

Penerapan Algoritma *Greedy* dalam Meraih IP tertinggi dengan Waktu yang Terbatas

Rizki Halasan / 13515095

*Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13515095@std.stei.itb.ac.id*

Abstrak-- Masa-masa kuliah adalah salah satu masa yang paling penting dalam kehidupan mahasiswa. Pada masa ini, pada umumnya mahasiswa lebih diberi kebebasan dibandingkan pada saat mereka masih menjadi siswa. Dengan diberikannya kebebasan ini, mahasiswa dituntut untuk bertanggung jawab terhadap aktivitas yang dilakukannya. Pembagian waktu menjadi sangat penting bagi mahasiswa untuk meraih tujuan yang ia cita-citakan. Pada setiap mata kuliah disediakan sebuah indikator keberhasilan mahasiswa dalam mengerti mata kuliah tersebut, yang disebut Indeks Prestasi (IP). IP tersebut tentu perhatian bagi mahasiswa dan hampir semua mahasiswa menginginkan IP yang bagus. Dengan menggunakan algoritma Greedy, penulis akan mengilustrasikan cara meraih IP tertinggi untuk perkuliahan satu semester dengan waktu yang terbatas.

Kata Kunci--IP, SKS, Waktu total, Algoritma Greedy,

I. PENDAHULUAN

Bisa berkuliah adalah salah satu nikmat terbesar yang bisa dialami oleh mahasiswa. Pada kuliah ini, terdapat sangat banyak ilmu yang bisa didapat, baik berupa ilmu akademik yang diberikan di kelas maupun eksplorasi dan ilmu organisasi yang diberikan melalui organisasi yang ada di kampus. Oleh karena ilmu yang tersedia sangat banyak, mahasiswa dapat menginvestasikan waktu yang mereka miliki untuk memilih ilmu yang ingin didapatkan selama berkuliah.

Sistem perkuliahan sangatlah berbeda dengan sistem sekolah pada saat mahasiswa tersebut masih menjadi siswa. Apabila di sekolah siswa sudah diberikan pelajaran yang harus diikuti beserta jadwal pelajarannya, di perkuliahan mahasiswa seringkali dapat memilih mata kuliah dengan waktu

perkuliahannya. Selain itu, pada saat berkuliah mahasiswa sudah dianggap dewasa sehingga apabila mahasiswa tidak masuk kuliah ataupun tidak mengerjakan tugas mahasiswa tersebut tidak ditanya oleh dosen pengajarnya. Hal ini sangat berbeda dengan sekolah, apabila siswa tidak masuk ataupun tidak mengerjakan tugas guru akan memberikan penanganan khusus seperti memarahi siswa tersebut maupun memanggil orang tua mahasiswa tersebut.

Oleh karena mahasiswa sudah dianggap dewasa dan dapat mengurus dirinya secara maksimal, maka mahasiswa harus membagi waktu dengan seoptimal mungkin. Alangkah baiknya mahasiswa pada saat berkuliah tidak mengalokasikan waktunya hanya untuk belajar, melainkan juga mengalokasikan sebagian waktunya untuk berorganisasi, bermain, dan berinteraksi dengan warga sekitar. Tuntutan-tuntutan ini membuat waktu mahasiswa untuk belajar semakin terbatas.

Pada sistem perkuliahan terdapat indikator yang menilai keberhasilan mahasiswa terhadap mata kuliah tersebut. Indikator itu dinamakan Indeks Prestasi (IP). IP tersebut adalah hasil akumulasi dari nilai-nilai yang didapatkan mahasiswa dari metode-metode penilaian dari mata kuliah tersebut, seperti kuis, ujian, absen, dan lain-lain. Metode-metode penilaian dari setiap mata kuliah berbeda satu sama lainnya sehingga mahasiswa sebaiknya mengetahui sistem penilaian pada setiap mata kuliah untuk meraih indeks terbaik pada mata kuliah tersebut.

Selain IP, terdapat pula Sistem Kredit Semester (SKS) pada setiap matakuliah yang menjadi bobot dari mata kuliah tersebut. Menurut Buku Peraturan Akademik & Kemahasiswaan Institut Teknologi Bandung 2015, satu SKS beban akademik Program Sarjana setara dengan upaya mahasiswa sebanyak 3

jam seminggu dalam satu semester reguler yang meliputi 1 jam kegiatan tatap muka di kelas, 1 jam kegiatan terstruktur yang dilakukan dalam rangka kegiatan kuliah, dan 1 jam kegiatan mandiri.



Gambar 1. Suasana Perkuliahan, sumber : <http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2015-2016/Foto4.jpg>

Pada kenyataannya, mata kuliah yang mempunyai SKS yang sama belum tentu memiliki beban riil yang sama. Sebagai contoh, hampir semua (apabila tidak semua) mahasiswa STEI ITB merasakan bahwa untuk memahami mata kuliah PAR (Pengantar Analisis Rangkaian), dibutuhkan waktu lebih banyak dibandingkan dengan mata kuliah TTKI (Tata Tulis Karya Ilmiah) walaupun bobot kedua matakuliah tersebut sama (2 SKS). Pada kasus tersebut, tidaklah salah apabila mahasiswa berusaha memastikan nilai A untuk TTKI terlebih dahulu karena bebannya tidak seberat PAR. Kenyataan ini dapat dimanfaatkan mahasiswa untuk mendapatkan IP tertinggi yang mereka bisa dengan waktu yang bisa mereka sediakan untuk belajar.

II. DASAR TEORI

A. Teori Mengenai Penilaian Perkuliahan
Indeks Prestasi (IP) adalah nilai yang berhasil didapatkan mahasiswa dalam satu semester. Indeks tersebut merupakan penjumlahan dari jumlah SKS dikali dengan Indeks mata kuliah dibagi dengan jumlah SKS. Pada umumnya rentang indeks adalah dari 0 sampai 4. Apabila dirumuskan, rumus IP adalah sebagai berikut :

$$IP = \frac{\sum(SKS * I)}{\sum SKS}$$

Ket :

IP = Indeks Prestasi

SKS = SKS per mata kuliah

I = Indeks mata kuliah

IP tiap semester mulai dari semester yang diakumulasikan mulai dari semester terakhir mahasiswa mengambil mata kuliah disebut Indeks Prestasi Kumulatif (IPK). Rumus dari IPK adalah sebagai berikut :

$$IPK = \frac{\sum IP * SKS}{\sum SKS}$$

Ket:

IPK = Indeks Prestasi Kumulatif

IP = Indeks Prestasi per Semester

SKS = Jumlah SKS per Semester

B. Algoritma *Greedy*

Algoritma *greedy* adalah algoritma yang membentuk solusi langkah per langkah. Pada setiap langkah akan dipilih keputusan paling optimal. Keputusan tersebut tidak memperhatikan keputusan selanjutnya dan tidak dapat diubah lagi pada langkah selanjutnya.

- a. Prinsip Utama Algoritma *Greedy*
Prinsip utama dalam algoritma *greedy* adalah setiap langkah, kita mengambil keputusan paling optimal untuk langkah tersebut tanpa memperhatikan langkah selanjutnya. Solusi tersebut dinamakan solusi optimum lokal. Pengambilan nilai optimum lokal pada setiap langkah diharapkan akan mengarah kepada solusi optimum global, yaitu tercapainya solusi optimum yang melibatkan keeluruhan langkah dari awal sampai akhir.
- b. Elemen Algoritma *Greedy*
Elemen-elemen yang digunakan dalam penerapan algoritma *greedy* adalah
 - 1) Himpunan Kandidat
Himpunan yang berisi elemen pembentuk solusi
 - 2) Himpunan Solusi
Himpunan yang terpilih sebagai solusi persoalan
 - 3) Fungsi Seleksi
Fungsi yang memilih kandidat yang paling mungkin untuk mencapai solusi optimal.
 - 4) Fungsi Kelayakan
Fungsi yang memeriksa apakah suatu kandidat yang dapat bersama dengan

himpunan solusi yang sudah terbentuk tidak melanggar kendala yang ada.

- 5) Fungsi Objektif
Fungsi yang mengoptimalkan solusi

c. Skema Umum Algoritma *Greedy*

Asumsikan terdapat elemen-elemen algoritma *greedy* sebagai berikut:

- C = Himpunan Kandidat
- S = Himpunan Solusi
- select() = Fungsi Seleksi
- feasible() = Fungsi Kelayakan

Skema umum algoritma *greedy* dapat kita tuliskan sebagai berikut:

- 1) Inisialisasi S dengan kosong
- 2) Pilih sebuah kandidat dari C dengan select().
- 3) Kurangi C dengan kandidat yang telah dipilih dari langkah 2).
- 4) Periksa apakah kandidat yang dipilih bersama-sama dengan S yang sudah ada membentuk solusi yang layak. Jika iya, masukkan kandidat tersebut ke S, jika tidak buang kandidat tersebut.
- 5) Ulangi langkah ke 2) sampai tidak ada kandidat yang bisa dipilih lagi dari C.

d. Contoh Penerapan Algoritma *Greedy*

Contoh penerapan algoritma *greedy* adalah pada masalah penukaran uang. Misalkan seseorang ingin menukar uang senilai 31 dollar dengan nominal-nominal lain yang lebih kecil. Nominal-nominal yang tersedia adalah 1, 5, 10, dan 25 dollar. Carilah jumlah minimum uang yang diperlukan untuk penukaran tersebut. Apabila menggunakan algoritma *greedy*, maka kita akan mencari nominal terbesar yang tersedia untuk menukarkan koin tersebut. Pada kasus ini kita akan mengambil uang 25 dollar sebagai himpunan solusi kita, sehingga kita perlu mencari (31-25) dollar = 6 dollar lagi. Dengan mengulangi langkah yang sama, maka kita akan mengambil 5 dollar, sehingga uang yang tersisa adalah (6-5) dollar = 1 dollar. Kemudian kita akan mengambil uang 1 dollar dan tidak ada uang yang tersisa. Dengan algoritma *greedy* tersebut kita menukarkan uang 31 dollar dengan masing-masing 1 lembar uang 25 dollar, 5 dollar, dan 1 dollar.

II. ANALISIS PERMASALAHAN

A. Menghitung Waktu yang Dialokasikan Untuk Belajar

Untuk menghitung waktu yang dialokasikan kita asumsikan berapa rata-rata waktu belajar kita per minggu dan kalikan waktu tersebut dengan jumlah minggu kuliah. Sebagai contoh, apabila per minggu kita belajar selama 40 jam dan terdapat 14 minggu, maka total waktu = 40 * 14 = 560 jam

B. Memetakan Struktur Penilaian Tiap Mata Kuliah

Buat daftar setiap mata kuliah, jumlah SKS, dan komponen-komponen penilaian dan persentasenya. Sebagai contoh :

NO	Matkul	SKS	Penilaian	Persentase
1	Stima	3	UTS UAS Tubes Tucil	30 30 27 13
2	OOP	3	UTS UAS Tubes Kuis Praktikum	25 25 30 10 10
3	Basdat	3	UTS UAS Tubes Praktikum Kuis	25 25 25 15 10
4	OS	3	UTS UAS Tubes	35 35 30
5	Probstat	3	UTS UAS Tubes Kuis PR	30 30 15 10 10
6	DRPL	2	UTS UAS Tugas	30 30 40

C. Memprediksi Waktu Minimal yang Dibutuhkan Untuk lulus dan Untuk Mendapatkan Indeks A pada setiap mata kuliah

Untuk memprediksi waktu minimal untuk lulus maupun mendapatkan nilai A, buat asumsi berapa waktu yang dibutuhkan untuk

setiap komponen-komponen penilaian.
Sebagai contoh (waktu dalam jam):

NO	Matkul	Penilaian	lulus	A
1	Stima	UTS,	10	25
		UAS	10	25
		Tubes	15	45
		Tucil	15	45
2	OOP	UTS	20	45
		UAS	20	45
		Tubes	30	70
		Kuis	8	16
		Praktikum	10	15
3	Basdat	UTS	15	35
		UAS	15	35
		Tubes	15	30
		Praktikum	5	15
		Kuis	8	16
4	OS	UTS	5	15
		UAS	5	15
		Tubes	20	50
5	Probat	UTS	8	20
		UAS	8	20
		Tubes	5	12
		Kuis	5	10
		PR	5	10
6	DRPL	UTS	8	20
		UAS	8	20
		Tugas	10	25

D. Membuat Skala Prioritas untuk Masing-Masing Komponen Mata Kuliah

Untuk setiap komponen, penulis membuat skala prioritas. Skala prioritas ini nanti akan menjadi fungsi seleksi pada saat mengeksekusi algoritma *greedy* tersebut. Rumus dari skala prioritas adalah sebagai berikut:

$$sp = \frac{sks * pp}{wa - wl}$$

Ket :

sp = skala prioritas

sks = sks mata kuliah dari komponen penilaian tersebut

pp = persentase komponen penilaian

wa = waktu minimal yang dibutuhkan untuk mendapatkan nilai A pada komponen tersebut

wl = waktu minimal yang dibutuhkan untuk lulus pada komponen tersebut

IV. PENERAPAN ALGORITMA *GREEDY*

Elemen-elemen yang digunakan dalam algoritma *greedy* untuk program ini adalah”

1) Himpunan Kandidat

Himpunan kandidat untuk persoalan ini adalah komponen-komponen penilaian yang bisa diambil.

2) Himpunan Solusi

Himpunan solusi untuk persoalan ini adalah kumpulan komponen-komponen penilaian yang diharapkan menghasilkan IP terbaik.

3) Fungsi Seleksi

Fungsi seleksi untuk persoalan ini adalah menghitung skala prioritas untuk masing-masing komponen mata kuliah yang sudah dijelaskan pada Analisis permasalahan.

4) Fungsi Kelayakan

Fungsi kelayakan untuk persoalan ini adalah apabila waktu yang tersisa tidak mencukupi untuk belajar lagi.

5) Fungsi Objektif

Fungsi objektif untuk persoalan ini adalah memilih komponen penilaian berdasarkan fungsi seleksi.

Implementasi algoritma *greedy* tersebut adalah sebagai berikut (dengan kasus uji adalah contoh pada bagian II bagian B dan C) :

// Tipe bentukan Matkul

```
Type Matkul: <
    nama_matkul : string
    sks : integer
    jumlah_penilaian :
integer
    penilaian : array of string
    persentase_penilaian : array of
integer
    waktu_lulus : array of integer
    waktu_a : array of integer
    prioritas : array of real
>
```

// Tipe bentukan TabelMatkul

```
Type TabelMatkul : <
    daftar_matkul : array of
Matkul
    int banyakmatkul : integer
>
```

```
// prosedur untuk mengambil
//komponen-komponen mata kuliah
//yang optimal
```

```
procedure greedy(input/output
TabelMatkul,input/output waktu_total)
```

```
// Kamus Lokal
```

```
i, j, waktu_total, waktu_min : integer
maxprio : real
```

```
// Algoritma
// Mengambil semua matakuliah //dengan
waktu yang dibutuhkan //untuk lulus
terlebih dahulu
```

```
for i←1 to T.banyakmatkul do
  for int j←1 to
  T.daftar_matkul[i].jumlah_penil
aian do
    lulus =
    T.daftar_matkul[i].waktu_lulus[
j]; waktu_total = waktu_total -
lulus;
```

```
//Mengambil matakuliah dengan waktu
//yang dibutuhkan untuk mendapat nilai
//A sampai waktunya habis
do{
```

```
  maxprio ← 0; waktu_min ← 9999
  for i←1 to T.banyakmatkul do
    for j←1 to
    T.daftar_matkul[i].jumlah do
      if(T.daftar_matkul[i].prioritas
[j] > maxprio) then
        maxprio =
        T.daftar_matkul[i].prioritas[j]
      if(waktu_min >=
      T.daftar_matkul[i].waktu_a[j] -
      T.daftar_matkul[i].waktu_lulus[
j]) then
        waktu_min ←
        T.daftar_matkul[i].waktu_a[j] -
        T.daftar_matkul[i].waktu_lulus[
j]
```

```
for i←1 to T.banyakmatkul do
  for j←1 to
  T.daftar_matkul[i].jumlah_penil
aian do
    if
    ((T.daftar_matkul[i].prioritas[
j]=maxprio) and (waktu_total >=
T.daftar_matkul[i].waktu_a[j]-
(T.daftar_matkul[i].waktu_lulus
[j])) then
      print("Belajar Untuk
",T.daftar_matkul[i].penilaian[
j]," " ,
T.daftar_matkul[i].nama_matkul)
T.daftar_matkul[i].prioritas[j]
← 0;
waktu_total ← waktu_total -
T.daftar_matkul[i].waktu_a[j] +
T.daftar_matkul[i].waktu_lulus[
j]
```

```
else if ((
T.daftar_matkul[i].prioritas[j]
=maxprio) and (waktu_total <=
T.daftar_matkul[i].waktu_a[j] -
T.daftar_matkul[i].waktu_lulus[
j])) then
```

```
T.daftar_matkul[i].prioritas[j]= 0;
  } while(waktu_total >=
waktu_min);
```

```
Administrator: C:\windows\system32\cmd.exe
E:\tugas\sem4\stima\makalah>g++ -o main makalah.cpp
E:\tugas\sem4\stima\makalah>main.exe
Belajar Untuk UTS OS
Belajar Untuk UAS OS
Belajar Untuk UTS Probstat
Belajar Untuk UAS Probstat
Belajar Untuk Tubes Probstat
Belajar Untuk UTS Stima
Belajar Untuk UAS Stima
Belajar Untuk Kuis Probstat
Belajar Untuk PR Probstat
Belajar Untuk Tugas DRPL
Belajar Untuk Tubes Basdat
Belajar Untuk UTS DRPL
Belajar Untuk UAS DRPL
Belajar Untuk Praktikum Basdat
Belajar Untuk Kuis OOP
Belajar Untuk UTS Basdat
Belajar Untuk UAS Basdat
Belajar Untuk Kuis Basdat
Belajar Untuk Tubes OS
Belajar Untuk UTS OOP
Belajar Untuk UAS OOP
Belajar Untuk Tubes Stima
Belajar Untuk Tubes OOP
Belajar Untuk Praktikum OOP
Belajar Untuk Tucil Stima
Belajar Untuk UTS Stima
Belajar Untuk UAS Stima
Belajar Untuk Tubes Stima
Belajar Untuk Tucil Stima
Belajar Untuk UTS OOP
Belajar Untuk UAS OOP
Belajar Untuk Kuis OOP
Belajar Untuk Kuis Probstat
E:\tugas\sem4\stima\makalah>_
```

Gambar 2 : Hasil program
Sumber : Dokumentasi penulis

V. KESIMPULAN

Berdasarkan hasil analisis dan penerapan algoritma *greedy* dalam persoalan meraih IP tertinggi dalam waktu yang terbatas, maka dapat disimpulkan bahwa

- 1) Algoritma *greedy* dapat digunakan dalam menentukan prioritas belajar untuk menentukan IP tertinggi dengan waktu yang sudah dialokasikan.
- 2) Solusi yang dihasilkan algoritma *greedy* menyediakan solusi optimal untuk simulasi ini. Akan tetapi, algoritma ini belum tentu

menghasilkan IP yang terbaik pada kenyataannya karena ada saja faktor-faktor di luar kendali mahasiswa yang mempengaruhi IP.

VI. UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih terutama kepada Tuhan Yesus Kristus karena berkat anugerah dan berkat-Nya penulis dapat menyelesaikan makalah ini. Penulis juga mengucapkan terima kasih kepada Ibu Dr. Nur Ulfa Maulidevi ST, M.Sc. sebagai dosen mata kuliah IF 2211 Strategi Algoritma di kelas K-2.

REFERENSI

[1]Anany Levitin, “Design and Analysis of Algorithm”, Pearson Education Inc. 2012

[2]<http://stephanusar.blogspot.co.id/2012/11/pengertian-metode-greedy-dan-algoritma.html> diakses pada tanggal 16 Mei 2017

[3]Peraturan Akademik & Kemahasiswaan Institut Teknologi Bandung 2015

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 17 Mei 2017



Rizki Halasan

13515095