

PENYUSUNAN PENJADWALAN UJIAN MENGGUNAKAN ALGORITMA *RANK BASED ANT SYSTEM*

Ria Fuji Astuti, Neva Satyahadewi, Hendra Perdana

INTISARI

Penyusunan jadwal ujian merupakan salah satu permasalahan yang sering terjadi di suatu Perguruan Tinggi. Penjadwalan ujian merupakan proses penyusunan jadwal pelaksanaan ujian yang menginformasikan sejumlah mata kuliah yang diberikan, ruang tempat belajar, waktu serta mahasiswa yang mengambil mata kuliah tersebut. Salah satu metode untuk menyusun suatu jadwal yaitu Algoritma Ant Colony. Algoritma Ant Colony merupakan metode metaheuristik yang terinspirasi terhadap semut yang berkemampuan untuk berkoordinasi dalam mengumpulkan makanan. Pada penelitian ini digunakan algoritma AS_{Rank} untuk memperoleh jadwal ujian pada Program Studi Sistem Komputer FMIPA Untan. Dengan menginputkan jumlah mata kuliah, jumlah mahasiswa, dan jumlah ruangan, maka diperoleh suatu jadwal ujian yang optimal dengan menggunakan metode algoritma AS_{Rank} .

Kata Kunci: mata kuliah, penjadwalan ujian, algoritma AS_{Rank}

PENDAHULUAN

Penjadwalan terkait pada aktivitas membuat sebuah jadwal. Sebuah jadwal adalah sebuah tabel dari kegiatan-kegiatan yang disusun berdasarkan waktu ditempatkan atau disusunnya [1]. Di setiap semester, beberapa Perguruan Tinggi menghadapi permasalahan yang sama, yaitu bagaimana menjadwalkan ujian semester dengan tidak adanya kendala waktu yang bersamaan. Kendala lain yang dihadapi adalah jumlah ruangan yang terbatas, dan mahasiswa yang mengambil mata kuliah. Proses penjadwalan tersebut tetap memperhatikan sejumlah batasan dan ketentuan yang berlaku sesuai dengan kebijakan Perguruan Tinggi.

Metode optimasi yang diterapkan untuk menyusun jadwal ujian dalam sebuah permasalahan penjadwalan mata kuliah di Program Studi Sistem Komputer. Proses pencarian solusi yang dapat digunakan untuk mendapatkan solusi tersebut adalah dengan menggunakan algoritma berbasis *agent*. Oleh karena itu perlu ditetapkan suatu batasan dalam penyusunan jadwal yang bersifat harus dipenuhi (*hard constraint*) dan tidak harus dipenuhi (*soft constraint*) tetapi tetap menjadi acuan dalam proses pembuatan jadwal. Sebuah solusi jadwal dikatakan layak jika solusi tersebut telah memenuhi semua ketentuan *hard constraint* tanpa ada pelanggaran. Sebuah solusi jadwal dikatakan optimal jika jumlah pelanggaran terhadap *soft constraint* minimum [2]. Salah satu algoritma berbasis *agent* yang dapat digunakan untuk menyelesaikan permasalahan optimasi kombinatorial seperti permasalahan penjadwalan ujian adalah Algoritma *Ant Colony*.

Algoritma *Ant Colony* merupakan algoritma yang terinspirasi dari perilaku semut yang mencari makan hingga semut itu kembali ke koloninya dengan memberikan tanda dengan jejak *pheromone*. Algoritma optimasi *colony* semut telah digunakan untuk menghasilkan penyelesaian yang mendekati optimal [3]. Sesuai dengan analisis untuk masalah penjadwalan ujian, maka akan lebih sesuai jika pembuatan aplikasi menggunakan Algoritma *Ant Colony Optimization* (ACO), khususnya *Rank Based Ant System* AS_{Rank} .

Penelitian ini bertujuan menganalisis Algoritma AS_{Rank} pada permasalahan penjadwalan ujian dan memperoleh jadwal ujian dengan Algoritma AS_{Rank} . Dalam penelitian ini adalah mata kuliah praktikum, kerja praktek dan tugas akhir tidak dijadwalkan. Selain itu, kapasitas ruang ujian dibatasi sebanyak 50 mahasiswa, satu mata kuliah dalam ujian dapat memiliki lebih dari satu ruangan, bila mata kuliah memiliki jumlah mahasiswa melebihi kapasitas ruang. Alokasi waktu dan hari yang digunakan sama, hanya yang membedakan adalah ruangan saja. Dan mata kuliah dengan semester yang sama, dibatasi maksimal terdapat dua ujian dihari yang sama.

Penelitian ini dimulai dengan inialisasi parameter dilakukan pengujian untuk menentukan parameter terbaik. Setelah parameter diperoleh kemudian dicari visibilitasnya, dimana visibilitas tersebut digunakan dalam menghitung peluang titik yang dilewati. Hal ini dilakukan hingga ditemukan rute terpendeknya.

RANK BASED ANT SYSTEM (AS_{Rank})

Setelah semua semut menyelesaikan rutenya, semut akan mengelompokkan diri berdasarkan peringkat (panjang pendeknya rute yang mereka temukan) [3]. Kemudian, semut meng-*update* slot-slot yang telah mereka lalui dengan jumlah *pheromone* yang berbeda, sesuai dengan tingkatannya. *Update pheromone* disini hanya dilakukan pada $(w - 1)$ semut terbaik dan semut yang memiliki solusi *best-so-far* rute yang diperbolehkan menambahkan *pheromone*. Adapun komponen-komponen dari AS_{Rank} untuk menyelesaikan masalah penjadwalan dapat digambarkan sebagai berikut:

- Semut adalah agen perantara untuk mencoba lintasan rute.
- Colony* adalah sekumpulan semut yang digunakan untuk mencoba lintasan-lintasan rute.
- Route* adalah jalur lintasan *slot* yang akan dilalui *colony*.
- Elite* adalah jalur lintasan slot yang ditemukan berdasarkan jarak terpendek.
- Elite trails* adalah jalur lintasan *elite* yang sudah ditambahkan informasi penguapan dan deposit *pheromone*.
- Best global route* adalah jalur lintasan terbaik berdasarkan *elite trails*.

PENETAPAN PARAMETER DAN PHEROMONE AWAL

Dalam menyelesaikan algoritma (AS_{Rank}) maka dilakukan proses inialisasi harga parameter-parameter, yaitu:

- Intensitas *pheromone* antar titik dan perubahan *update pheromone* (τ_{rs}). Nilai (τ_{rs}) akan selalu diperbaharui pada setiap iterasi algoritma, mulai dari iterasi pertama sampai iterasi maksimum yang ditentukan atau telah mencapai hasil yang optimal.
 - Banyaknya titik (n) dan juga koordinat (x, y) atau jarak antar titik (d_{rs}) simetrik nilai
$$d_{rs} = \sqrt{(x_r - x_s)^2 + (y_r - y_s)^2}$$
 - Titik awal dan titik tujuan, dalam kasus ini adalah sama.
 - Parameter penguapan *pheromone* (Q)
 - Tetapan pengendali intensitas jejak semut ($\alpha > 0$). Parameter yang mengontrol bobot (*weight*) relatif dari *pheromone*,
 - Tetapan pengendali visibilitas ($\beta > 0$). Parameter pengendali jarak visibilitas antar titik
$$\eta_{rs} = \frac{1}{d_{rs}}$$
 - Banyaknya semut (m)
 - Tetapan penguapan *pheromone* (ρ), nilainya $0 < \rho \leq 1$. Hal ini untuk mencegah jumlah *pheromone* yang tak terhingga.
 - Jumlah iterasi maksimum (NC_{max}), bersifat tetap selama algoritma berjalan.
-

- j. Parameter yang menyatakan adanya rute terbaik (w)
- k. Peringkat semut (z)
- l. Panjang rute yang dilewati semut ke- z (c^z)
- m. Panjang rute terbaik (c^{bs})

ATURAN TRANSISI

Dalam aturan transisi sebuah semut k yang berada pada titik r akan memilih untuk menuju ke titik s . Hal ini dipilih berdasarkan distribusi peluang dengan persamaan sebagai berikut:

$$P_{rs}^k = \begin{cases} \frac{[\tau_{rs}]^\alpha [\eta_{rs}]^\beta}{\sum_{u \in J_r^k} [\tau_{rs}]^\alpha [\eta_{rs}]^\beta} & , \text{ untuk } s \in J_r^k \\ 0 & , \text{ untuk } s \text{ lainnya} \end{cases} \quad (1)$$

Dengan:

- P_{rs}^k : Peluang dari semut k pada titik r yang memilih untuk menuju ke titik s
- J_r^k : Himpunan titik yang akan dikunjungi oleh semut k yang sedang berada pada titik r .

PERHITUNGAN PANJANG RUTE DAN PENCARIAN JALUR TERPENDEK

Perhitungan panjang rute setiap semut k (C^k) dilakukan setelah setiap semut menyelesaikan rutanya masing-masing. Perhitungan ini dilakukan berdasarkan $tabu_k$ dengan persamaan berikut:

$$C^k = d_{tabu_k(n), tabu_k(1)} + \sum_{s=1}^{n-1} d_{tabu_k(s), tabu_k(s+1)} \quad (2)$$

dengan:

- $d_{tabu_k(s), tabu_k(s+1)}$: Jarak dari titik s sampai titik $s + 1$ pada $tabulist$ yang ditempati oleh semut k
- $d_{tabu_k(n), tabu_k(1)}$: Jarak antara titik n (akhir) dan titik pertama (awal) pada $tabulist$ yang ditempati oleh semut k
- C^k : Panjang rute yang dilalui semut k

UPDATE PHEROMONE

Saat melakukan *update pheromone* hanya $w - 1$ semut terbaik dan semut yang memiliki solusi *best-so-far* yang diperbolehkan meninggalkan *pheromone*. Semut yang ke- z terbaik memberikan kontribusi *pheromone* sebesar $\max(0, w - z)$. Sementara jalur *best-so-far* rute memberikan kontribusi *pheromone* paling besar yaitu sebanyak w , dimana w adalah parameter yang menyatakan adanya rute terbaik dan z adalah peringkat semut. Dalam (AS_{Rank}) aturan *pheromone update*-nya diberikan sebagai berikut:

$$\tau_{rs} = (1 - \rho) \tau_{rs} + \sum_{z=1}^{w-1} (w - z) \Delta \tau_{rs}^z + w \Delta \tau_{rs}^{bs} \quad (3)$$

$$\Delta \tau_{rs} = \frac{1}{C^z} \text{ dan } \Delta \tau_{rs}^{bs} = \frac{1}{C^{bs}}$$

dengan:

- C^z : Panjang rute yang dilalui semut ke- z
 C^{bs} : Panjang rute terbaik.
 $0 < \rho \leq 1$: Sebuah parameter tingkat *evaporasi pheromone*

Colony semut akan meninggalkan *pheromone* pada lintasan antar titik yang dilaluinya. Adanya penguapan dan perbedaan jumlah semut yang lewat, menyebabkan kemungkinan terjadinya perubahan harga intensitas *pheromone* antar titik.

IMPLEMENTASI ALGORITMA RANK BASED ANT SYSTEM

Pengujian program algoritma (AS_{Rank}) ini menggunakan kasus penjadwalan ujian pada sistem perkuliahan di Universitas Tanjungpura Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA) Program Studi Sistem Komputer Tahun Ajaran 2014/2015. Mata kuliah yang akan dibuat penjadwalan terlebih dahulu diurutkan berdasarkan jumlah *slot* terbanyak yang dibutuhkan mata kuliah. Selain berdasarkan jumlah *slot*, mata kuliah tersebut juga diurutkan berdasarkan jumlah mahasiswa.

Tabel 1 Rincian Informasi Mata Kuliah

No	MATA KULIAH	SEM	JUMLAH MHS	TOTAL PESERTA	JUMLAH SLOT
1	Kalkulus B	2	58	110	4
	Kalkulus A		52		
2	Pend. Agama (A)	2	41	57	4
	Pend. Agama (B)	2	9		
	Pend. Agama (C)	2	5		
	Pend. Agama (D)	2	2		
3	Algoritma dan pemograman II (A)	2	34	75	2
	Algoritma dan pemograman II (B)	2	41		
4	Matematika Diskret	2	69	69	2
5	Struktur data (A)	2	28	65	2
	Struktur data (B)	2	37		
6	Aljabar Linear dan Matriks	2	65	65	2
7	Pemograman Visual (A)	4	32	58	2
	Pemograman visual (B)	4	26		
8	Bahasa Inggris II	2	58	58	2
9	Fisika II	2	57	57	2
10	Komputer dan masyarakat	8	56	56	2
11	Sistem operasi (A)	4	28	55	2
	Sistem operasi (B)	4	27		
12	Tehnik dan sistem digital	4	54	54	2
13	Etika profesi	8	52	52	2
14	Elektronika	4	51	51	2
15	Jaringan komputer (A)	4	25	50	2
	Jaringan komputer (B)	4	25		
16	ISBD	4	50	50	1
17	ArsitekTUR komputer	4	45	45	1
18	Sistem mikroprosesor	6	45	45	1

Lanjutan Tabel

No	MATA KULIAH	SEM	JUMLAH MHS	TOTAL PESERTA	JUMLAH SLOT
19	Mikrokontroler	6	44	44	1
20	Tehnik penulisan ilmiah	6	41	41	1
21	Komunikasi data	6	39	39	1
22	Keamanan jaringan	6	33	33	1
23	Pengolahan sinyal digital	6	32	32	1
24	Interaksi manusia dan komputer	8	27	27	1
25	Sistem waktu nyata	8	23	23	1
26	Jaringan syaraf tiruan	6	18	18	1
TOTAL SLOT					45

Berdasarkan Tabel 1, total *slot* yang dibutuhkan dalam penjadwalan mata kuliah adalah sebanyak 45. Kemudian jumlah ruang yang disediakan sebanyak 4. Pembagian waktu/sesi ujian dibagi menjadi 3 yaitu sesi pagi (07.30-09.30), sesi siang (10.00-12.00) dan sesi sore (13.30-15.30). Alokasi lama waktu ujian untuk mata kuliah disamakan yaitu selama 120 menit. Kemudian alokasi hari yang digunakan adalah selama 10 hari kerja (Senin-Jumat). Berdasarkan ketentuan tersebut maka total *slot* yang tersedia adalah sebanyak $4 \times 3 \times 10 = 120$ *slot*.

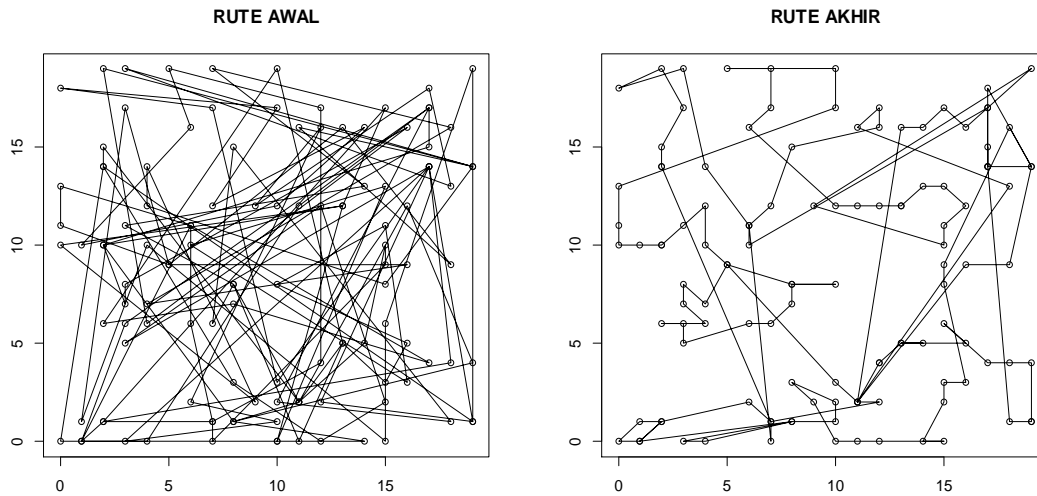
PENENTUAN PARAMETER AWAL

Penyelesaian permasalahan diawali dengan menginisialisasi setiap titik (*slot*), dimana jumlah *slot* sudah ditentukan. Dalam proses penjadwalan ini, tidak mempunyai lokasi/koordinat, sehingga untuk menentukan koordinat tiap *slot* menggunakan koordinat *cartesius* yang ditentukan secara acak. Kemudian koordinat tersebut akan digunakan untuk mencari jarak antar *slot*.

Dalam menentukan rute yang optimal dengan menggunakan algoritma (AS_{Rank}) diperlukan parameter-parameter pendukung. Parameter tersebut untuk proses inisialisasi awal dalam menentukan rute yang optimal. Parameter tersebut untuk proses inisialisasi awal dalam menentukan rute yang optimal. Parameter *alpha* adalah sebuah parameter pengendalian intensitas jejak semut yang memiliki syarat ($\alpha > 0$) Sedangkan parameter *beta* adalah sebuah parameter pengendalian jarak yang memiliki syarat ($\beta > 0$). Pada penyelesaian masalah penjadwalan ujian ini digunakan $\alpha = 1, \beta = 2$. Nilai *pheromone* awal ditentukan terlebih dahulu dan nilainya di set sama untuk semua *node* yaitu sebesar 0.5. Kemudian nilai *theta* awal diperoleh dengan *invers* jarak antar *slot*. Inisialisasi parameter *rho* sebesar 0.5 dan parameter *Q* sebesar 2.5.

Proses berikutnya adalah menentukan rute-rute yang akan diuji berdasarkan jumlah semut, yaitu sama dengan jumlah *slot* yang dibutuhkan sebanyak 45. Hal ini untuk menghindari jumlah semut yang berlebih sehingga akan menimbulkan ketidakefektifan dalam penyelesaian. Penempatan semut pada awal algoritma yaitu dengan setiap alternatif rute hanya boleh ditempati satu semut saja, hal ini untuk menghindari penumpukan semut pada satu jalur yang sama. Peluang *slot-slot* yang akan dikunjungi dari *slot* yang sekarang sedang dikunjungi dihitung menggunakan Persamaan (1). Proses perhitungan tersebut diulang sampai semua *slot* sudah dikunjungi sehingga terbentuk rute yang baru. Setelah setiap semut menghasilkan rute yang baru, maka dilakukan perhitungan jarak untuk menentukan rute terbaik. Rute terbaik ini akan dipakai untuk meng-*update pheromone*. Penguapan *pheromone* dihitung setiap kali semut-semut tersebut melakukan 1 kali perjalanan. Besarnya deposit *pheromone* dan penguapan *pheromone* dihitung berdasarkan Persamaan (2).

Proses mencari rute baru ini diulang sampai jumlah iterasi maksimum yang ditentukan atau sudah mencapai iterasi maksimum. Bagian akhir dari penelitian ini adalah menampilkan rute terbaik yang dihasilkan dan grafik nilai rute terbaik.



Gambar 1 Output Rute awal menggunakan AS_{Rank}

Gambar 2 Output Rute Terbaik menggunakan AS_{Rank}

Rute awal yang terbentuk dari algoritma AS_{Rank} :

97 → 44 → 81 → 23 → 90 → 21 → 88 → 49 → 91 → 53 → 48 → 47 → 42 → 54 →
 34 → 59 → 82 → 17 → 29 → 1 → 56 → 32 → 117 → 27 → 85 → 65 → 111 →
 18 → 61 → 100 → 83 → 89 → 31 → 96 → 72 → 14 → 58 → 37 → 108 → 30 → 6 →
 4 → 13 → 51 → 84 → 15 → 16 → 24 → 113 → 50 → 66 → 45 → 41 → 109 → 74 →
 114 → 118 → 77 → 25 → 93 → 119 → 20 → 99 → 8 → 92 → 80 → 75 → 36 → 94 →
 11 → 26 → 28 → 67 → 104 → 40 → 86 → 52 → 7 → 73 → 78 → 98 → 12 → 65 →
 101 → 102 → 103 → 64 → 3 → 110 → 2 → 35 → 38 → 115 → 43 → 62 → 5 → 39 →
 71 → 22 → 60 → 69 → 112 → 107 → 105 → 10 → 33 → 87 → 63 → 55 → 79 → 9 →
 70 → 116 → 106 → 97

Rute terbaik yang terbentuk dari algoritma AS_{Rank} :

97 → 28 → 40 → 22 → 96 → 35 → 118 → 29 → 119 → 52 → 76 → 112 → 8 → 61 →
 34 → 62 → 41 → 45 → 43 → 83 → 44 → 55 → 93 → 59 → 114 → 103 → 25 → 13 →
 18 → 95 → 110 → 49 → 16 → 31 → 20 → 65 → 82 → 106 → 88 → 19 → 4 → 98 →
 81 → 50 → 66 → 102 → 57 → 36 → 101 → 85 → 71 → 21 → 92 → 87 → 58 → 37 →
 105 → 11 → 77 → 32 → 86 → 63 → 39 → 113 → 9 → 68 → 48 → 117 → 84 → 17 →
 109 → 47 → 90 → 23 → 89 → 67 → 100 → 14 → 120 → 7 → 1 → 99 → 15 → 70 →
 27 → 111 → 91 → 107 → 46 → 3 → 10 → 64 → 38 → 116 → 94 → 54 → 30 → 51 →
 75 → 79 → 74 → 115 → 26 → 80 → 56 → 24 → 78 → 108 → 12 → 72 → 73 → 33 →
 69 → 104 → 6 → 53 → 2 → 42 → 60 → 5 → 97

Urutan mata kuliah yang akan diinput ke dalam *slot* berdasarkan kriteria yang telah dijelaskan pada Tabel 2 Proses penginputan mata kuliah ke dalam *slot* berdasarkan rute yang terbentuk.

Tabel 2 Output Penjadwalan Ujian Minggu Pertama

R U A N G 1	Sesi	Hari				
		Senin	Selasa	Rabu	Kamis	Jumat
	07.30-09.30	Sistem Operasi B		Komputer dan masyarakat B	Etika Profesi B	Algoritma dan pemograman II B1
	10.00-12.00		Agama B	Struktur Data A	ISBD	
	13.30-15.30		Sistem Waktu Nyata	Kalkulus B1	Matematika Diskret B	Mikrokontroler
R U A N G 2	Sesi	Hari				
		Senin	Selasa	Rabu	Kamis	Jumat
	07.30-09.30	Sistem Operasi A	Elektronika B	Komputer dan Masyarakat A	Etika Profesi A	Algoritma dan pemograman II A2
	10.00-12.00		Agama A	Struktur Data B		Interaksi Manusia dan Komputer
	13.30-15.30	Pemograman Visual B	Teknik Penulisan Ilmiah	Kalkulus A1	Matematik Diskret A	Bahasa Inggris II B
R U A N G 3	Sesi	Hari				
		Senin	Selasa	Rabu	Kamis	Jumat
	07.30-09.30	Tehnik dan Sistem Digital A	Elektronika B	Jaringan Syaraf Tiruan	Aljabar Linear dan Matriks A	Algoritma dan pemograman II A1
	10.00-12.00		Agama C	Komunikasi Data	Jaringan Komputer A	
	13.30-15.30	Pemograman Visual A	Etika Profesi B	Kalkulus A2	Fisika II B	Bahasa Inggris II A
R U A N G 4	Sesi	Hari				
		Senin	Selasa	Rabu	Kamis	Jumat
	07.30-09.30	Tehnik dan Sistem Digital B	Sistem Mikroprosesor	Arsitektur Komputer	Aljabar Linear dan Matriks B	Algoritma dan pemograman II B2
	10.00-12.00	Keamanan Jaringan	Agama D		Jaringan Komputer B	
	13.30-15.30	Pengolahan Sinyal Digital	Etika Profesi A	Kalkulus B2	Fisika II A	

PENUTUP

Algoritma AS_{Rank} dapat memberikan solusi untuk masalah penjadwalan ujian mata kuliah dengan menghasilkan suatu jadwal ujian mata kuliah yang optimal. Suatu jadwal dapat dikatakan optimal apabila tidak terjadi ujian pada mahasiswa yang sama untuk mata kuliah yang berbeda. Penyelesaian permasalahan penjadwalan ujian mata kuliah pada Program Studi Sistem Komputer FMIPA Untan dengan menggunakan algoritma AS_{Rank} berlangsung dalam jangka waktu 5 hari. Selanjutnya, ujian dibagi menjadi 3 sesi waktu yaitu sesi I pukul 07.30-09.30, sesi II 10.00-12.00, dan sesi III 13.30-15.30 dengan ketersediaan ruangan sebanyak 4 ruangan. Pada hari Jumat ujian dilaksanakan dalam dua sesi yaitu sesi I pukul 07.30-09.30 dan sesi II pukul 13.30-15.30.

DAFTAR PUSTAKA

- [1]. Anggasyati, 2008, Penerapan Algoritma Rank Based Ant System (AS_{Rank}) Pada Optimasi Penjadwalan Sumber Daya Proyek, *Jurnal Teknik Informatika Institut Teknologi Telkom*, Bandung.
- [2]. Saragih, H., Aplikasi Sistem Perangkat Lunak Menggunakan Algoritma Ant untuk Mengatur Penjadwalan Kuliah, *Jurnal Tehnik dan Ilmu Komputer*, Volume 01 Nomor 03, 241-254.
- [3]. Purnomo, H.D., 2004, *Cara Mudah Belajar Metode Optimasi Metaheuristik Menggunakan Matlab*, Gava media: Yogyakarta.

RIA FUJI ASTUTI

: FMIPA Untan Pontianak, riafujiastuti59@gmail.com

NEVA SATYAHADEWI

: FMIPA Untan Pontianak, neva.satya@math.untan.ac.id

HENDRA PERDANA

: FMIPA Untan Pontianak, hendra.perdana@math.untan.ac.id