

Pengembangan Program Komputer Penjadwalan Matakuliah Berdasarkan Pewarnaan Graf dengan Algoritma Welsh-Powell Terbobot

Andi Pujo Rahadi¹, Ernest Beardly Pani²

^{1,2} Pendidikan Matematika, Universitas Advent Indonesia,
Jl. Kolonel Masturi 288 Bandung Barat, Jawa Barat, 40559.
andi.rahadi@unai.edu¹, ernestbeardly@yahoo.com²

Abstrak – Penjadwalan matakuliah di suatu universitas merupakan permasalahan yang kompleks, dikarenakan melibatkan berbagai sumber daya yang memiliki kemungkinan tumpang tindih (bentrok) cukup besar. Beberapa universitas tidak lagi menyusun jadwal matakuliahnya secara manual. Mereka telah menggunakan berbagai program komputer penjadwalan berbasis web yang cukup populer, sebagai contoh Lantiv, Unitime, dan sebagainya. Program-program komputer tersebut merupakan program yang dikembangkan di negara lain. Bagaimana dengan program produksi Indonesia? Penelitian ini merupakan upaya untuk menjawab hal tersebut. Program yang dikembangkan dalam penelitian ini dilandaskan pada teori pewarnaan graf, khususnya pewarnaan simpul (*vertex*) dengan algoritma Welsh-Powell Terbobot. Program komputer yang dihasilkan diberi nama ARC Scheduler yang ditulis dengan bahasa C++ dan IDE Code Blocks. Program ini telah diujicobakan di Universitas Advent Indonesia pada tahun 2018 dan 2019 dengan respon pengguna sangat puas. Program ARC Scheduler terus dikembangkan dalam hal pengoperasian secara remote, baik berbasis web maupun berbasis android.

Kata kunci: Penjadwalan Matakuliah, Pewarnaan Graf, Algoritma Welsh Powell Terbobot, ARC Scheduler.

PENDAHULUAN

Setiap universitas melakukan penyusunan jadwal matakuliah pada awal semester atau awal tahun ajaran yang akan diselenggarakan. Berbagai sumber daya perlu dipertimbangkan dalam penyusunan jadwal tersebut, antara lain mahasiswa, dosen, kapasitas ruang, hari kerja, jam kerja, dan sebagainya. Mengakomodasi semua sumber daya tersebut dan menghasilkan jadwal perkuliahan yang bebas bentrok tentu bukanlah pekerjaan yang mudah. Bila jumlah mahasiswa dan dosen tidak banyak, penyusunan jadwal masih bisa dilakukan secara manual. Bagaimana dengan universitas yang memiliki ribuan bahkan puluhan ribu mahasiswa?

Jumlah mahasiswa yang semakin besar akan berakibat potensi konflik jadwal semakin besar dan penyusunan jadwal secara manual menjadi pekerjaan yang kompleks dan membutuhkan waktu yang panjang.

Mengingat penyusunan jadwal merupakan pekerjaan yang kompleks, beberapa program komputer telah dikembangkan untuk membantu pekerjaan penyusunan jadwal ini. Program yang cukup banyak dikenal di antaranya *Lantiv Studio* [1] dan *Unitime* [2]. Sejauh ini belum ada program komputer penyusunan jadwal matakuliah yang dikembangkan di Indonesia. Untuk itulah penulis berupaya melakukan penelitian pengembangan ini.

A. Tujuan Penelitian

Tujuan dari penelitian ini adalah membangun suatu program komputer penjadwalan matakuliah yang memenuhi kriteria-kriteria berikut

- (1) **Bebas konflik:** Jadwal matakuliah di semua program studi (prodi) tidak ada yang bentrok.
- (2) **Cepat:** Proses penyusunan jadwal menjadi lebih mudah dan lebih cepat.
- (3) **Fleksibel:** Dapat mengakomodir permintaan khusus dosen maupun mahasiswa terkait kondisi tertentu.
- (4) **Adaptif:** mudah disesuaikan dengan perubahan-perubahan yang mungkin diperlukan di pertengahan semester.

Penelitian-penelitian mengenai hal ini telah berlangsung puluhan tahun sejak Welsh-Powell, 1967 [3] dan Werra, 1984 [4]. Para peneliti menggunakan berbagai pendekatan, berbagai algoritma, dan dilakukan di banyak negara. Dalam paper ini akan disajikan penggunaan teori pewarnaan graf dengan algoritma Welsh Powell Terbobot untuk mengembangkan program komputer penjadwalan matakuliah yang dapat memenuhi kriteria-kriteria di atas. Nama program komputer yang dikembangkan adalah *ARC Scheduler (Automated Random Coloring Scheduler)*.

Dalam subbab kajian pustaka akan diuraikan beberapa penelitian terkait dengan pengembangan program komputer penjadwalan matakuliah serta ringkasan mengenai graf dan pewarnaannya.

B. Kajian Pustaka

Salah satu teori yang dapat digunakan untuk mengembangkan program komputer penjadwalan matakuliah adalah teori pewarnaan graf [5]. Dalam sub bab ini akan dijelaskan secara ringkas mengenai graf dan pewarnaan *vertex* graf, serta bagaimana penjadwalan matakuliah dapat dimodelkan sebagai graf. Pengertian graf, derajat *vertex*, pewarnaan graf, dan bilangan kromatik disajikan pada definisi 1, 2, 3, dan 4.

Definisi 1. Suatu Graf $G=(V,E)$ terdiri dari V , suatu himpunan simpul (*vertex*) dan E , suatu himpunan sisi (*edge*). Setiap *edge* berkaitan (*incidence*) dengan satu atau dua *vertex*, yaitu titik ujung *edge* tersebut. Sebuah *edge* dikatakan menghubungkan titik-titik ujungnya, dan dua *vertex* dikatakan saling *adjacent* bila ada sebuah *edge* yang menghubungkannya [6].

Edge yang *incidence* dengan tepat 1 *vertex* disebut loop. Bila pada graf G ada 2 *vertex* dihubungkan oleh lebih dari 1 *edge* maka pada graf G dikatakan terdapat multiple *edge*. Graf yang memuat loop ataupun multiple *edge* merupakan graf yang tidak sederhana. Dengan kata lain suatu graf dikatakan sederhana (*simple graph*) bila graf itu tidak memuat multiple *edge* maupun loop.

Definisi 2. Derajat dari suatu *vertex* v anggota graf G , dilambangkan dengan $deg(v)$, adalah banyaknya *edge* G yang *incident* dengan v .

Matakuliah-matakuliah dapat dimodelkan sebagai *vertex-vertex* di dalam suatu graf sederhana di mana dua matakuliah dikatakan *adjacent* bila menggunakan sumber daya yang sama. Sumber daya yang dimaksud dapat merupakan mahasiswa, dosen, ruangan, dan sebagainya. Setelah model graf terbentuk, dilakukan pewarnaan terhadap *vertex-vertex*nya. Definisi formal mengenai pewarnaan *vertex* graf dan bilangan kromatik disajikan pada definisi 3 dan 4.

Definisi 3. Sebuah pewarnaan pada suatu graf sederhana adalah pemberian suatu warna pada setiap *vertex* pada graf sedemikian hingga tidak ada dua *vertex adjacent* yang mendapatkan warna yang sama [6]. Setiap warna dapat direpresentasikan oleh suatu bilangan asli.

Definisi 4. Bilangan kromatik χ adalah banyaknya warna minimum yang diperlukan untuk mewarnai suatu graf.

Sebagai contoh, bilangan kromatik dari graf planar [7] adalah 4. Hal ini berarti setiap graf planar

dapat diwarnai hanya dengan 4 warna berbeda sebarang banyak *vertex*nya.

Dalam memodelkan jadwal matakuliah sebagai suatu graf, setiap warna merepresentasikan slot waktu yang digunakan. Misalkan suatu universitas memberlakukan 5 hari kerja dan 8 jam kerja per hari, maka slot waktu selama 1 pekan adalah 40 slot. Dengan demikian pada graf yang dibentuk banyaknya warna tidak boleh melebihi 40. Dalam pengelolaan matakuliah yang efisien, banyaknya warna yang digunakan haruslah mendekati atau sama dengan bilangan kromatik dari model graf yang dibentuk. Beberapa contoh pewarnaan tanpa bantuan program komputer tersaji pada penelitian Rahadi [8] dan Jusuf [9].

Untuk melakukan pewarnaan pada suatu graf, diperlukan suatu aturan pewarnaan, dinamakan algoritma pewarnaan. Bila algoritma pewarnaan telah ditentukan, maka proses penulisan code dapat dimulai. Perbandingan performansi berbagai algoritma pewarnaan dapat ditemukan pada [10] dan [11]. Beberapa algoritma pewarnaan yang telah digunakan dalam pengembangan program komputer penjadwalan matakuliah di antaranya

- (1) Algoritma Genetika [12]
- (2) Algoritma Welsh Powell [13]
- (3) Algoritma Kunang-kunang [14]

Penelitian ini mengadaptasi algoritma Welsh Powell dan melakukan variasi yaitu memberikan bobot kepada setiap *vertex*. Oleh karena itu algoritma yang digunakan dalam penelitian ini dinamakan Algoritma Welsh Powell Terbobot.

C. Algoritma Yang Digunakan

Algoritma Welsh Powell Terbobot diawali dengan membentuk *vertex* terurut menurut *degree*nya dari *degree* terbesar sampai terkecil. Untuk meningkatkan prioritas matakuliah dilakukan pembobotan berdasarkan Persamaan 1 dan Persamaan 2. Selanjutnya *vertex* yang memiliki *degree* terbesar diberi warna 1, kemudian *vertex* itu dihapus, itulah iterasi pertama. Pada iterasi kedua diperhatikan apakah *vertex* dengan *degree* terbesar *adjacent* dengan *vertex* yang telah diwarnai? Bila *adjacent* maka *vertex* ini diberi warna 2, namun bila tidak maka ia diberi warna 1, lalu dihapus. Dilanjutkan iterasi ketiga secara looping hingga semua *vertex* mendapatkan warna. Pewarnaan dinyatakan valid bila banyaknya warna tidak melebihi slot waktu yang tersedia.

Variasi pembobotan pada algoritma tersebut dilakukan dengan memberikan *degree* baru kepada setiap *vertex* berdasarkan Persamaan 1 dan Persamaan 2. Bilangan n menunjukkan banyaknya *vertex* yang hendak dijadwalkan. Setiap matakuliah dapat dimodelkan sebagai satu atau lebih *vertex*.

$$\forall v_i, \quad i = 1, 2, \dots, n$$

$$deg(v_i) = deg(v_i) \times \frac{A+B+2}{2} \quad (1)$$

di mana:

A: Jumlah slot waktu dosen yang diblok,

B: Jumlah slot waktu mahasiswa yang diblok.

$$\forall v_i, \quad i = 1, 2, \dots, n$$

$$deg(v_i) = deg(v_i) + (S + 2) + k + 1 \quad (2)$$

di mana

S: Slot waktu yang belum digunakan

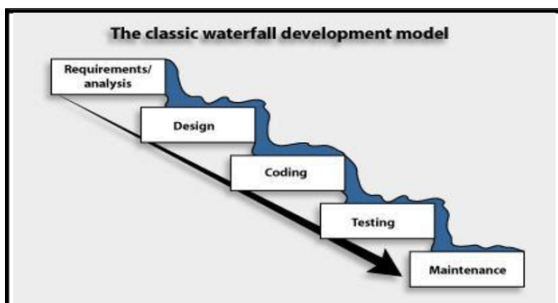
k : Suatu konstanta dengan $k \in \mathbb{N} + \{0\}$

i : 1, 2, 3, ..., n

n : Jumlah *vertex*.

METODE PENELITIAN

Penelitian pengembangan ARC Scheduler menggunakan metodologi Waterfall, yang disajikan pada Gambar 1. Metodologi Waterfall diawali dengan Analisis awal, dilanjutkan dengan Desain, kemudian Penulisan code, Pengujian (*testing*), dan Perbaikan (*maintenance*) [15].



Gambar 1 Model Pengembangan *Waterfall* [20]

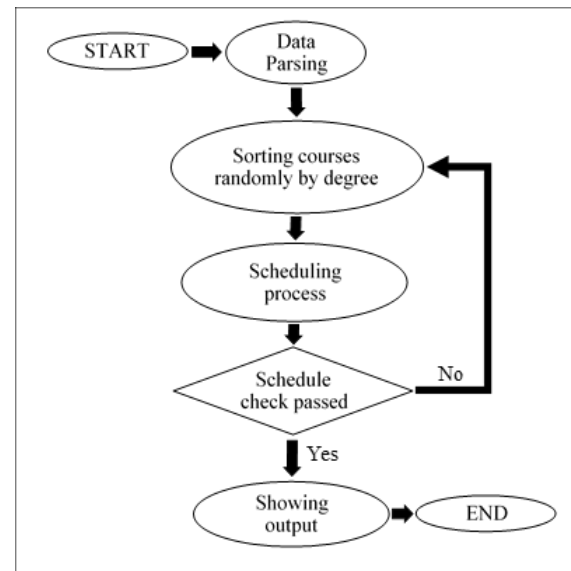
Dalam penelitian ini dipilih Universitas Advent Indonesia (Unai), berlokasi di Bandung, Jawa Barat sebagai sample pengujian. Unai merupakan universitas yang terdiri dari 6 fakultas dan mengelola 14 program studi. Berdasarkan informasi di laman resminya [16], jumlah mahasiswa Unai adalah 2005 orang dan jumlah dosen adalah 82 orang. Perkuliahan dilaksanakan dengan sistem semester (6 bulan).

A. Analisis Awal

Data yang diperlukan untuk penjadwalan kursus dikumpulkan dari Biro Administrasi Akademik (BAA) dan Departemen IT Service kampus Unai. Selain data daftar matakuliah, dilakukan pula wawancara dengan BAA mengenai permintaan-permintaan khusus dosen. Permintaan khusus yang dimaksud misalnya dosen berhalangan mengajar di jam tertentu atau berhalangan mengajar pada hari tertentu.

B. Desain

Desain dari program yang akan dikembangkan disajikan pada Gambar 2.



Gambar 2 Desain pengembangan ARC Scheduler

(1) Penguraian data (*Data Parsing*)

Data untuk semua mata kuliah diuraikan berdasarkan kode mata kuliah, nama mata kuliah, jenis mata kuliah, instruktur, sks (satuan kredit semester), jurusan, section, mahasiswa kasus khusus, dan jadwal tetap yang diperoleh dari wawancara. Data tersebut dimasukkan ke dalam file input (.csv) dengan bantuan Microsoft Excel. Format file input disajikan pada Gambar 3.

Untuk memperjelas cara pembuatan file input, diperhatikan baris ke 3 pada Gambar 3 sebagai contoh. Pada kolom *Code* dituliskan kode matakuliah yaitu IF173101. Pada kolom *Name* dituliskan nama matakuliah yaitu Administrasi dan Manajemen Gereja. Pada kolom *Type* dituliskan M yang artinya Matakuliah, bila dituliskan P maka itu memiliki arti Praktikum. Pada kolom *Instructor* dituliskan kode dari dosen yang ditugaskan mengajar matakuliah ini. Kode program studi dituliskan pada kolom *Dep* dan kode kelompok mahasiswa yang akan mengambil matakuliah tersebut dituliskan pada kolom *Section*. Kolom *Additional* disediakan untuk menuliskan NIM mahasiswa yang mengulang ataupun yang hendak mengambil matakuliah ini lebih cepat dari seharusnya.

(2) Mengurutkan *degree*

Pada tahap ini dari semua matakuliah dibentuk *vertex-vertex* dengan ketentuan setiap *vertex* mewakili 1 sks atau 2 sks dari suatu matakuliah.

	A	B	C	D	E	F	G	H
1	CODE	NAME	Type	INSTRUCT	CREDITS	DEP	SEC	ADDITIONAL
2	ENG3106A	Academic Writing	M	201108	4	20103	ABI4	
3	IF173101	Administrasi dan Manajemen Gereja	M	101618	3	10103	AIF3	1611003
4	ENG3104A	Advanced Grammar	M	201108	3	20103	ABI3	
5	SI163524	Akuisisi Manajemen dan Strategi SI	M	HN1682	3	80203	ASI3	
6	MM18120	Akuntansi Manajer	M	301127	3	30403	AMM	
7	TI161101	Algoritma dan Pemrograman I	M	401106	2	80103	ATI1	
8	SI161103	Algoritma dan Pemrograman I	M	401106	2	80203	ASI1	
9	SI162314	Aljabar Linear	M	401103	3	80203	ASI2	
10	MA153520	Aljabar Linier	M	201230	3	20203	AMA3	
11	BO172310	Anatomi dan Fisiologi Manusia II	M	501107	4	40103	ABO2	1741001
12	BO172311	Anatomi dan Fisiologi Tumbuhan	M	401104	3	40103	ABO2	1741001
13	IF171103	Aplikasi Komputer	M	HN1682	2	10103	AIF1	
14	BI181101	Aplikasi Komputer	M	201213	3	20103	ABI1	1521027
15	MA151103	Aplikasi Komputer	M	201213	3	20203	AMA1	
16	BO173525	Aplikasi Komputer Sains	M	401113	2	40103	ABO3	
17	FA173528	Aplikasi Komputer Sains	M	401113	2	40303	AFA3	

Gambar 3 Format file input

Dua *vertex* adjacent bila keduanya diajar oleh dosen yang sama atau diambil oleh mahasiswa yang sama atau menggunakan ruangan yang sama. Selanjutnya berdasarkan penguraian data maka ARC Scheduler akan menghitung *degree* dari masing-masing *vertex* tersebut dan menyusunnya (*sort*) dari *degree* terbesar hingga *degree* terkecil secara acak. Penyusunan secara acak dilakukan dalam hal ada beberapa *vertex* yang memiliki *degree* yang sama.

(3) Pewarnaan Vertex

Tahap ini merupakan inti dari program computer yang dikembangkan. Pewarnaan *vertex* dilakukan berdasarkan algoritma Welsh Powell Terbobot. Pada contoh kasus 1 disajikan penerapan algoritma ini.

(4) Menampilkan luaran

Luaran ditampilkan pada menu Main View. Selanjutnya pengguna dapat mencetak luaran dalam format *.pdf* atau dalam format *.mht*. Aktivitas penjadwalan juga dapat disimpan dalam format *.arcsch* sehingga dapat diedit pada waktu-waktu mendatang.

Contoh Kasus 1:

Misal terdapat tiga matakuliah: MK1, MK2, MK2 dengan SKS masing-masing: 3, 2, 5 sks dan diajar oleh dosen masing-masing: A, B, A. Misalkan kelompok mahasiswa yang mengambil berturut-turut adalah K1, K1, K2.

Solusi contoh kasus 1:

1. Mendefinisikan *vertex*.

MK1, MK2, dan MK2 masing-masing dibagi menjadi 2, 1, dan 3 *vertex*.

MK1 - v_1, v_2

MK2 - v_3

MK3 - v_4, v_5, v_6

Kapasitas masing-masing *vertex*:

$C(v_1) = 2$ sks, $C(v_2) = 1$ sks, $C(v_3) = 2$ sks, $C(v_4) = 2$ sks, $C(v_5) = 2$ sks, $C(v_6) = 1$ sks.

2. *Degree* awal

MK1 adjacent dengan MK2 dan MK3. Jadi v_1 adjacent dengan v_2, v_3, \dots , dan v_6 . v_2 adjacent dengan v_1, v_3, \dots , dan v_6 . Jadi $\text{deg}(v_1) = \text{deg}(v_2) = 5$
 MK2 adjacent dengan MK1 sehingga v_3 adjacent dengan v_1 , dan v_2 . Jadi $\text{deg}(v_3) = 2$
 MK3 adjacent dengan MK1 sehingga v_4 adjacent dengan v_1, v_2, v_5 , dan v_6 . v_5 adjacent dengan v_1, v_2, v_4 , dan v_6 . v_6 adjacent dengan v_1, v_2, v_4 , dan v_5 . Jadi $\text{deg}(v_4) = \text{deg}(v_5) = \text{deg}(v_6) = 4$.

3. Perhitungan ulang *degree*

Degree setiap *vertex* dihitung ulang menggunakan Persamaan 1.

4. Scheduling

- a. *Vertex* diurutkan secara acak.
- b. *Vertex* diurutkan dari *degree* terbesar kepada *degree* terkecil.
- c. Proses penjadwalan

- (i) Mewarnai *vertex* (yang memiliki *degree* terbesar dari semua *vertex* yang belum diwarnakan) kepada slot waktu yang belum digunakan dan memenuhi kondisi. Jika pewarnaan gagal, maka *degree vertex* tersebut dihitung ulang menggunakan Persamaan 2, dan proses penjadwalan ulang dilakukan dari langkah 4.a.

- (ii) Mewarnakan (dengan warna yang sama di (i)) *vertex* lain yang belum diwarnakan dan tidak adjacent dengan *vertex* di (i).
- (iii) Lakukan langkah (i) dan (ii) sampai semua *vertex* terwarnai.

5. Pemeriksaan warna

Bila banyaknya warna yang digunakan tidak melebihi banyaknya slot waktu yang tersedia, maka pewarnaan dinyatakan berhasil / valid. Bila tidak demikian, maka proses pewarnaan diulangi atau dihentikan.

Solusi Contoh Kasus 1 selesai.

C. Penulisan Code

Berdasarkan Algoritma Welsh Powell Terbobot, ditulis rangkaian code menggunakan Bahasa C++, yang diakses dari [17]. Adapun lingkungan pengembangan (*integrated development environment*) yang digunakan adalah CodeBlocks yang dapat diakses dari [18]. File input ditulis menggunakan program Microsoft Excel 2013. Seluruh proses penulisan code dilakukan dalam waktu sekitar 6 bulan hingga dihasilkan program *ARC Scheduler (Automated Random Coloring Scheduler)* versi yang pertama. Program versi pertama ini telah mampu menghasilkan luaran dalam format text. Program ini terus menerus dikembangkan hingga saat artikel ini ditulis telah dihasilkan ARC Scheduler versi ketujuh.

Hardware yang digunakan untuk mengembangkan program ini adalah sebuah laptop dengan spesifikasi prosesor AMD A4-9125 Radeon R3 2.3 GHz, RAM 4 GB, dan kapasitas *hard disk* 500 GB.

D. Ujicoba Program

Data yang dikumpulkan untuk tahap ujicoba penelitian ini adalah daftar matakuliah yang akan dibuka pada semester ganjil tahun ajaran 2018/2019. Adapun jumlah sks matakuliah tersebut adalah 1217 sks. Oleh karena keterbatasan halaman dalam naskah ini, file daftar matakuliah tersebut dapat diakses secara online pada [19] atau pembaca dapat menghubungi editor jurnal yang menerbitkan artikel ini.

Hasil ujicoba disajikan pada bab Hasil dan Pembahasan.

E. Rencana Pengembangan Lanjut

Program Penjadwalan Matakuliah Otomatis ini terus dikembangkan dalam beberapa aspek. Aspek yang pertama adalah tampilan dan kenyamanan pengguna. Tampilan menu utama dan tampilan menu data didesain agar memberikan kenyamanan optimal bagi pengguna. Aspek yang kedua adalah fleksibilitas, yaitu dengan

menambahkan fitur Block dan Unblock. Fitur block dan unblock dapat digunakan untuk mengunci atau membuka slot waktu tertentu berdasarkan permintaan dosen ataupun kebijakan tertentu dari pimpinan fakultas dan pimpinan Universitas. Aspek-aspek yang menjadi rencana pengembangan lanjut program ini di antaranya:

- (1) Program dapat dijalankan secara remote berbasis web dan atau berbasis aplikasi android.
- (2) Program dipisahkan menjadi front end untuk pengguna dan back end untuk administrator program.
- (3) Pengurusan paten.

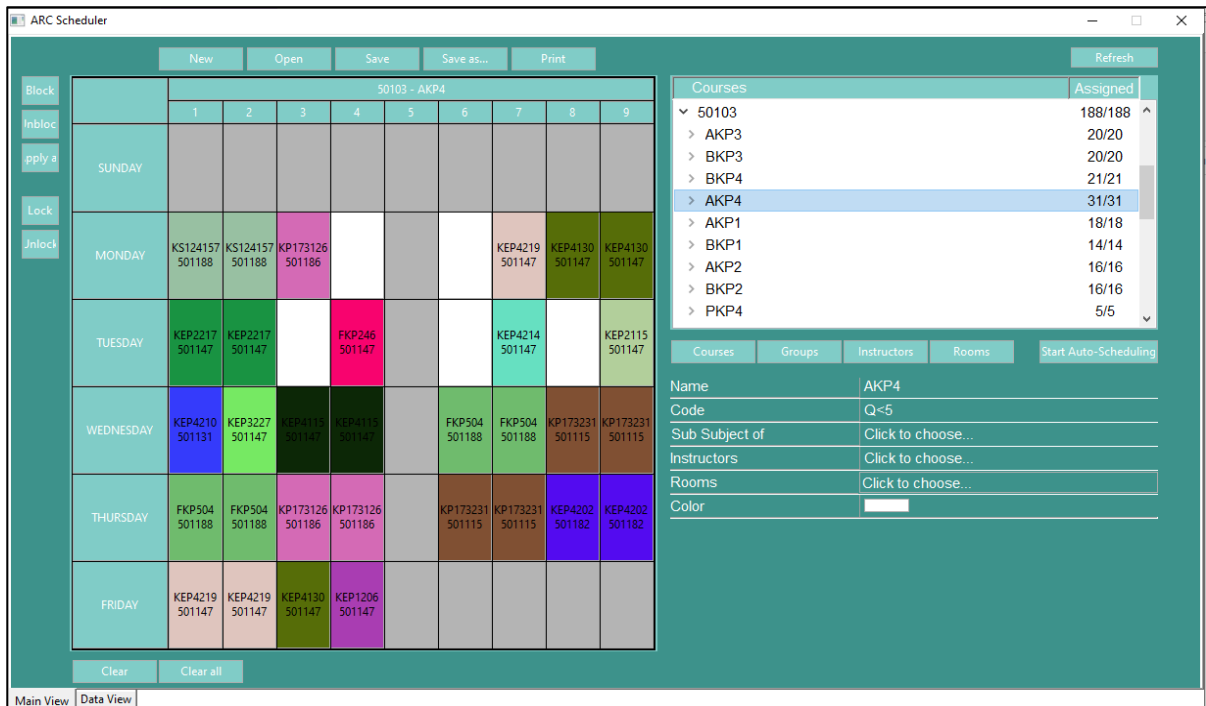
HASIL DAN PEMBAHASAN

Dalam bab ini akan dipaparkan hasil ujicoba program dengan system blackbox dan hasil wawancara dengan pengguna program. Yang dimaksud dengan system blackbox adalah file input dapat terbaca dengan baik oleh program dan program mampu menghasilkan output sesuai dengan algoritma yang telah dirancang.

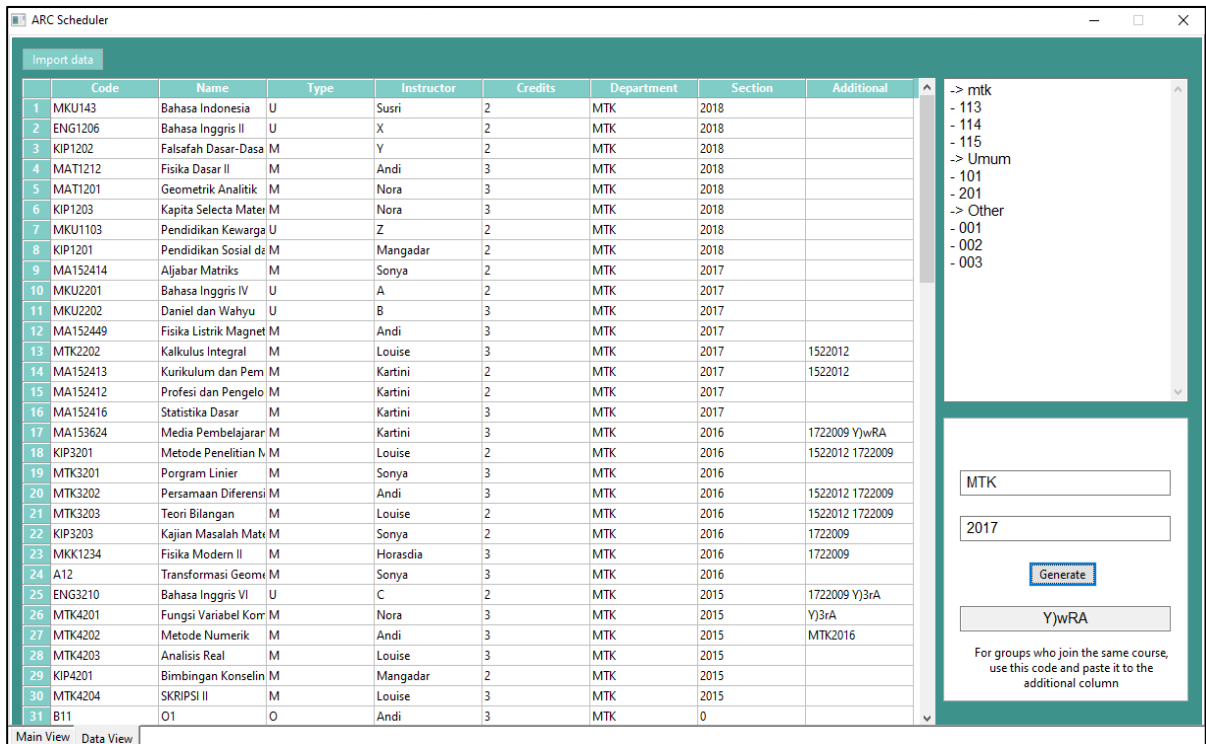
Tampilan menu utama dan tampilan menu data disajikan pada gambar 4 dan gambar 5, sedangkan output program disajikan pada gambar 6. Berdasarkan gambar 4, 5, dan 6 disimpulkan bahwa program mampu membaca input dengan benar dan menghasilkan output yang sesuai dengan algoritma yang digunakan. Waktu yang diperlukan untuk menyelesaikan seluruh penjadwalan kurang dari 5 detik, untuk 1217 sks matakuliah.

Pengguna yang diwawancarai dalam ujicoba ini adalah ibu Louise M. Saija selaku dekan Fakultas Keguruan dan Ilmu Pendidikan (FKIP) Universitas Advent Indonesia. Wawancara dilakukan dengan tatap muka dan pengisian kuesioner. Berdasarkan wawancara tersebut, pengguna merasa sangat puas dengan performa dan manfaat ARC Scheduler. Pengguna menilai ARC Scheduler mampu mengakomodasi permintaan khusus dosen, di antaranya ada dosen yang tidak dapat mengajar di pagi hari dan ada dosen yang meminta tidak mengajar di hari Jumat.

Program Komputer ARC Scheduler juga telah diikutsertakan dalam Pameran Inovasi Perguruan Tinggi (PIPT) pada bulan Desember 2019 di Jatinangor Jawa Barat, yang diselenggarakan oleh LLDIKTI 4 Jawa Barat dan Banten. Para pengunjung PIPT 2019 memberikan penilaian baik terhadap program computer ini. Lebih lanjut para pengunjung PIPT menyarankan untuk mengembangkan pula program ini untuk dapat membantu penjadwalan di bidang-bidang lain, sebagai contoh, penjadwalan *shift* kerja karyawan perusahaan atau pabrik atau rumah sakit.



Gambar 4 Tampilan menu utama



Gambar 5 Tampilan menu data

Jadwal Mata Kuliah MTK Angkatan 2015							Jadwal Mata Kuliah MTK Angkatan 2016							
TIME	DAY	SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	TIME	DAY	SUNDAY	MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
0800-0900							0800-0900			Metode Penelitian Matematika (Louise) [114]	Transformasi Geometri (Sonya) [113]			
0900-1000							0900-1000					Transformasi Geometri (Sonya) [113]		Fisika Modern II (Horasdia) [113]
1000-1100						Bahasa Inggris VI (C) [201]	1000-1100				Kajian Masalah Matematika (Sonya) [115]			
1100-1200							1100-1200			Persamaan Diferensial Biasa (Andi) [113]		Program Linier (Sonya) [115]		
1200-1300							1200-1300							
1300-1400							1300-1400							
1400-1500			Analisis Real (Louise) [114]	Metode Numerik (Andi) [113]	Metode Variabel Kompleks (Nora) [113]	SKRIPSI II (Louise) [113]	1400-1500			Program Linier (Sonya) [115]	Persamaan Diferensial Biasa (Andi) [113]	Media Pembelajaran Matematika (Kartini) [114]		
1500-1600		Analisis Real (Louise) [114]	Bimbingan Konseling (Mangadar) [113]	Fungsi Variabel Kompleks (Nora) [113]	Metode Numerik (Andi) [113]		1500-1600				Teori Bilangan (Louise) [113]		Media Pembelajaran Matematika (Kartini) [114]	
1600-1700							1600-1700							

Gambar 6 Tampilan luaran program ARC Scheduler

KESIMPULAN DAN SARAN

Program Komputer *ARC Scheduler* (*Automated Random Coloring Scheduler*) telah dikembangkan menggunakan algoritma Welsh Powell Terbobot berdasarkan teori pewarnaan *vertex* graf, dengan manfaat menghasilkan jadwal seluruh matakuliah yang dikelola Universitas secara otomatis. File input berhasil dibaca oleh program dan file output telah memenuhi kriteria-kriteria yang dirumuskan pada tujuan penelitian, termasuk di antaranya dapat mengakomodir permintaan khusus dosen maupun kebijakan universitas. *Running time* uji coba untuk 1217 sks matakuliah tidak lebih dari 5 detik. Pengguna memberikan respon sangat puas dengan program yang dikembangkan dan memberikan saran agar program penjadwalan juga dikembangkan pada bidang-bidang lain di luar pendidikan.

Dengan demikian disimpulkan bahwa program *ARC Scheduler* merupakan program komputer yang mudah digunakan, memiliki *running time* cepat, menghasilkan jadwal bebas bentrok, dan dapat mengakomodir permintaan khusus dosen dan kebijakan khusus universitas.

Untuk penelitian lanjutan, disarankan beberapa hal berikut

- (1) menggunakan variasi pembobotan yang berbeda,
- (2) mengembangkan versi terpisah yaitu front end dan back end,
- (3) memberikan bukti teoritis bagi persamaan 1 dan persamaan 2,
- (4) mengembangkan program penjadwalan kerja perusahaan atau pabrik menggunakan algoritma Welsh Powell Terbobot.
- (5) Mempertimbangkan jarak antar ruang kuliah dan kondisi gedung bertingkat.

UCAPAN TERIMAKASIH

Penulis mengucapkan terimakasih kepada semua pihak yang mendukung penelitian ini di antaranya Ibu Louise M. Saija, Angelica Marchelia Simbolon, Yohanes Kristanto, Bapak Francis M. Hutabarat, Bapak Horasdia Saragih, Bapak Andrew Pakpahan, dan Panitia Pameran Inovasi Perguruan Tinggi 2019 LLDIKTI 4.

REFERENSI

- [1] Lantiv, "SCHEDULING STUDIO™ 2021: Course Scheduling Software." [Online]. Available: <https://scheduling-studio.lantiv.com/>. [Accessed: 04-May-2020].
- [2] Unitime, "UniTime | University Timetabling Online." [Online]. Available: <https://unitime.org/>. [Accessed: 04-May-2020].
- [3] D. J. A. Welsh and M. B. Powell, "An upper bound for the chromatic number of a graph and its application to timetabling problems," *Comput. J.*, vol. 10, no. 1, pp. 85–86, Jan. 1967, doi: 10.1093/comjnl/10.1.85.
- [4] D. de Werra, "An introduction to timetabling," *Eur. J. Oper. Res.*, vol. 19, no. 2, pp. 151–162, 1985, doi: 10.1016/0377-2217(85)90167-5.
- [5] F. K. S. Dewi, "Pembangunan Perangkat Lunak Pembangkit Jadwal Kuliah dan Ujian Dengan Metode Pewarnaan Graf," *J. Buana Inform.*, 2010, doi: 10.24002/jbi.v1i1.295.
- [6] K. H. Rosen, *Discrete Mathematics Applications and Its Eighth Edition*. New York: McGraw-Hill Education, 2019.
- [7] K. Appel, W. Haken, and J. Koch, "Every planar map is four colorable. Part II: Reducibility," *Illinois J. Math.*, vol. 21, no. 3, pp. 491–567, 1977, doi: 10.1215/IJM/1256049012.
- [8] A. P. Rahadi, "Penjadwalan Matakuliah Menggunakan Pewarnaan Graf Dengan Algoritma Largest First," *J. Pedagogik Mat.*, vol. 2, no. 1, pp. 1–13, Jan. 2019, doi: 10.35974/jpd.v2i1.1067.
- [9] H. Jusuf, "Pewarnaan Graph pada Simpul untuk Mendeteksi Konflik Penjadwalan Kuliah." [Online]. Available: https://www.researchgate.net/publication/277756776_Pewarnaan_Graph_pada_Simpul_untuk_Mendeteksi_Konflik_Penjadwalan_Kuliah. [Accessed: 05-May-2020].
- [10] M. ASLAN, "A Performance Comparison of Graph Coloring Algorithms," *Int. J. Intell. Syst. Appl. Eng.*, 2016, doi: 10.18201/ijisae.273053.
- [11] X. Zhou and T. Nishizeki, "Graph Coloring Algorithms," *IEICE Trans. Inf. Syst.*, 2000.
- [12] N. Cauvery, "Timetable Scheduling using Graph Coloring," *Int. J. P2P Netw. Trends Technol.*, 2011.
- [13] S. Astuti, "Penyusunan Jadwal Ujian Matakuliah Dengan algoritma Pewarnaan Graf Welch Powell| Astuti | JURNAL DIAN," *J. DIAN*, vol. 11, no. 1, p. 68, 2011.
- [14] H. Setiawan, H. Hanafi, L. R. Prilanti, and Kestriilia, "Implementasi Algoritma Kunang-Kunang untuk Penjadwalan Mata Kuliah di Universitas Ma Chung 269."
- [15] E. Crookshanks, *Practical software development techniques : tools and techniques for building enterprise software*. Apress, 2014.
- [16] UNAI, "Universitas Advent Indonesia." [Online]. Available: <https://unai.edu/wp-content/uploads/2018/12/Laporan-Wakil-Rektor-I-Bidang-Akademik-UNAI-T.A.-15-16.pdf>. [Accessed: 06-May-2020].
- [17] Microsoft, "Download Visual C++ Redistributable for Visual Studio 2015 from Official Microsoft Download Center." [Online]. Available: <https://www.microsoft.com/en-us/download/details.aspx?id=48145>. [Accessed: 08-May-2020].
- [18] The Code Blocks Team, "Code::Blocks." [Online]. Available: <http://codeblocks.org/>. [Accessed: 10-May-2020].
- [19] E. B. Pani and A. Simbolon, "20182019 Ganjil - Google Sheets." [Online]. Available: <https://docs.google.com/spreadsheets/d/1Sbfe53t387v1qtf3M72146MSfMltNQSqLmwJcCgQwQ/edit#gid=1433553467>. [Accessed: 06-May-2020].
- [20] M. Al-Obaedy, "Classic-Waterfall-Development-Model.png (616×328)." [Online]. Available: https://www.researchgate.net/profile/Mohaned_Al-Obaidy/publication/259772190/figure/fig1/AS392568656416769@1470607069202/Classic-Waterfall-Development-Model.png. [Accessed: 05-May-2020].