

Penerapan Decision Tree dalam memilih GPU

Frederic Ronaldi - 13519134¹

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

¹13519134@std.stei.itb.ac.id

Abstrak—*Graphic Processing Unit* atau yang sering disebut juga *GPU* adalah sebuah perangkat yang memproses grafis dan menampilkan hasilnya pada layar monitor. Terdapat dua jenis *GPU* yaitu *GPU* terintegrasi dan *GPU* diskrit. *GPU* terintegrasi adalah *GPU* yang bersatu dengan *CPU* sehingga kemampuannya lebih rendah daripada *GPU* diskrit. Hingga saat ini, ada dua perusahaan besar yang memproduksi *chip GPU* yaitu Nvidia dan AMD. Dengan adanya banyak model dan generasi *GPU* Nvidia dan AMD, kita dapat menggunakan bantuan pohon keputusan untuk membantu kita dalam pengambilan keputusan dalam memilih model *GPU* yang sesuai dengan kebutuhan kita. Hal itu dikarenakan jika kita asal dalam memilih *GPU*, itu sama halnya dengan mengorbankan uang, waktu, dan juga listrik.

Kata Kunci—AMD, GPU, Nvidia, pohon keputusan

I. PENDAHULUAN

Graphic Processing Unit atau yang sering disebut dengan *GPU* adalah rangkaian elektronika terintegrasi pada silicon wafer yang digunakan sebagai pusat pemrosesan matematika untuk kebutuhan grafis sehingga grafis dapat ditampilkan pada monitor. Berbeda dengan *CPU* yang berfungsi sebagai pusat pemrosesan seluruh kegiatan komputer, *GPU* memproses segala sesuatu yang berbau visual/grafis. Secara sederhananya, *CPU* memproses data dengan jumlah core yang lebih sedikit namun bekerja dengan cepat dan *GPU* memproses data dengan jumlah core yang jauh lebih banyak namun bekerja dengan lambat. Tanpa *GPU*, grafis komputer tak dapat dimunculkan pada layar monitor. Namun, beberapa *CPU modern* saat ini sudah mempunyai *GPU* yang terintegrasi walaupun kecepatannya tidak secepat *GPU* diskrit (terpisah), sehingga tanpa *GPU* diskrit sekalipun, kita masih tetap dapat melihat layar monitor.



Gambar 1.1 GPU

Sumber: <https://www.hellotech.com/blog/whats-a-GPU-what-GPU-do-you-have>

(diakses pada 8 Desember 2020, 19:12)

Jika kita berbicara tentang *GPU*, kita juga harus berbicara tentang gaming karena kedua hal ini paling sering dibandingkan sebagai perbandingan performa suatu *GPU*. *GPU* yang lebih kuat akan membuat game berjalan lebih mulus, hal ini biasanya dihitung dengan *FPS/frame per second*. Semakin tinggi *FPS* dalam suatu game, maka *GPU* tersebut dapat dibilang lebih cepat dalam menjalankan game tersebut. Selain itu, *GPU* juga dapat digunakan dalam hal produktivitas seperti parallel computing, video editing/pembuatan konten dan kecerdasan buatan. Kecerdasan buatan lebih cepat diproses dengan *GPU* karena *GPU* mempunyai core yang sangat banyak sehingga dapat memproses beberapa kalkulasi secara bersamaan serta *GPU* juga mempunyai *Memory*-nya sendiri yang lebih besar dari *CPU* yang dinamakan *VRAM*.

Saat ini, ada dua perusahaan besar yang memproduksi *chip GPU*, yaitu Nvidia dan AMD. Keduanya memiliki ciri khasnya masing-masing dan saling berkompetisi dalam membuat *GPU* yang murah dan kuat. Setelah Nvidia dan AMD membuat *chip GPU* dan *GPU* referensi, *chip* tersebut akan didistribusikan melewati pihak ketiga yang akan memodifikasi kembali *GPU* tersebut menjadi *GPU custom/third party GPU*. Beberapa brand diantaranya adalah *MSI, ASUS, Zotac*, dll.

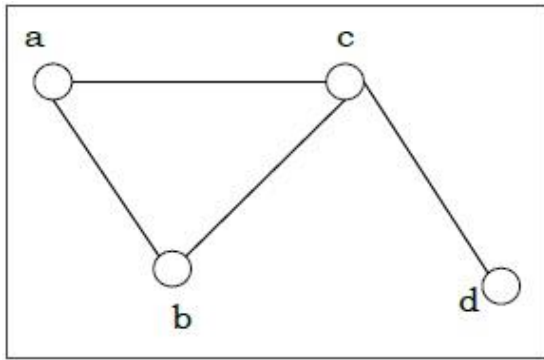
Pada makalah ini penulis akan menerapkan pohon keputusan untuk mendapatkan *GPU* yang cocok dengan kebutuhan kita mau itu *GPU* referensi maupun *GPU* custom. Dengan menggunakan pohon keputusan, kita dapat membandingkan *GPU* yang tersedia dan memperoleh *GPU* yang paling cocok dengan kebutuhan kita sehingga kita tidak menyalahkan waktu, uang dan listrik kita.

II. TEORI DASAR

Pada makalah ini, ada beberapa teori-teori dasar yang diperlukan dalam mengaplikasikan pengambilan keputusan pada pemilihan *GPU* yaitu graf, pohon, dan *GPU* itu sendiri.

1. Graf

Graf adalah sekumpulan objek-objek diskrit yang terdiri atas vertex/simpul dan edge/sisi yang menghubungkan simpul-simpul tersebut. Graf dapat ditulis sebagai pasangan himpunan dan ditulis dengan notasi $G = (V, E)$ dengan G sebagai graf, V sebagai himpunan tidak kosong dari simpul/vertices, dan E sebagai himpunan dari sisi/edges.



Gambar 2.1 Graf dengan 4 simpul dan 4 sisi

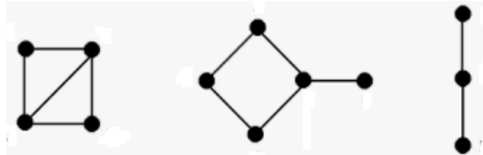
Sumber: https://www.tutorialspoint.com/discrete_mathematics/images/graph.jpg

(diakses pada 9 Desember 2020, 09:04)

Sebuah graf akan dikatakan memiliki sisi ganda jika ada dua sisi yang saling menghubungkan dua buah simpul yang sama. Selain itu, sebuah graf akan dikatakan memiliki gelang/kalang/loop jika sebuah sisi berawal dan berakhir pada simpul yang sama. Berdasarkan ada tidaknya gelang atau sisi ganda pada suatu graf, graf digolongkan menjadi dua jenis:

a. Graf sederhana

Graf yang setiap simpulnya tidak memiliki sisi ganda dan gelang.



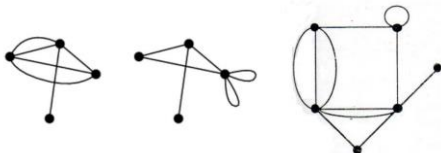
Gambar 2.2 Graf sederhana

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

(diakses pada 9 Desember 2020, 09:20)

b. Graf tak sederhana

Graf yang memiliki setidaknya satu buah sisi ganda atau gelang. Graf tak sederhana dibagi lagi menjadi dua jenis, yaitu graf ganda dan graf semu. Graf ganda yaitu graf yang memiliki sisi ganda, sedangkan graf semu adalah graf yang memiliki sisi gelang.



Gambar 2.2 Graf tak sederhana

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

(diakses pada 9 Desember 2020, 09:20)

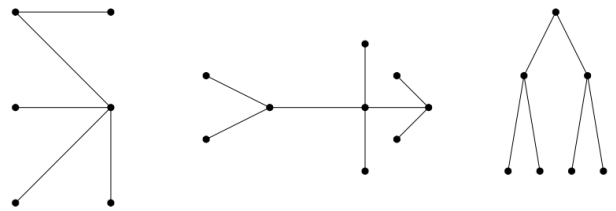
Sebuah graf juga dapat memiliki orientasi/arah yang berarti setiap sisi dalam graf tersebut memiliki simpul awal dan simpul akhir. Graf yang demikian disebut dengan graf berarah sedangkan graf yang sisinya tidak mempunyai orientasi/arah disebut graf tak berarah. Untuk membedakan antara graf berarah dan graf tidak berarah, setiap sisi dalam

graf berarah digambarkan dengan tanda panah menuju simpul akhirnya.

Ada beberapa istilah lain yang penting dalam graf beberapa diantaranya adalah dua simpul dikatakan bertetangga jika keduanya terhubung langsung melalui sisi, derajat merupakan jumlah sisi yang dimiliki oleh sebuah simpul, upagraf/subgraf adalah himpunan elemen graf beserta relasinya yang merupakan bagian dari graf tersebut, lintasan merupakan sisi-sisi yang dilalui untuk mencapai suatu simpul dari sebuah simpul, sirkuit adalah sebuah graf yang memiliki jalan untuk kembali ke titik asal, dan bobot adalah besaran dari suatu sisi.

2. Pohon

Pohon adalah graf tak-berarah yang semua simpulnya terhubung dan tidak mengandung sirkuit. Kumpulan pohon yang saling lepas disebut hutan, setiap komponen di dalam graf terhubung tersebut adalah pohon. Setiap simpul dalam suatu pohon saling terhubung dengan lintasan tunggal sehingga jika terdapat n simpul dalam suatu pohon, sisi yang terbentuk adalah $n-1$ sisi.

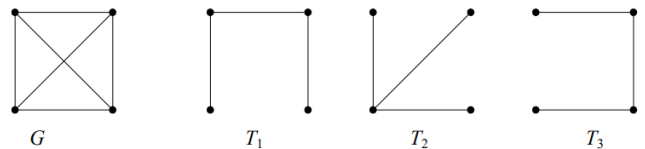


Gambar 2.3 Hutan dengan 3 komponen pohon

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>

(diakses pada 9 Desember 2020, 09:30)

Pohon merentang adalah suatu upagraf merentang yang berupa pohon juga. Maka dari itu, pohon merentang juga dapat diperoleh dengan cara memutus sirkuit yang terbentuk di dalam graf dan setiap graf terhubung pasti mempunyai paling sedikit satu buah pohon merentang. Pohon merentang minimum adalah pohon merentang yang memiliki jumlah bobot pada setiap sisinya paling minimum.

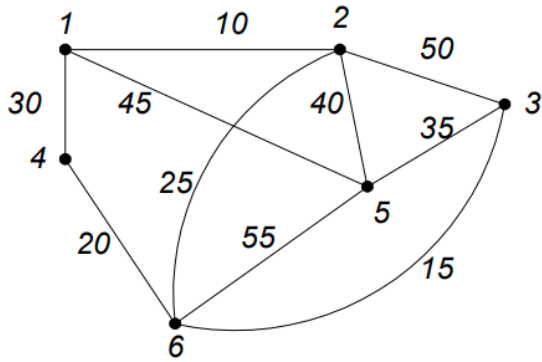


Gambar 2.4 Pohon merentang dari Graf G

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>

(diakses pada 9 Desember 2020, 09:30)

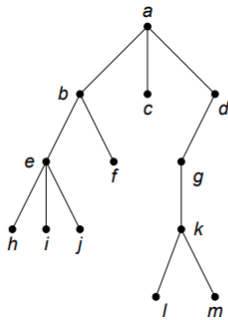
Dalam kehidupan sehari-hari, kita dapat menggunakan pohon merentang minimum dalam banyak hal, contoh sederhananya adalah menentukan jumlah ruas jalan seminimum mungkin yang menghubungkan semua kota sehingga setiap kota tetap terhubung satu sama lain. Dalam mencari pohon merentang minimum dalam suatu graf ada dua algoritma yang sering dipakai, yaitu algoritma prim dan kruskal.



Gambar 2.5 Graf yang berbobot

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>
(diakses pada 9 Desember 2020, 09:30)

Pohon berakar adalah pohon yang setiap simpulnya mempunyai nilai atau akar dan sisi-sisinya diberi arah sehingga menjadi graf berarah. Walaupun merupakan graf berarah, pohon berakar sering kali digambar tanpa tanda panah.



Gambar 2.6 Pohon berakar

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>
(diakses pada 9 Desember 2020, 09:30)

Ada beberapa istilah penting dalam pohon berakar, diantaranya adalah:

1. Anak (children) dan orang tua (parent)
Sesuai dengan namanya, orang tua/parent adalah simpul yang lebih tinggi daripada anak-anaknya, contohnya dalam gambar 2.6, a adalah orang tua, sedangkan b,c, dan d adalah anak-anak simpul a.
2. Lintasan
Lintasan adalah sisi-sisi yang harus dilalui untuk mencapai simpul b dari simpul a. Contohnya lintasan a ke j dalam gambar 2.6 adalah a,b,e, dan j. Panjang lintasan dari a ke j adalah 3.
3. Saudara kandung
Saudara kandung adalah simpul-simpul yang memiliki orang tua yang sama dengan simpul tersebut selain simpulnya sendiri. Contohnya dalam gambar 2.6, f adalah saudara kandung e.
4. Upapohon (subtree)
Upapohon adalah subtree dari pohon berakar. Subtree dari sebuah pohon berakar merupakan sebuah pohon berakar pula.
5. Derajat

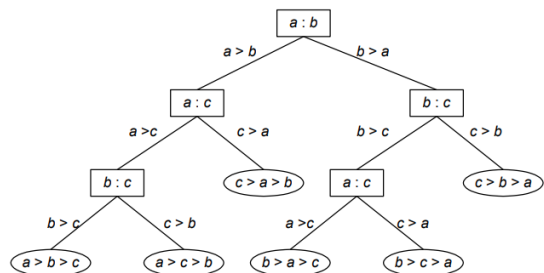
Derajat sebuah simpul adalah banyaknya upapohon pada simpul tersebut. Contohnya, derajat dari simpul a pada gambar 2.6 adalah 3 dan derajat dari simpul d adalah satu.

6. Daun
Daun adalah simpul yang berderajat nol atau simpul yang berada di paling bawah pohon. Pada gambar 2.6, simpul h,i,j,f,c,l dan m adalah daun.
7. Simpul dalam
Simpul yang memiliki anak adalah simpul dalam. Pada gambar 2.6, simpul b,d,e,g, dan k adalah simpul dalam.
8. Aras (level) atau tingkat
Aras/level adalah tingkatan yang dihitung dari akar yang paling atas dan dimulai dari nol. Pada gambar 2.6, level dari a adalah nol, level dari b, c dan d adalah 1, level dari e,f, dan g adalah 2, dan seterusnya.
9. Tinggi (height) atau kedalaman (depth)
Kedalaman adalah tingkatan maksimum yang dimiliki oleh suatu pohon. Pada gambar 2.6, pohon tersebut memiliki kedalaman sebesar 4 tingkat.

Selain pohon berakar, ada pula beberapa jenis pohon lain. Salah satunya adalah pohon terurut yang merupakan pohon berakar yang urutan anak-anaknya penting. Ada pula, pohon n-ary yang merupakan pohon yang setiap simpulnya memiliki paling banyak n anak. Maka dari itu, pohon biner adalah pohon yang setiap simpulnya memiliki paling banyak 2 anak. Anak kiri dan anak kanan pada pohon biner dibedakan sehingga pohon biner termasuk dalam pohon terurut. Pohon biner dapat digunakan dalam banyak kebutuhan misalnya adalah untuk membentuk pohon ekspresi, pohon keputusan, kode awalan, kode Huffman, dan pohon pencarian biner.

3. Pohon keputusan

Seperti yang dipaparkan diatas, pohon keputusan adalah salah satu aplikasi dari pohon biner. Pohon keputusan dapat membantu kita dalam pengambilan suatu keputusan terhadap persoalan yang ada. Pohon keputusan ini dapat bekerja dengan cara saling membandingkan parameter-parameter yang tersedia untuk masing-masing keputusan yang akan diambil. Setiap simpul pada pohon keputusan adalah suatu keputusan, sedangkan setiap daun pada pohon keputusan adalah solusi dari persoalan yang dimodelkan. Suatu keputusan akan semakin dekat kepada solusi untuk setiap level yang dilewati.



Gambar 2.7 Pohon keputusan

Sumber: <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Pohon-2020-Bag1.pdf>
(diakses pada 9 Desember 2020, 09:30)

III. PEMBAHASAN

Baru-baru ini, Nvidia dan AMD mengeluarkan model-model *GPU* dari generasi terbaru. Model-model *GPU* dari Nvidia dan AMD dari generasi lama sudah tidak relevan jika dipilih setelah generasi baru sudah dikelarkan, sehingga model-model *GPU* lama akan dieliminasi dalam makalah kali ini. Dalam makalah kali ini, saya sudah memilih enam model *GPU* generasi terbaru dari Nvidia dan AMD yang akan dijadikan contoh dalam makalah ini. Keenam model *GPU* itu adalah Nvidia GeForce RTX 3090, Nvidia GeForce RTX 3080, Nvidia GeForce RTX 3070, AMD Radeon RX 6900XT, AMD Radeon RX 6800XT, dan AMD Radeon RX 6800. Keenam model yang dipilih pada kali ini adalah *GPU* referensi atau *GPU* yang berasal langsung dari pabrik Nvidia dan AMD dan bukan third party/custom.

Untuk dapat memanfaatkan pohon keputusan dengan baik, diperlukan parameter-parameter tambahan untuk masing-masing *GPU* sebagai penentu keputusan. Berikut adalah spesifikasi terkait untuk model-model *GPU* yang disebutkan diatas:

1. Nvidia GeForce RTX 3090



Gambar 3.1 Nvidia GeForce RTX 3090

Sumber: <https://GPUcheck.com/>

Boost Clock: 1.70 GHz

Memory: 24GB GDDR6X

Average FPS: 117.9 FPS

Power: 350 Watt

Harga: 1499 USD / 21,2 juta rupiah

2. Nvidia GeForce RTX 3080



Gambar 3.2 Nvidia GeForce RTX 3080

Sumber: <https://GPUcheck.com/>

Boost Clock: 1.71 GHz

Memory: 10GB GDDR6X

Average FPS: 105.0 FPS

Power: 320 Watt

Harga: 699 USD / 9,9 juta rupiah

3. Nvidia GeForce RTX 3070



Gambar 3.3 Nvidia GeForce RTX 3070

Sumber: <https://GPUcheck.com/>

Boost Clock: 1.73 GHz

Memory: 8GB GDDR6

Average FPS: 80.8 FPS

Power: 220 Watt

Harga: 499 USD / 7,1 juta rupiah

4. AMD Radeon RX 6900XT



Gambar 3.4 AMD Radeon RX 6900XT

Sumber: <https://GPUcheck.com/>

Boost Clock: up to 2.0 GHz

Memory: 16GB GDDR6

Average FPS: 105.9 FPS

Power: 300 Watt

Harga: 999 USD / 14,2 juta rupiah

5. AMD Radeon RX 6800XT



Gambar 3.5 AMD Radeon RX 6800XT

Sumber: <https://GPUcheck.com/>

Boost Clock: up to 2.0 GHz

Memory: 16GB GDDR6

Average FPS: 100.1 FPS

Power: 300 Watt

Harga: 649 USD / 9,2 juta rupiah

6. AMD Radeon RX 6800



Gambar 3.6 AMD Radeon RX 6800

Sumber: <https://GPUcheck.com/>

Boost Clock: up to 1.8GHz

Memory: 16GB GDDR6

Average FPS: 79.7 FPS

Power: 250 Watt

Harga: 579 USD / 8,2 juta rupiah

Catatan: Untuk perhitungan *FPS/frame per second* dilakukan dengan *CPU* yang sama untuk masing-masing *GPU* dan pada game yang sama sehingga performa *GPU* dapat terlihat dengan lebih jelas.

Dari spesifikasi diatas, ada beberapa istilah yang perlu kita mengerti terlebih dahulu. Boost clock adalah kecepatan maksimum yang dapat dijalankan oleh *GPU* selama dalam keadaan normal. *GPU* juga memiliki *Memory* sendiri yang cara kerjanya serupa dengan RAM bahkan memiliki kecepatan lebih tinggi daripada RAM. Average *FPS* adalah rata-rata yang *FPS/frame per second* didapatkan dalam game. *Power* adalah daya maksimum yang dapat digunakan oleh *GPU* untuk menjalankan fungsinya dengan baik secara normal, baik itu untuk *chipnya* maupun pendinginnya.

Jika berbicara tentang memilih suatu *GPU* yang cocok untuk kita, itu sangat bergantung dengan kebutuhan dan untuk apa kita menggunakan *GPU* tersebut. Untuk performa dalam gaming saja, hal tersebut sudah tergambar jelas pada spesifikasi “Average *FPS*”. Untuk produktivitas, kita membutuhkan 3 parameter untuk dibandingkan yaitu driver, boost clock dan *Memory*. Untuk model *GPU* dari AMD, driver untuk produktivitas masih belum terlalu dikembangkan, sehingga untuk hal produktivitas, Nvidia masih berada jauh diatas AMD.

Dari keenam model *GPU* tersebut, akan dipilih satu *GPU* yang akan saya gunakan, saya akan menggunakan pohon keputusan untuk membantu saya dalam memilih *GPU* yang paling cocok untuk saya. Ada beberapa parameter yang sudah saya tetapkan sebelum memilih *GPU* yang paling cocok untuk saya, yaitu:

1. Saya akan menggunakan *GPU* ini untuk bermain game dan juga produktivitas. Produktivitas yang akan dilakukan tidak terlalu berat.
2. Saya ingin menggunakan *GPU* tersebut dalam gaming juga dan performa gamenya diatas 60 *FPS*.
3. *Power supply* yang saya gunakan tidak dapat menampung *Power* yang sangat besar. Pada saat ini, *Power supply* saya hanya dapat menampung maksimal 300 watt untuk *GPU*.
4. Harga dari *GPU* tersebut harus terjangkau agar harga/performanya tinggi.

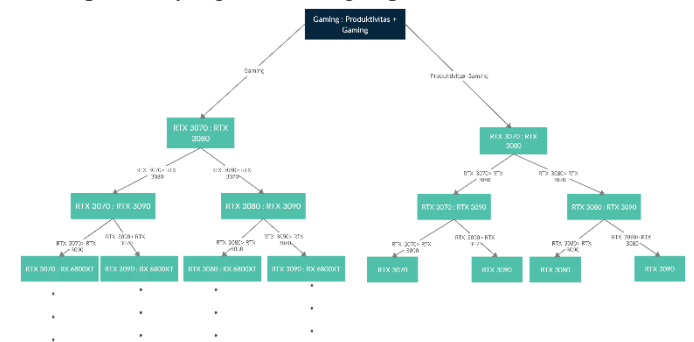
Dengan menggunakan pohon keputusan, berdasarkan parameter diatas, kita akan mencari dan mengeliminasi satu per satu model *GPU* yang kurang cocok dengan saya hingga tersisa satu model *GPU* yang paling cocok. Ada beberapa langkah untuk merancang pohon keputusan yang sesuai dengan permasalahan kita, yaitu tentang pemilihan model *GPU* yang paling cocok dengan kita.

1. Pertama, kita akan membandingkan kepentingan utama pembeli dalam membeli *GPU*, yakni untuk gaming saja atau *GPU* tersebut akan digunakan juga dalam urusan produktivitas. Jika *GPU* tersebut digunakan untuk produktivitas maka model-model *GPU* Nvidia lebih disarankan dan kita dapat mengeliminasi model-model *GPU* AMD.
2. Selanjutnya akan diambil sembarang dua model *GPU* dari model-model *GPU* yang belum tereliminasi dan akan dibandingkan masing-masing spesifikasinya

apakah sesuai dengan parameter yang sudah ditetapkan sebelumnya.

3. Parameter pertama, jika pembeli akan memakai *GPU* tersebut untuk gaming, maka parameter pertama yang wajib dibandingkan adalah parameter “Average *FPS*”. Lihat apakah pembeli sudah menetapkan target *FPS* minimum yang dia ingin capai. Jika salah satu dari model *GPU* tersebut tidak mencapai, maka *GPU* tersebut tereliminasi dan kita bisa lanjut ke langkah 6. Jika kedua model tersebut mempunyai “Average *FPS*” yang serupa, maka kita dapat membandingkan parameter kedua dan melanjutkan ke langkah keempat agar lebih yakin.
4. Parameter kedua, komputer tidak akan berjalan tanpa *Power supply* dan salah satu komponen yang akan menggunakan daya tersebut adalah *GPU*, karena itu kita akan membandingkan daya maksimum *Power supply* kita dengan *GPU* yang sedang dibandingkan. Jika salah satu dari model *GPU* tersebut tidak mencapai target, maka model *GPU* tersebut dieliminasi dan kita dapat lanjut ke langkah 6. Jika kedua model tersebut mempunyai daya maksimum yang serupa, maka kita dapat membandingkan parameter ketiga dan melanjutkan ke langkah kelima agar lebih yakin.
5. Parameter ketiga, harga adalah satu parameter yang penting dalam segala permasalahan. Harga *GPU* yang lebih mudah pasti akan dipertahankan dan yang lebih mahal akan dieliminasi. Setelah salah satu *GPU* tereliminasi maka kita dapat lanjut ke langkah 6. Jika kedua model tersebut mempunyai harga yang serupa, maka kedua model *GPU* tersebut mempunyai tingkatan yang serupa, kita dapat memilih salah satu dan melanjutkan ke langkah keenam.
6. Ulangi langkah kedua hingga kelima untuk setiap model *GPU* yang belum tereliminasi hingga didapatkan model *GPU* terakhir yang belum tereliminasi. Model *GPU* itulah yang menjadi solusi dari permasalahan kita. Model *GPU* itu adalah model *GPU* yang paling cocok dengan kita.

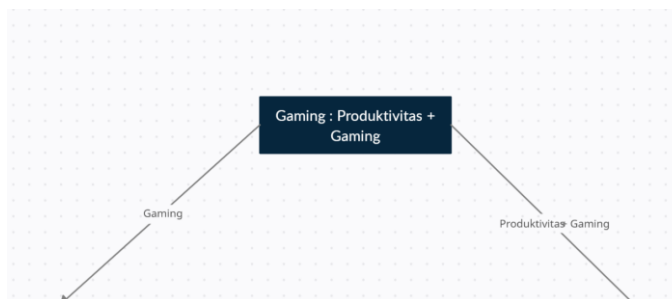
Mengikuti langkah-langkah diatas, maka kita dapat membuat pohon keputusan yang sesuai dengan permasalahan ini.



Gambar 3.7 Pohon keputusan permasalahan pemilihan model *GPU* yang paling cocok (subtree kiri disederhanakan)

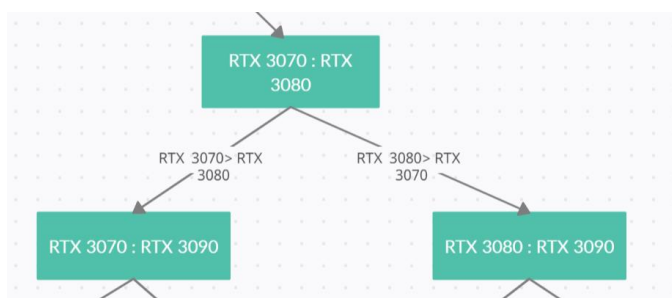
Dengan adanya pohon keputusan tersebut, kita akan menggunakan pohon keputusan tersebut untuk membantu saya

dalam memilih *GPU* yang paling cocok untuk saya. Pertama, kita akan membandingkan kepentingan utama saya dalam menggunakan *GPU*.



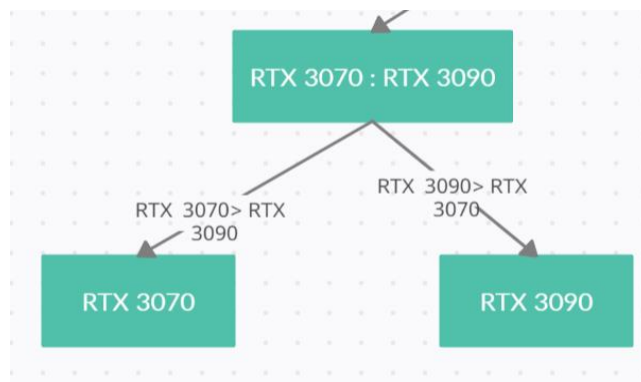
Gambar 3.8 Tahap pertama

Berdasarkan paparan terkait parameter yang dibutuhkan, maka saya akan memilih subtree kanan yaitu produktivitas dan gaming. Pilihan ini akan mengeliminasi subtree sebelah kanan yang terdapat solusi untuk model-model *GPU* AMD, sehingga seluruh model *GPU* AMD tereliminasi karena tidak mendukung produktivitas. Selanjutnya akan dilanjutkan dengan perbandingan dua model-model *GPU* tersisa.



Gambar 3.9 Tahap kedua

Selanjutnya adalah membandingkan *GPU* Nvidia RTX 3070 dengan Nvidia RTX 3080. Pertama kita akan melihat parameter pendukungnya yang pertama yaitu “Average FPS”. Terlihat bahwa kedua *GPU* ini sudah memenuhi FPS minimal yang saya butuhkan yaitu 60 FPS sehingga kedua *GPU* ini setara dalam parameter pertama ini. Kita juga akan kembali membandingkan untuk parameter pendukung kedua, yaitu daya maksimum. Terlihat bahwa spesifikasi *GPU* Nvidia RTX 3080 memiliki daya maksimum yang lebih besar daripada daya maksimum yang dapat ditampung oleh Power supply saya sehingga sesuai dengan langkah-langkah yang sebelumnya ditulis, kita dapat mengeliminasi Nvidia RTX 3080 dan menuju pada subtree kiri/langkah ketiga yaitu kembali membandingkan Nvidia RTX 3070 dengan Nvidia RTX 3090.



Gambar 3.10 Tahap ketiga

Kita akan melihat kembali parameter pendukungnya. Mulai dari yang pertama yaitu “Average FPS”. Kedua *GPU* ini memiliki “Average FPS” yang melebihi dari kebutuhan saya yaitu 60, sehingga kita akan kembali untuk membandingkan parameter pendukung kedua yaitu daya maksimum. Terlihat bahwa spesifikasi dari *GPU* Nvidia RTX 3090 memiliki daya maksimum yang lebih besar daripada daya maksimum yang dapat ditampung oleh Power supply saya sehingga Nvidia RTX 3090 juga dieliminasi dan kita akan menuju pada subtree kiri.

Karena kita sudah mencapai daun pada pohon keputusan, maka kita sudah dapat menarik solusi pada permasalahan ini. Model *GPU* yang paling cocok untuk saya adalah gunakan adalah Nvidia RTX 3070. Model *GPU* ini memenuhi parameter yang paling cocok untuk saya.

Dengan menggunakan prinsip yang sama, permasalahan terkait pemilihan model *GPU* juga dapat diselesaikan. Model-model *GPU* pilihan dan parameter minimal dapat diganti agar sesuai dengan kebutuhan.

IV. KESIMPULAN

Dalam kehidupan sehari-hari, kerap kali kita sulit dalam memutuskan sesuatu, apalagi jika hal tersebut berkaitan dengan hal yang kita tidak mengerti seluruhnya. Dengan menggunakan pohon keputusan, permasalahan seperti itu dapat diselesaikan dengan lebih mudah dan dicari solusi terbaiknya. Sebagai contoh, permasalahan dalam pemilihan model *GPU* yang paling cocok dapat diselesaikan dengan pohon keputusan. Dalam memilih model *GPU* kita memerlukan spesifikasi singkat untuk masing-masing modelnya sehingga kita dapat membandingkan antar model *GPU*. Dalam memilih model *GPU*, ada beberapa parameter penting yang menjadi tolak ukur suatu *GPU*, yaitu FPS, TDP/daya maksimum, dan harga.

V. UCAPAN TERIMA KASIH

Pertama-tama penulis ingin mengucapkan terima kasih kepada Tuhan Yang Maha Esa atas berkatNya, saya dapat menyelesaikan makalah ini dengan baik. Kemudian saya ingin berterima kasih kepada Dra. Harlili, M.Sc. selaku dosen mata kuliah IF2120 Matematika Diskrit yang telah mengajari saya materi ini dengan sangat baik. Terakhir saya ingin berterima kasih kepada keluarga serta teman-teman yang sudah menyemangati saya dan memberi banyak masukan dalam proses pembuatan makalah ini. Saya berharap dengan adanya makalah

ini dapat menambah wawasan kita semua.

REFERENSI

- [1] <https://www.wepc.com/tips/what-does-GPU-mean/>, diakses pada tanggal 8 Desember 2020, 08.18
- [2] Munir, Rinaldi, Slide Perkuliahan IF2120 Graf (Bagian1) dan Pohon (Bagian 1 dan 2), diakses pada tanggal 9 Desember 2020, 09.20.
- [3] <https://www.hellotech.com/blog/whats-a-GPU-what-GPU-do-you-have>, diakses pada tanggal 9 Desember 2020, 10.12
- [4] <https://GPUcheck.com/>, diakses pada tanggal 9 Desember 2020, 21.39
- [5] <https://www.wepc.com/tips/what-does-GPU-mean/>, diakses pada tanggal 10 Desember 2020, 08.18
- [6] <https://www.nvidia.com/en-us/geforce/graphics-cards/30-series/>, diakses pada tanggal 10 Desember 2020, 18.17
- [7] <https://www.amd.com/en/graphics/amd-radeon-rx-6000-series>, diakses pada tanggal 10 Desember 2020, 18.20

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, 11 Desember 2020



Frederic Ronaldi 13519134