tersebut untuk mengetahui jalan cerita lain yang terdapat di dalam *game* itu.

Pada *game* dengan jalan cerita bercabang, pemain akan menemukan jalan cerita yang berbeda tergantung aksi yang dilakukan oleh pemain. Jadi, aksi yang dilakukan oleh player akan menentukan ke mana arah jalan cerita. Karena jalan cerita berbeda, maka akan memungkinkan pemain untuk mendapatkan akhir cerita yang berbeda pula.

Pohon keputusan dapat digunakan sebagai salah satu cara untuk membangun sebuah jalan cerita yang bercabang pada sebuah permainan.

II. LANDASAN TEORI

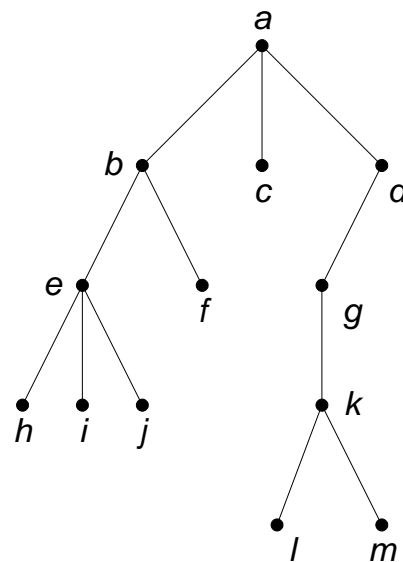
2.1 Pohon

A. Definisi Pohon

Pohon merupakan graf yang khusus atau sering disebut sebagai graf tak-berarah. Pohon terdiri dari pohon bebas dan pohon berarah. Beberapa pohon yang saling lepas akan membentuk hutan yang tidak terhubung dan tidak membentuk sirkuit.

B. Pohon Berakar

Pohon berakar merupakan pohon yang memperlakukan simpul tertentu sebagai akar. Simpul yang ditetapkan sebagai akar dapat memiliki arah yang menunjuk ke simpul lain. Akar merupakan simpul yang memiliki derajat masuk sama dengan nol, berbeda dengan daun yang memiliki keluar sama dengan nol.



Gambar 1 Pohon Berakar. Sumber: Slide kuliah pohon

Pohon memiliki beberapa terminologi sesuai dengan gambar 2 yaitu sebagai berikut:

1. Anak dan Orangtua
Suatu simpul yang ditunjuk oleh simpul lain, simpul yang ditunjuk merupakan anak dari simpul yang menunjuk ke simpul tersebut. Pada gambar 1 simpul b,c,d merupakan anak dari simpul a.
2. Lintasan
Lintasan merupakan runtunan simpul-simpul dari v_1, v_2, \dots, v_n sedemikian sehingga v_k merupakan orangtua dari v_{k+1} . Pada gambar 1 lintasan a ke j adalah a,b,e,j dengan panjang lintasan 3.
3. Keturunan dan Leluhur
Jika terdapat dua simpul dari x ke y, maka simpul x dikatakan sebagai leluhur dari simpul y. Keturunan x yaitu simpul y. Pada gambar 1 dari simpul a ke j, a merupakan leluhur dari j dan j merupakan keturunan dari simpul a.
4. Upapohon
Upapohon merupakan bagian dari sebuah pohon, di mana semua simpul yang ada di upapohon juga terdapat di dalam pohon.
5. Derajat
Derajat merupakan jumlah cabang yang dimiliki oleh sebuah simpul. Pada contoh gambar 1 simpul a memiliki derajat 3.
6. Daun
Simpul yang memiliki derajat keluar sama dengan nol disebut sebagai daun. Simpul m pada gambar 2 merupakan daun.
7. Simpul dalam
Simpul yang memiliki anak dan memiliki orangtua disebut sebagai simpul dalam. Simpul b, c, d, e, f, g, k pada gambar 1 merupakan simpul dalam.
8. Aras
Tingkatan sebuah simpul, dimulai dari 0 yang merupakan aras dari akar.
9. Tinggi
Merupakan aras maksimum sebuah pohon. Pada gambar 1, tinggi pohon adalah 4.

2.2 Unity

Game Engine adalah software yang dirancang untuk mempermudah para pengembang *game* dalam merealisasikan ide mereka. *Game engine* menangani hal-hal yang berhubungan dengan mekanisme game seperti *rendering, physics, input* pemain, *graphical user interface*, dll. *Game engine* mempunyai fitur seperti komponen yang dapat dipakai kembali. Unity merupakan salah satu dari sekian banyak *game engine* yang tersedia di pasar. Unity juga merupakan salah satu *game engine* komersial yang paling *powerful* untuk membuat *game*. Unity merupakan sebuah engine yang *cross-platform* dan memiliki tampilan yang *friendly* sehingga cukup

mudah untuk dipelajari. Harga dari Unity sendiri gratis untuk versi *personal* dan sistem berlangganan untuk versi berbayar. Pemrograman dengan Unity mirip dengan pemrograman berorientasi objek. Setiap benda yang diciptakan di Unity memiliki komponen-komponen. Dengan memanfaatkan komponen-komponen ini, kita dapat membuat sebuah benda yang unik. Komponen ini dapat berupa komponen bawaan dari *game engine*nya sendiri atau dapat juga berupa *source code*.

III. IMPLEMENTASI POHON UNTUK MEMBANGUN JALAN CERITA

A. Komponen Dasar Penentu Jalan Cerita

Komponen dasar yang menentukan jalan cerita *game* adalah aksi yang dilakukan oleh pemain selama *game* berlangsung ataupun di awal permainan. Setiap aksi yang dilakukan oleh pemain memiliki konsekuensi tertentu yang mengakibatkan berubahnya jalan cerita.

Representasi komponen-komponen tersebut di dalam pohon adalah sebagai berikut:

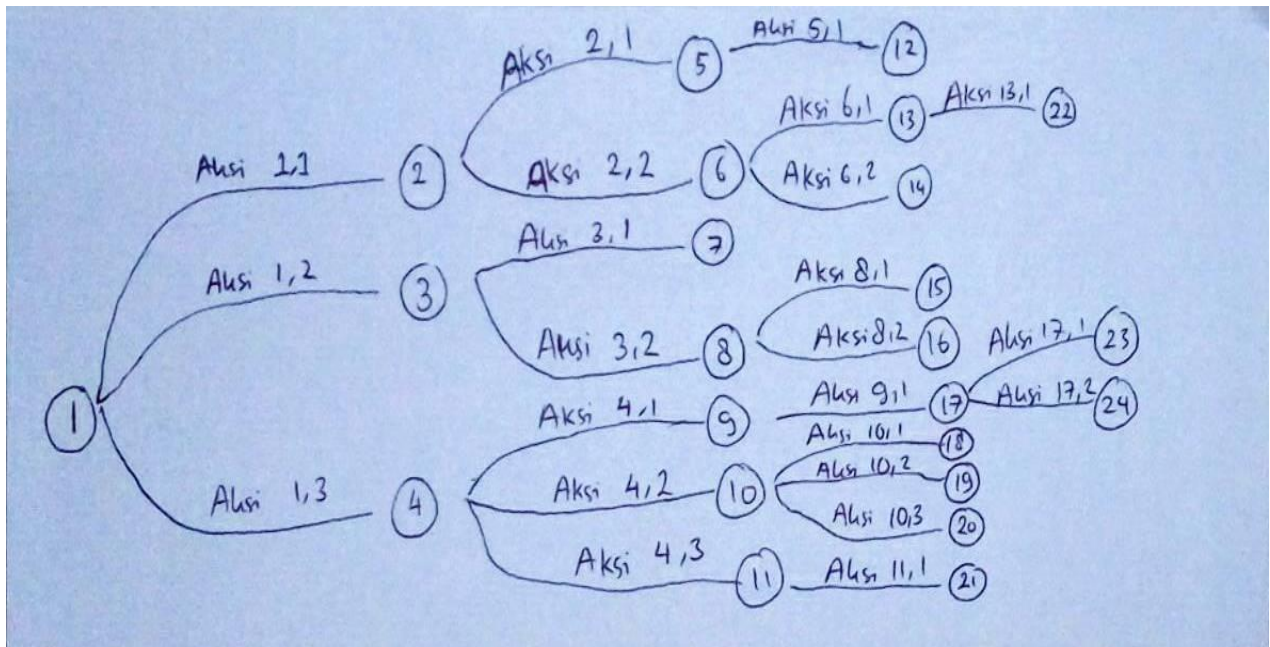
1. Simpul direpresentasikan sebagai kondisi pemain dalam permainan tersebut. Simpul dapat memiliki beberapa sifat yang mirip.
2. Sisi direpresentasikan sebagai aksi yang dilakukan oleh pemain. Jadi setiap aksi akan memindahkan pemain dari suatu kondisi ke kondisi lainnya.

Seiring dengan berjalannya *game*, pemain akan melakukan pilihan atau aksi yang akan membawa pemain ke kondisi lain. Setelah kondisi-kondisi yang dibutuhkan terpenuhi, maka player akan memasuki akhir permainan sesuai dengan kondisi-kondisi yang telah dilewati.

Setelah mencapai akhir dari permainan, pemain tidak dapat melakukan aksi lagi. Dengan kata lain, representasi akhir permainan dalam pohon berupa simpul daun.

B. Menentukan Jalan Cerita Permainan

Contoh dari *game* yang ingin dibuat adalah *visual novel*. Pada *game* jenis ini, komponen dasar yang menentukan arah jalan ceritanya adalah aksi dan kondisi pemain. Aksi di sini berupa opsi yang dipilih pemain ketika melakukan komunikasi dengan *non-player character*. Aksi dan kondisi tersebut akan merepresentasikan sebuah pohon bercabang di mana kondisi sebagai simpul dan kondisi sebagai sisi yang menghubungkan dua buah simpul. Pohon yang memodelkan jalan cerita pada *game* jenis *visual novel* adalah sebagai berikut:

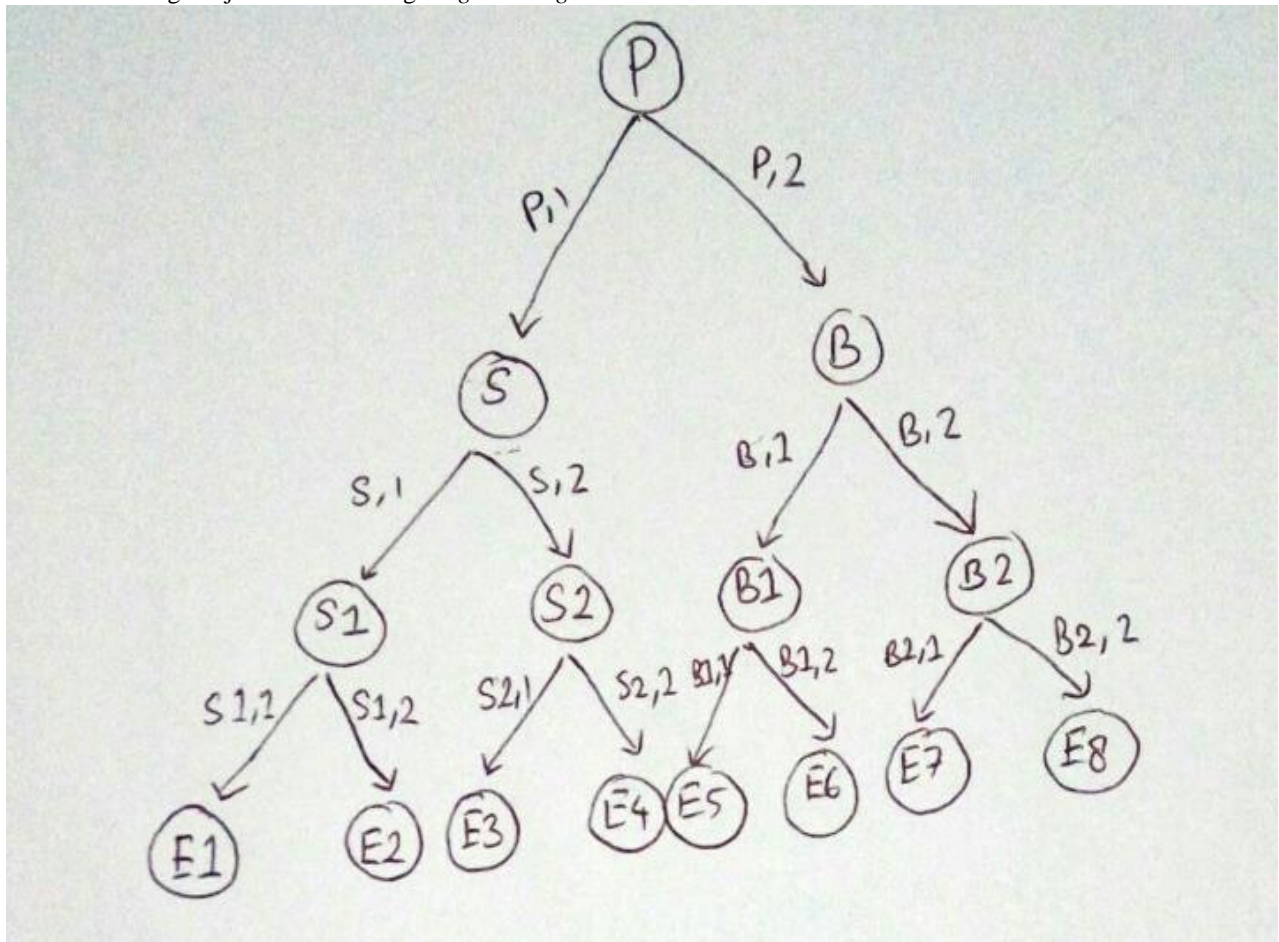


Gambar 2 Contoh pohon memodelkan jalan cerita bercabang

IV. MEMBANGUN JALAN CERITA DENGAN UNITY

Penulis akan menggunakan contoh untuk memberikan ilustrasi membangun jalan cerita dengan *game engine*

Unity. Konsep *game* yang akan dibangun mengambil tema produsen dan konsumen. Pohon yang merepresentasikan keberjalanan cerita adalah sebagai berikut:



Gambar 3 Gambar pohon yang merepresentasi konsep *game* produsen dan konsumen

Pohon pada gambar 3 terdiri dari aksi dan kondisi, di mana kondisi merupakan simpul dan aksi merupakan sisi yang menghubungkan dua simpul. Kondisi dan aksi pada *game* ini terdiri dari:

1. P : Merupakan kondisi awal permainan, di mana pemain akan diberikan opsi untuk memilih bermain sebagai produsen atau konsumen. Untuk memilih, pemain perlu melakukan aksi yang terdiri dari dua pilihan yaitu:
 - P,1: Bermain sebagai produsen
 - P,2: Bermain sebagai konsumen
2. S : Pemain sekarang adalah produsen. Sebagai produsen pemain memiliki beberapa aksi yang dapat dilakukan supaya perusahaan yang dimiliki dapat berkembang. Tetapi apabila pemain salah mengambil tindakan maka perusahaan akan bangkrut. Adapun aksi yang dapat dilakukan pemain sebagai berikut:
 - S,1: Melakukan penelitian untuk menciptakan produk baru.
 - S,2: Melakukan produksi massal produk yang sudah ada.
3. S1 : Pada tahap ini penelitian berhasil dilakukan, perusahaan telah memiliki produk baru yang siap dipasarkan kepada publik. Dengan demikian, pemain diminta untuk melakukan aksi. Aksi terdiri dari dua yaitu:
 - S1,1 : Melakukan produksi massal produk baru dan distribusikan ke pasar.
 - S1,2 : Melanjutkan penelitian untuk menciptakan produk baru.
4. S2 : Produk telah selesai diproduksi dan telah di distribusikan ke beberapa daerah. Kini pemain kembali di minta untuk melakukan aksi selanjutnya, aksi terdiri dari dua:
 - S2,1 : Menghentikan produksi dan menunggu hasil penjualan.
 - S2,2 : Melanjutkan produksi massal dengan harapan mendapat untung besar.
5. B : Kondisi pemain sekarang adalah sebagai konsumen. Sebagai konsumen, tentu membutuhkan beberapa kebutuhan. Pemain diminta melakukan aksi untuk memenuhi kebutuhan konsumen yang terdiri dari:
 - B,1 : Membeli makanan untuk memenuhi kebutuhan primer konsumen.
 - B,2 : Membeli produk yang lagi *nge-trend* di pasaran.
6. B1 : Kebutuhan primer konsumen telah terpenuhi, dan masih tersisa sejumlah uang di dompet konsumen. Pemain diminta melakukan aksi terhadap uang tersebut. Aksi tersebut berupa:
 - B1,1: Menyimpan uang dalam tabungan di bank.
 - B1,2 : Memenuhi kebutuhan sekunder dengan uang

yang tersisa di dompet konsumen.

7. B2: Karena membeli produk keluaran terbaru, sekarang uang konsumen yang tersisa sangat mengkhawatirkan.
 - B2,1: Dengan uang yang tersisa mencoba memenuhi kebutuhan primer.
 - B2,2: Mencoba menjual kembali produk tersebut kepada orang lain.
8. E1: Merupakan kemungkinan akhir kemungkinan pertama, di mana pemain sebagai produsen menghasilkan untung dari penjualan produk baru.
9. E2: Akhir permainan ke-dua, di mana penelitian tersebut berujung pada kegagalan dan tidak membuahkan hasil.
10. E3: Kemungkinan akhir permainan ke-tiga di mana penjualan di pasar buruk sehingga perusahaan merugi.
11. E4: Kemungkinan akhir permainan ke-empat di mana perusahaan bangkrut.
12. E5: Kemungkinan akhir permainan ke-lima di mana uang di dalam bank berbunga dan konsumen dapat memenuhi kebutuhan sehari-hari.
13. E6: Kemungkinan akhir permainan ke-enam di mana konsumen memiliki pakaian baru tetapi tidak mempunyai uang lagi untuk memenuhi kebutuhan lainnya.
14. E7: Kemungkinan akhir permainan ke-tujuh di mana dengan uang yang bersisa, konsumen hanya sanggup membeli satu kotak nasi putih.
15. E8: Kemungkinan akhir permainan ke-delapan di mana tidak ada orang yang tertarik untuk membeli produk tersebut dari konsumen.

Seperti yang sudah dituliskan sebelumnya, dengan memanfaatkan *source code* sebagai komponen dari suatu objek, kita dapat membentuk sebuah jalan cerita sesuai dengan spesifikasi di atas. Realisasi *source code* di dalam Unity adalah sebagai berikut:

```
public struct node {
    public int left;
    public int right;
    public string info;
}
node[] s,b;
node current;
bool isProdusen;
```

Gambar 4 Deklarasi variabel di Unity

Pada gambar 4, merupakan deklarasi variabel untuk pohon keputusan di dalam Unity. Terdapat sebuah tipe bentukan dengan nama *node*, *node* terdiri dari tiga bagian yaitu *left*, *right*, dan *info*. *Left* berfungsi untuk menampung indeks nilai yang menunjuk ke cabang kiri, dan *right* untuk cabang kanan. *String info* menampung informasi kondisi pemain sekarang. Variabel *s* digunakan apabila bermain sebagai produsen dan *b* apabila bermain dengan konsumen. *Current* adalah simpul (kondisi) pemain sekarang. Dan boolean *isProdusen* untuk melakukan

pengecekan apakah pemain bermain sebagai produsen atau konsumen.

```
void Start () {
    s = new node[7];
    b = new node[7];
    s [0].left = 1;
    s [0].right = 4;
    s [1].left = 2;
    s [1].right = 3;
    s [4].left = 5;
    s [4].right = 6;

    b [0].left = 1;
    b [0].right = 4;
    b [1].left = 2;
    b [1].right = 3;
    b [4].left = 5;
    b [4].right = 6;
}
```

Gambar 5 Inisialisasi pohon

Pada gambar 5, merupakan proses pembentukan pohon sesuai dengan gambar 3. Prosedur start otomatis dijalankan di awal permainan.

```
void Pilih(bool kiri){
    if (kiri) {
        current = s [0];
        isProdusen = true;
    } else {
        current = b [0];
        isProdusen = false;
    }
}
```

Gambar 6 Pemilihan antara produsen dan konsumen

Pada gambar 6, jika pemain memilih produsen maka variabel current akan di inisialisasi sebagai upohon produsen. Sebaliknya, jika konsumen yang dipilih maka di inisialisasi sebagai upohon konsumen.

```
void Aksi(bool kiri){
    if (kiri && isProdusen) {
        current = s [current.left];
    } else if (!kiri && isProdusen) {
        current = s [current.right];
    } else if (kiri && !isProdusen) {
        current = b [current.left];
    } else if (!kiri && !isProdusen) {
        current = b[current.right];
    }
}
```

Gambar 7 Prosedur Aksi

Pada gambar 7, menggambarkan prosedur aksi. Prosedur ini akan dipanggil setiap kali pemain melakukan aksi. Fungsi dari prosedur ini adalah memindahkan player dari suatu simpul ke simpul lain tergantung aksi yang dilakukan pemain.

V. SIMPULAN

Salah satu cara untuk membangun *game* dengan jalan cerita yang bercabang yaitu dengan menggunakan pohon keputusan. Untuk membangun *game* juga terdapat banyak cara, salah satunya yaitu menggunakan *game engine* seperti Unity. Jadi, dengan menggunakan pohon keputusan, maka jalan cerita sebuah *game* dapat lebih terstruktur dengan baik.

VI. UCAPAN TERIMA KASIH

Penulis ingin mengungkapkan rasa terima kasih kepada Tuhan Yang Maha Esa, atas kasih dan rahmat-Nya makalah ini dapat terselesaikan. Ucapan terima kasih juga penulis ucapkan kepada Bapak Dr. Ir. Rinaldi Munir, M.T dan ibu Dra. Harlili S., M.Sc. selaku dosen pembimbing mata kuliah IF2120 Matematika Diskrit yang telah memberikan ilmu kepada penulis sehingga makalah bisa terselesaikan.

DAFTAR PUSTAKA

- [1] Munir, Rinaldi. 2006. *Diktat Kuliah IF2120 Matematika Diskrit*. Bandung.
- [2] http://www.gamecareerguide.com/features/529/what_is_a_game.php Diakses tanggal 8 Desember 2016 pukul 20.30.

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, 8 Desember 2016



Winarto, 13515061