1. **Pertemuan 11**

Program Bilangan Bulat

1. Teknik Branch and Bound dan latihan soal

Metode Branch and Bound adalah sebuah teknik algoritma yang secara khusus mempelajari bagaimana caranya memperkecil Search Tree menjadi sekecil mungkin.

Sesuai dengan namanya, metode ini terdiri dari 2 langkah yaitu :

* 1. Branch yang artinya membangun semua cabang tree yang mungkin menuju solusi.
	2. Bound yang artinya menghitung node mana yang merupakan active node (E-node) dan node mana yang merupakan dead node (D-node) dengan menggunakan syarat batas constraint (kendala).

FIFO Branch and Bound

* 1. Adalah teknik Branch and Bound yang menggunakan bantuan queue untuk perhitungan Branch and Bound secara First In First Out.

LIFO Branch and Bound

* 1. Adalah teknik Branch and Bound yang menggunakan bantuan stack untuk perhitungan Branch and Bound secara Last In First Out.

Least Cost Branch and Bound

* 1. Teknik ini akan menghitung cost setiap node. Node yang memiliki cost paling kecil dikatakan memiliki kemungkinan paling besar menuju solusi.

 Langkah-langkah metode Branch dan Bound dapat dilakukan seperti berikut :

1. Selesaikan LP dengan metode simpleks biasa
2. Teliti solusi optimumnya. Jika variabel basis yang diharapkan bulat adalah bulat, solusi optimum bulat telah tercapai.
3. Nilai solusi pecah yang layak dicabangkan ke dalam sub-sub masalah. Tujuannya adalah untuk menghilangkan solusi kontinyu yang tidak memenuhi persyaratan bulat dalam masalah itu.
4. Untuk setiap sub-masalah, nilai solusi optimum kontinyu fungsi tujuan ditetapkan sebagai batas atas. Solusi bulat terbaik menjadi batas bawah (***pada awalnya***, ini adalah solusi kontinyu yang dibulatkan ke bawah). Sub-sub masalah yang memiliki batas atas kurang dari batas bawah yang ada, tidak diikut sertakan pada analisa selanjutnya. Suatu solusi bulat layak adalah sama baik atau lebih baik dari batas atas untuk setiap sub masalah yang dicari. Jika solusi yang demikian terjadi, suatu sub masalah dengan batas atas terbaik dipilih untuk dicabangkan. Kembali ke **langkah 3**.

**Penetapan Batas *(Bounding)***

     Pada algoritma *branch and bound*terdapat dua batas yaitu batas atas (upper bound) dan batas bawah (lower bound).

1. Pada masalah maksimisasi:

Batas atas merupakan solusi ILP relaksasi dari sub masalah tersebut sedangkan batas bawahnya adalah nilai dari sub masalah tersebut ataupun solusi dari sub masalah lain yang semua variabel keputusan yang harus bernilai integer sudah bernilai integer (solusi**terbaik**yang **sejauh ini**diperoleh).

1. Pada masalah minimisasi:

Batas bawah merupakan solusi ILP relaksasi dari sub masalah tersebut sedangkan batas atasnya adalah nilai dari sub masalah tersebut ataupun solusi dari sub masalah lain yang semua variabel keputusan yang harus bernilai integer sudah bernilai integer ( solusi terkecil (**terbaik)**yang **sejauh ini**diperoleh ).

**Penghentian Pencabangan *(Fathoming)***

   Pencabangan atau pencarian solusi pada suatu sub masalah dihentikan jika:

1. *Infeasible*atau tidak mempunyai daerah layak.
2. Semua variabel keputusan yang harus bernilai integer sudah bernilai integer
3. Pada masalah maksimisasi, penghentian pencabangan pada suatu sub masalah dilakukan jika batas atas dari sub masalah tersebut tidak lebih besar atau sama dengan batas bawah.
4. Sedangkan pada masalah minimisasi penghentian pencabangan pada suatu sub masalah dilakukan jika batas bawah tidak lebih lebih kecil atau sama dengan batas atas.

**Kondisi Optimal**

Kondisi optimal pada Branch and bound antara lain :

* 1. Jika tidak ada lagi sub masalah yang perlu dicabangkan lagi maka solusi optimal sudah diperoleh.
	2. Pada masalah maksimisasi solusi optimal merupakan solusi submasalah yang saat ini menjadi batas bawah *(lower bound)*
	3. Pada masalah minimisasi solusi optimal merupakan solusi submasalah yang saat ini menjadi batas atas *(upper bound).*
1. Latihan Soal

Sebuah perusahaan manufaktur elektronik “The Flash” memproduksi 2 buah produk kipas angin dan lampu gantung. Tiap-tiap produk tersebut membutuhkan 2 tahapan produksi, yaitu penyolderan (perakitan komponen elektronik) dan assembling (perakitan komponen non-elektronik) penyolderan membutuhkan waktu 2 jam untuk lampu dan 3 jam untuk kipas angin, sedangkan assembling membutuhkan waktu 6 jam untuk lampu dan 5 jam untuk kipas angin. Perusahaan tersebut hanya mempunyai waktu untuk penyolderan 12 jam dan assembling 30 jam kerja per minggu-nya. Bila lampu gantung memberikan keuntungan sebanyak Rp. 7000 dan Kipas angin memberikan keuntungan Rp. 6000 per unit, formulasi keputusan produksi perusahaan The Flash adalah sebagai berikut:

Maksimisasi profit =      7X1 + 6X2

Ditujukan pada:              2X1 + 3X2 ≤ 12

                                        6X1 + 5X2 ≤ 30

                                          X1, X2 ≥ 0

X1 = Lampu

X2 = Kipas Angin

Dengan metode linear programming dapat kita hitung bahwa solusi optimal dari The Flash adalah memproduksi  Lampu dan  Kipas Angin. Kita menyadari bahwa perusahaan tidak bisa membuat dan menjual barang dalam bentuk pecahan, jadi kita memutuskan bahwa kita menghadapi permasalahan integer programming / pemrograman bulat.