

**EVALUASI MANDIRI OPTIMISASI
MATAKULIAH OPTIMISASI
MENGENALI OPTIMISASI INTEGER**



Disusun oleh:
SHARA ALYA GIFANI MUHYISUNAH
G1D021038

Dosen Pengampu:
Ir. Novalio Daratha S.T., M.Sc., Ph.D.

**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS BENGKULU
2024**

Optimasi Integer adalah salah satu cabang dalam optimasi matematis yang berfokus pada pemecahan masalah di mana variabel keputusan hanya dapat mengambil nilai bilangan bulat (integer), bukan bilangan real (kontinu). Dalam masalah optimasi integer, tujuan umumnya adalah untuk memaksimalkan atau meminimalkan suatu fungsi objektif yang dibatasi oleh sejumlah kendala, dengan syarat bahwa solusi yang dicari terdiri dari nilai-nilai bulat.

Karakteristik Optimasi Integer:

1. Variabel Integer: Semua atau sebagian dari variabel keputusan dalam model optimasi dibatasi untuk mengambil nilai bilangan bulat. Ini sering digunakan ketika keputusan yang diambil berhubungan dengan jumlah unit diskrit, seperti jumlah produk yang diproduksi, jumlah kendaraan yang digunakan, atau jumlah pekerja yang dipekerjakan.
2. Fungsi Tujuan: Seperti pada masalah optimasi lainnya, dalam optimasi integer juga terdapat fungsi tujuan yang ingin dimaksimalkan atau diminimalkan. Fungsi ini bisa berupa biaya, keuntungan, waktu, atau metrik lainnya.
3. Kendala Linear atau Non-Linear: Kendala dalam optimasi integer bisa berupa persamaan atau pertidaksamaan yang bisa bersifat linear (Linear Integer Programming - LIP) atau non-linear (Nonlinear Integer Programming - NIP)..

CODE PROGRAM

```
using JuMP
```

```
using GLPK
```

```
# Membuat model optimisasi
```

```
model = Model(GLPK.Optimizer)
```

```
# Mendefinisikan variabel keputusan
```

```
@variable(model, y1 >= 0, Int) # Produk X
```

```
@variable(model, y2 >= 0, Int) # Produk Y
```

```
# Menambahkan fungsi tujuan  $4y_1 + 6y_2$ 
```

```
@objective(model, Max, 4 * y1 + 6 * y2)
```

```
# Menambahkan kendala
```

```
@constraint(model, 3 * y1 + 2 * y2 <= 120) #  $3y_1 + 2y_2 \leq 120$ 
```

```
@constraint(model, 2 * y1 + 3 * y2 <= 90) #  $2y_1 + 3y_2 \leq 90$ 
```

```
# Menyelesaikan model
```

```
optimize!(model)
```

```
# Menampilkan hasil
```

```
println("Status: ", termination_status(model))
```

```
println("Jumlah Produk X yang diproduksi: ", value(y1))
```

```
println("Jumlah Produk Y yang diproduksi: ", value(y2))
```

```
println("Keuntungan maksimum: ", objective_value(model))
```

Variabel keputusan yang bertipe integer: Pada kode ini, variabel keputusan y_1 dan y_2 didefinisikan sebagai variabel integer dengan perintah `@variable(model, y1 >= 0, Int)` dan `@variable(model, y2 >= 0, Int)`. Ini berarti nilai variabel y_1 dan y_2 hanya dapat berupa bilangan bulat (integer), bukan bilangan pecahan atau kontinu. Hal ini menjadikan model ini sebagai Integer Linear Programming (ILP).

Fungsi objektif dan kendala linear: Meskipun ini adalah masalah optimasi integer, fungsi tujuan dan kendala yang ada tetap linear, yang menjadikannya Linear Integer Programming (LIP). Fungsi tujuan $4y_1 + 6y_2$ dan kendala $3 * y_1 + 2 * y_2 \leq 120$ serta $2 * y_1 + 3 * y_2 \leq 90$ adalah bentuk linear, tetapi karena variabel keputusan adalah integer, model ini tetap merupakan masalah optimasi integer.