

Tugas Besar I IF2211 Strategi Algoritma

Aplikasi Algoritma *Greedy* pada Smart Building Environment

Waktu pemberian tugas: 2 Februari 2015

Batas pengumpulan : 23 Februari 2014 jam 7.30

Arsip pengumpulan : - CD yang berisi *Source* dan *Exe* program disertai *readme.txt*
- Laporan (*hard copy*)

Tempat pengumpulan : Di atas loker Lab IRK

Sekilas Smart Building Environment Menggunakan Smart Devices

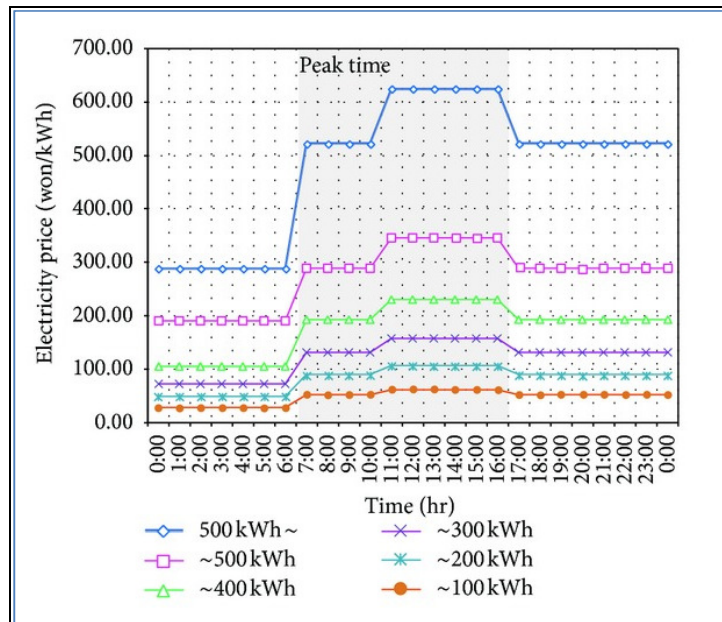
Saat ini teknologi *smart grid* sudah semakin maju. *Smart grid* yang dimaksudkan di sini adalah teknologi yang merupakan integrasi dari berbagai proses dalam sistem tenaga listrik, mulai dari sumber penghasil listrik hingga sampai pada konsumen pengguna listrik, baik rumah tangga maupun perkantoran. Dengan memanfaatkan teknologi komunikasi dan pemrosesan komputer, *smart grid* bertujuan untuk dapat memanfaatkan tenaga listrik yang seefektif dan seefisien mungkin.

Lingkungan *smart building*, juga mulai banyak menggunakan teknologi ini, dengan memanfaatkan *smart meter*. Dengan smart meter, penggunaan listrik setiap waktunya dapat langsung diketahui baik oleh konsumen maupun oleh pihak penyedia layanan listrik. Penyedia layanan listrik, dapat menerapkan tarif yang berbeda-beda untuk setiap slot waktu tertentu dalam satu hari. Pada saat-saat tertentu, biasanya saat penggunaan listrik cukup banyak, tarif yang diberlakukan adalah tarif mahal. Hal ini bertujuan agar konsumen dapat mengalihkan penggunaan listrik pada slot waktu lain, sehingga tidak menumpuk beban nya pada slot waktu tertentu.

Tarif listrik tersebut menerapkan system progresif, artinya semakin banyak penggunaan listrik, maka tarif per kWh juga akan meningkat. Gambar 1 menunjukkan contoh tarif listrik yang bersifat dinamik. Pada gambar tersebut, terlihat bahwa pada jam 07.00 – 16.00 adalah waktu puncak, di mana pemakaian listrik umumnya meningkat. Pada saat tersebut, tarif dasar listrik yang diberlakukan juga berbeda. Tarif juga bersifat progresif, dalam arti saat penggunaan listrik meningkat maka tarif per kWh juga meningkat. Sebagai contoh pada Gambar 1, penggunaan listrik pada jam 00.00 – 06.00 tarif dasarnya adalah sekitar 20/ kWh. Namun jika penggunaannya sudah mencapai 200 kWh, maka tarifnya menjadi 50/ kWh. Peningkatan tarif per kWh berlaku seiring meningkatnya jumlah pemakaian listrik.

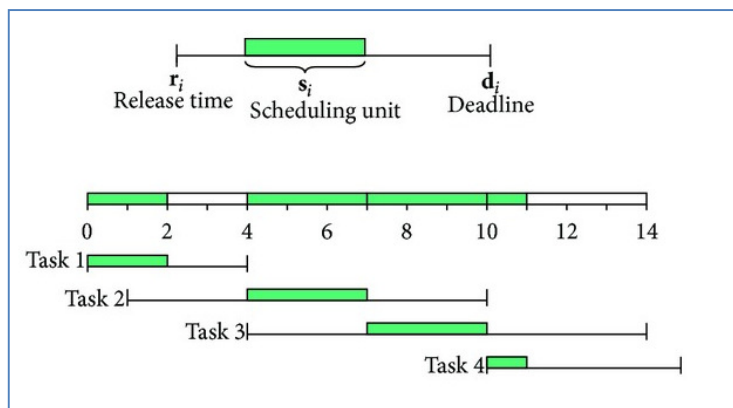
Dengan adanya informasi dari smart meter dan informasi tarif dari penyedia listrik, maka pemilik smart building dapat mengatur penggunaan peralatan listrik di rumahnya agar mendapatkan biaya listrik seminimal mungkin, namun kebutuhannya terpenuhi. Pengaturan yang perlu dilakukan oleh pemilik smart building adalah penentuan kapan suatu peralatan listrik harus dinyalakan, dan kapan harus dimatikan. Sebagai contoh, jika terdapat 23 alat listrik, dan satu slot waktu untuk penentuan on atau off alat listrik adalah 30 menit, maka banyaknya kemungkinan jadwal untuk satu hari adalah:

$$2^{23 \times 24 \times 2} = 2^{1104}$$



Gambar 1. Contoh Tarif Listrik yang Dinamis¹

Setiap peralatan listrik, akan memiliki batasan kapan bisa dinyalakan, dan kapan harus dimatikan, sehingga alokasi waktu penyalaan harus dilakukan pada rentang waktu tersebut. Contoh pada Gambar 2 dapat dilihat, setiap aktivitas memiliki batasan waktu mulai dan waktu selesai. Oleh karena itu aktivitas tersebut harus dialokasikan di antara batasan waktu tersebut.



Gambar 2. Contoh persoalan alokasi task

Di dalam tugas ini, Anda diminta mengaplikasikan algoritma *greedy* untuk mendapatkan tagihan biaya listrik seminimal mungkin, namun kebutuhan peralatan listrik yang diinginkan dapat terpenuhi semaksimal mungkin. Setiap peralatan listrik akan memiliki waktu minimum harus dinyalakan, bersifat wajib atau opsional untuk dinyalakan, serta waktu tertentu dapat beroperasi. Di akhir aplikasi yang anda buat, harus ada perhitungan kompleksitas dari pendekatan *greedy* yang Anda gunakan. Pada saat demo tugas, asisten akan memberikan file teks yang berisi daftar tarif dan daftar peralatan listrik. Setiap kelompok akan dibandingkan hasil akhirnya, dan kelompok dengan

¹Sumber Gambar: Lee E, Bahn H. Electricity Usage Scheduling in Smart Building Environments Using Smart Devices. *The Scientific World Journal* 2013;2013:468097. doi:10.1155/2013/468097.

optimasi terbaik, serta kompleksitas yang optimal, adalah kelompok pemenang dan mendapatkan nilai bonus. Jadwal yang dihasilkan dapat ditampilkan dengan visualisasi yang menarik dan 'mudah' dibaca oleh yang menggunakan.

Spesifikasi program :

1. Program mampu membaca input file berisi daftar tarif setiap slot waktu, daftar peralatan listrik konsumen, dan kebutuhan waktu penggunaan alat listrik dengan satuan slot waktu.

Input File	Template
46,4,6,1300.	<banyaknya slot waktu, contoh per 30 menit untuk 24 jam>,<banyaknya selang tarif>,<banyaknya nilai progresif><total kWh yang bisa digunakan>
0,6,100,20,6,12,100,50,... 18,24,100,50.	<mulai_selang1>,<akhir_selang1>,<kWh_kurangdari_100>,<biaya_per_kWh>,<mulai_selang2>,<akhir_selang2>,<kWh_kurangdari_100>,<biaya_per_kWh>...<sampai_banyaknya_selang_tarif>.
0,6,200,50,6,12,200,90,... 18,24,200,90.	<mulai_selang1>,<akhir_selang1>,<kWh_kurangdari_200>,<biaya_per_kWh>,<mulai_selang2>,<akhir_selang2>,<kWh_kurangdari_200>,<biaya_per_kWh>,<mulai_selang3>,<akhir_selang3>,<kWh_kurangdari_300>,<biaya_per_kWh>...<sampai_banyaknya_selang_tarif>.
.	.
.	.
.	.
0,6,>500,300,6,12,>500,500,... 18,24,>500,500.	<sampai banyaknya nilai progresif>
20	<banyaknya_peralatan_listrik>
Rice_cooker,100,1,0,6,wajib,1.	<alat>,<kWh_yang_dibutuhkan_oleh_alat_listrik><waktu_yg_harus_dijadwalkan_dalam_slot_dalam_satuan_30_menit>,<release_time>,<deadline>,<sifat_alat>,<banyaknya_penyalaaan_dalam_1_hari>
Kulkas,200,46,0,24,wajib,1	<alat>,<kWh_yang_dibutuhkan_oleh_alat_listrik><waktu_yg_harus_dijadwalkan_dalam_slot>,<release_time>,<deadline>,<sifat_alat>
.	.
.	.
.	.
AC,500,1,12,24,wajib,5	<sampai banyaknya peralatan listrik>

2. Program menampilkan hasil penjadwalan dalam visualisasi yang memudahkan pengguna melihat, dengan menampilkan tarif yang dikeluarkan berdasarkan jadwal yang dibuat, dan terlihat alat listrik mana yang mendapatkan alokasi, dan mana yang tidak mendapatkan alokasi.
3. Setiap kelompok harus menyatakan secara eksplisit semua spesifikasi detail tambahan yang dibutuhkan.

Lain – lain :

1. Anda dapat menambahkan fitur fungsional lain yang menunjang program yang anda buat (unsur kreatifitas).Sebagai contoh, hasil penjadwalan bisa dilakukan “drag and drop”, dan saat dilakukan terlihat task mana yang dilanggar dan perubahan tarif yang dihasilkan.Jika diperlukan penyederhanaan, jelaskanlah alasannya pada laporan dan deskripsikanlah penyederhanaan yang dilakukan tersebut.
2. Program ini harus Anda buat berbasis *Graphical User Interface* (GUI) dengan menggunakan kaskas visual programming klasik seperti Visual C, Visual C++, Borland Delphi, Borland C++ Builder, dll.
3. Tugas dikerjakan per kelompok dengan jumlah anggota adalah 3 orang dan boleh lintas kelas.
4. Program harus modular dan mengandung komentar yang jelas.

5. Mahasiswa harus membuat program sendiri, tetapi belajar dari contoh-contoh program *game* serupa yang sudah ada tidak dilarang (tidak boleh mengkopi *source code* dari program orang lain).
6. Pengumpulan paling lambat adalah tanggal 23Februari 2014pukul 7.30. Keterlambatan akan mengurangi nilai.
7. Program disimpan di dalam *folder* StrAlgo1-xxxxx. Lima digit terakhir adalah NIM anggota terkecil. Didalam folder tersebut terdapat tiga folder *bin*, *src* dan *doc* yang masing-masing berisi :
 - a. Folder *bin* berisi *executable file (exe)*
 - b. Folder *src* berisi *source code* dari program
 - c. Folder *doc* berisi dokumentasi program dan *readme*

Folder ini disimpan dalam bentuk CD untuk dikumpulkan bersama berkas laporan dimasukan kedalam amplop coklat.
8. Semua pertanyaan menyangkut tugas ini harus dikomunikasikan melalui milis agar dapat dicermati oleh semua peserta kuliah IF2211 (milis IF2211@students.if.itb.ac.id).
9. Demo program akan dilaksanakan setelah pengumpulan. Jadwal demo akan diumumkan pada saat pengumpulan di Lab IRK. Pengaturan jadwal demo akan dikoordinasikan oleh asisten.
10. Tiap anggota harus memahami proses pembuatan program, karena akan ada pertanyaan-pertanyaan yang harus dijawab per individu.
11. Pada saat demo, asisten akan memanggil per kelompok sesuai jadwal yang diisi sebelumnya. Kelompok yang tidak berkepentingan dilarang masuk. Demo dilakukan di Lab IRK.

Isi laporan :

Cover: *Cover* laporan ada foto anggota kelompok (foto bertiga). Foto ini menggantikan logo “gajah” ganesha.

Bab 1: Deskripsi masalah (dapat meng-*copy paste* file tugas ini)

Bab 2: Dasar teori (berisi deskripsi singkat algoritma *greedy*).

Bab 3: Analisis Pemecahan Masalah. Di dalam bab ini diuraikan langkah-langkah pemecahan masalah dengan algoritma *greedy*. Anda harus menuliskan apa yang menjadi elemen-elemen algoritma *greedy*-nya pada masalah ini (himpunan kandidat, himpunan solusi, fungsi seleksi, dll). Nilai bonus bagi anda jika dapat membuktikan apakah algoritma *Greedy* anda itu memberikan solusi optimal dan contoh kontranya (contoh yang tidak selalu memberikan solusi optimal).

Bab 4: Implementasi dan pengujian. Bab ini berisi:

- a. Spesifikasi teknis program, termasuk di dalamnya struktur data, fungsi dan prosedur (*header* fungsi dan prosedur saja, tidak perlu *source code*), antarmuka, dan lain-lain yang dianggap perlu.
- b. *Capture layar* yang memperlihatkan hasil penjadwalan.
- c. Analisis hasil pengujian.

Bab 5: Kesimpulan dan saran (hasil yang dicapai, saran pengembangan).

Tuliskan juga referensi (buku, web), yang dipakai/diacu di dalam Daftar Referensi.

Keterangan laporan :

1. Laporan ditulis dalam bahasa Indonesia yang baik dan benar, tidak perlu panjang tetapi tepat sasaran dan jelas.
2. Laporan tidak perlu memakai *cover* mika dan dijilid. Cukup dibuat agar laporan tidak akan tercecer bila dibaca.
3. Laporan boleh menggunakan kertas rius, boleh bolak-balik, boleh dalam satu halaman kertas terdapat dua halaman tulisan asalkan masih terbaca.
4. Identitas per halaman harus jelas (misalnya : halaman, kode kuliah).

Penilaian :

1. Kebenaran program (40%) : program mampu berjalan sesuai dengan spesifikasi yang diberikan.
2. Demo – pemahaman Anda dalam pembuatan program (30%)
3. Laporan (20%)
4. *Interface, features* program, dan unsur kreativitas (10%)

-selamat mengerjakan-