

SKRIPSI

**OTOMATISASI SISTEM PENJADWALAN UJIAN
AKHIR SEMESTER MENGGUNAKAN METODE
ALGORITMA GENETIKA**

**(Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah
Magelang)**



AGUNG DWI NOFIANTO

NPM : 14.0504.0045

**PROGRAM STUDI TEKNIK INFORMATIKA S1
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAGELANG
TAHUN 2020**

SKRIPSI

**OTOMATISASI SISTEM PENJADWALAN UJIAN
AKHIR SEMESTER MENGGUNAKAN METODE
ALGORITMA GENETIKA**

**(Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah
Magelang)**



AGUNG DWI NOFIANTO

NPM : 14.0504.0045

**PROGRAM STUDI TEKNIK INFORMATIKA S1
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAGELANG
TAHUN 2020**

SKRIPSI

**OTOMATISASI SISTEM PENJADWALAN UJIAN
AKHIR SEMESTER MENGGUNAKAN METODE
ALGORITMA GENETIKA**

**(Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah
Magelang)**

Disusun Sebagai Salah Satu Syarat Memperoleh Gelar Sarjana Komputer
(S.Kom) Program Studi Teknik Informatika Jenjang Strata Satu (S-1) Fakultas
Teknik Universitas Muhammadiyah Magelang



AGUNG DWI NOFIANTO

NPM. 14.0504.0045

**PROGRAM STUDI TEKNIK INFORMATIKA S1
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAGELANG
TAHUN 2020**


HALAMAN PENEGASAN

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Agung Dwi Nofianto

NPM : 14.0504.0045

Magelang, 14 Februari 2020



AGUNG DWI NOFIANTO
NPM. 14.0504.0045

PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Agung Dwi Nofianto

NPM : 14.0504.0045

Program Studi : Teknik Informatika S1

Fakultas : Teknik

Alamat : Bayem RT/01 RW/04 Kutoarjo, Purworejo

Judul Skripsi : OTOMATISASI SISTEM PENJADWALAN UJIAN AKHIR
SEMESTER MUNGGUNAKAN ALGORITMA GENETIKA
(Studi Kasus : Fakultas Ilmu Kesehatan Universitas
Muhammadiyah Magelang)

Dengan ini menyatakan bahwa Skripsi ini merupakan hasil karya sendiri dan bukan merupakan plagiat dari hasil karya orang lain. Dan bila di kemudian hari terbukti bahwa karya ini merupakan plagiat, maka saya bersedia menerima sanksi administrasi maupun sanksi apapun.

Demikian surat pernyataan ini saya buat dengan penuh kesadaran dan sebenarnya serta penuh tanggung jawab.

Magelang, 14 Februari 2020

Yang menyatakan,



AGUNG DWI NOFIANTO

NPM. 14.0504.0045

HALAMAN PENGESAHAN
SKRIPSI
OTOMATISASI SISTEM PENJADWALAN UJIAN AKHIR
SEMESTER MENGGUNAKAN METODE ALGORITMA
GENETIKA

(Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang)

Disusun Oleh :

AGUNG DWI NOFIANTO

NPM. 14.0504.0045

Telah dipertahankan di depan Dewan Penguji

Pada Tanggal 14 Februari 2020

Susunan Dewan Penguji

Pembimbing I

Pembimbing II


Nugroho Agung Prabowo, ST., M.Kom.

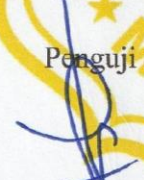
NIDN. 0624077302


Emilyya Uly Artha, M.Kom.

NIDN. 0512128101

Penguji I

Penguji II


Purwono Hendradi, M.Kom.

NIDN. 0624077101


Endah Ratna Arumi, M.Cs.

NIDN. 0601129001


Skripsi ini telah diterima sebagai salah satu persyaratan

untuk memperoleh gelar Sarjana Komputer

Tanggal, 14 Februari 2020

Dekan




Yun Arifatul Fatimah, ST., MT., Ph.D

NIK. 987408139

KATA PENGANTAR


Puji syukur kehadiran Allah SWT, atas segala rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan lancar. Skripsi ini disusun sebagai salah satu syarat untuk mencapai gelar Sarjana Komputer di Program Studi Teknik Informatika S1 Fakultas Teknik Universitas Muhammadiyah Magelang.

Penyelesaian Skripsi ini banyak memperoleh bantuan dan bimbingan dari berbagai pihak. Oleh karena itu, diucapkan terima kasih sebesar – besarnya kepada :

1. Allah SWT yang telah memudahkan segala urusan saya, sehingga dapat menyelesaikan skripsi ini.
2. Kedua orang tua dan keluarga yang telah memberikan dukungan baik secara moril maupun materi hingga terselesaikannya skripsi ini.
3. Yun Arifatul Fatimah, S.T., M.T., Ph.D. selaku Dekan Fakultas Teknik Universitas Muhammadiyah Magelang.
4. Agus Setiawan, M.Eng. selaku Ketua Program Studi Teknik Informatika S1 Universitas Muhammadiyah Magelang.
5. Nugroho Agung Prabowo, S.T., M.Kom dan Emilya Uly Artha, M.Kom selaku Dosen pembimbing, pendamping yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan penyusunan skripsi ini.
6. Seluruh Dosen Fakultas Teknik Universitas Muhammadiyah Magelang yang telah memberikan ilmu dan pengetahuan yang bermanfaat.
7. Keluarga besar Kos Rustam, Mas Pras, dan Bu Tika Sekeluarga, yang selalu memberikan semangat dan doanya.

Akhir kata, semoga Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu yang tidak dapat disebutkan namanya satu persatu dan semoga Skripsi ini membawa manfaat bagi semua pihak.

Magelang, 14 Februari 2020


AGUNG DWI NOFIANTO
NPM. 14.0504.0045

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Muhammadiyah Magelang, yang bertandatangan di bawah ini :

Nama : Agung Dwi Nofianto
NPM : 14.0504.0045
Program Studi : Teknik Informatika S1
Fakultas : Teknik
Jenis karya : Skripsi

Menyatakan bahwa demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Fakultas Teknik Universitas Muhammadiyah Magelang **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty-Free Right)** atas karya ilmiah yang berjudul :


Otomatisasi Sistem Penjadwalan Ujian Akhir Semester Menggunakan Metode Algoritma Genetika (Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang)

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Fakultas Teknik Universitas Muhammadiyah Magelang berhak menyimpan, mengalihmedia/memformatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan Skripsi tersebut selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini dibuat dengan sebenar-benarnya tanpa paksaan dari pihak manapun.

Dibuat di : Magelang

Pada tanggal : 14 Februari 2020

Yang menyatakan


AGUNG DWI NOFIANTO
NPM. 14.0504.0045

ABSTRAK

OTOMATISASI SISTEM PENJADWALAN UJIAN AKHIR SEMESTER MENGGUNAKAN METODE ALGORITMA GENETIKA (STUDI KASUS : FAKULTAS ILMU KESEHATAN UNIVERSITAS MUHAMMADIYAH MAGELANG)

Nama : Agung Dwi Nofianto

Pembimbing : 1. Nugroho Agung Prabowo, S.T., M.Kom
2. Emilya Ully Artha, M.Kom

Permasalahan pembuatan penjadwalan ujian merupakan permasalahan yang membutuhkan banyak waktu dalam proses penyelesaiannya dan masalah ini sering dihadapi oleh Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang dikarenakan proses penjadwalan ujian akhir semester masih dilakukan secara manual menggunakan *microsoft excel*. Tujuan dari penelitian adalah menemukan cara menjadwalkan sejumlah komponen yang terdiri dari mata kuliah, ruang, waktu dan pengawas dengan batasan-batasan tertentu. Untuk mempermudah proses otomatisasi penjadwalan penulis menggunakan metode Algoritma Genetika, yaitu Algoritma yang menggunakan prinsip genetika dan seleksi alamiah dalam proses optimisasi penjadwalan. Algoritma ini dimulai dengan beberapa kombinasi populasi, yang terdiri dari kromosom-kromosom yang berisi data mata kuliah, waktu, ruangan dan pengawas, dan dilakukan proses seleksi untuk mendapatkan solusi terbaik. Hasil yang dicapai dari penelitian ini berupa jadwal yang optimum dimana jadwal yang diolah dengan jumlah kromosom 50 dan maksimal generasi 100 menghasilkan nilai *fitness* 1.

Kata kunci : *Penjadwalan, Algoritma genetika , Otomatis*

ABSTRACT

AUTOMATION OF SEMESTER FINAL EXAM SCHEDULING SYSTEM USING THE GENETIK ALGORITHM METHOD (CASE STUDY : FACULTY OF HEALTH SCIENCES, MUHAMMADIYAH UNIVERSITY, MAGELANG)

Nama : Agung Dwi Nofianto

Pembimbing : 1. Nugroho Agung Prabowo, S.T., M.Kom

2. Emilya Ully Artha, M.Kom

The problem of making exam scheduling is a problem that requires a lot of time in the process, and this problem is often faced by the Faculty of Health Sciences, University of Muhammadiyah Magelang since the final semester exam scheduling process is still done manually using Microsoft Excel. The purpose of the research is to find a way the scheduling to describe all components consisting of courses, classroom, time and supervisors with certain limitations. To facilitate the scheduling automation process the author used the Genetic Algorithm method, which is an algorithm that uses the principles of genetics and natural selection in the process of optimizing scheduling. This algorithm starts with various combinations, which consist of chromosomes containing course data, time, classroom and supervisor, and the selection process is carried out to get the best solution. The results obtained from this study consist of an optimal schedule in which the schedule is processed with the number of chromosomes of 50 and the maximum generation of 100 produces a fitness value of 1.

Keywords: *Scheduling, Genetic Algorithms, Automation*

DAFTAR ISI

HALAMAN KULIT MUKA	ii
HALAMAN JUDUL.....	iii
HALAMAN PENEGASAN.....	iv
PERNYATAAN KEASLIAN.....	v
HALAMAN PENGESAHAN.....	vi
KATA PENGANTAR	vii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	viii
ABSTRAK	ix
ABSTRACT	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL.....	xv
BAB 1	1
A. Latar Belakang	1
B. Perumusan Masalah	3
C. Batasan Masalah.....	3
D. Tujuan Penelitian	4
E. Manfaat Penelitian	4
BAB II.....	5
A. Penelitian Relevan.....	5
B. Penjelasan Teoritis Masing – Masing Variabel Penelitian	7
C. LANDASAN TEORI.....	20
BAB III	21
A. Metode Penelitian.....	21
B. Analisis Sistem.....	31
C. Perancangan Unified Modeling Language Diagram.....	46
D. Perancangan Basis Data	50
E. Desain Database	55
F. Perancangan Interface	58
BAB IV	63

IMPLEMENTASI DAN PENGUJIAN	63
A. IMPLEMENTASI SISTEM.....	63
B. PENGUJIAN.....	90
BAB V.....	100
A. HASIL.....	100
B. Pembahasan.....	108
BAB VI.....	110
A. KESIMPULAN.....	110
B. SARAN	110
DAFTAR PUSTAKA	111

DAFTAR GAMBAR

Gambar 2. 1 Proses Pindah Silang	15
Gambar 2. 2 Proses Mutasi	18
Gambar 2. 3 Siklus Umum Algoritma Genetika.....	19
Gambar 3. 1 Flowchart Alur Penelitian	21
Gambar 3. 2 Gambar Alur pembuatan jadwal yang seharusnya.....	22
Gambar 3. 3 Gambar Alur pembuatan jadwal ujian yang berjalan.....	23
Gambar 3. 4 Flowchart sistem berjalan.....	33
Gambar 3. 5 Gambar Flowchart Siklus Pemecahan Masalah dengan Algoritma Genetika	36
Gambar 3. 6 Use Case Diagram.....	47
Gambar 3. 7 Activity Diagram.....	48
Gambar 3. 8 Sequence Diagram Login	48
Gambar 3. 9 Sequence Diagram Penjadwalan Ujian dengan Algoritma Genetika.....	49
Gambar 3. 10 Class Diagram	49
Gambar 3. 11 Gambar ERD	53
Gambar 3. 12 Relasi Matakuliah dan Jadwal.....	54
Gambar 3. 13 Relasi Matakuliah dan Jadwal.....	54
Gambar 3. 14 Relasi Waktu Ujian dan Jadwal	54
Gambar 3. 15 Relasi Ruangan dan Jadwal.....	55
Gambar 3. 16 Relasi Jam dan Waktu	55
Gambar 3. 17 Gambar Tampilan Halaman User.....	59
Gambar 3. 18 Gambar Tampilan Halaman Matakuliah.....	60
Gambar 3. 19 Gambar Halaman Pengawas.....	60
Gambar 3. 20 Gambar Tampilan Halaman Waktu.....	61
Gambar 3. 21 Gambar Tampilan Halaman Ruangan.....	61
Gambar 3. 22 Gambar Tampilan Penjadwal Ujian	62
Gambar 3. 23 Gambar Tampilan Jadwal Ujian.....	62
Gambar 4. 1 gambar implementasi tabel user.....	64
Gambar 4. 2 gambar implementasi tabel matakuliah.....	64
Gambar 4. 3 gambar implementasi tabel waktu.....	65
Gambar 4. 4 gambar implementasi tabel sesi.	65
Gambar 4. 5 gambar implementasi tabel ruang.	65
Gambar 4. 6 gambar implementasi tabel pengawas.....	66
Gambar 4. 7 gambar implementasi tabel jadwal ujian akhir semester.....	66
Gambar 4. 8 gambar login admin.....	66
Gambar 4. 9 Implementasi Antarmuka Beranda.....	67
Gambar 4. 10 Implementasi Antarmuka Data Matakuliah	68
Gambar 4. 11 Implementasi Antarmuka Tambah Matakuliah.....	68
Gambar 4. 12 Implementasi Antarmuka Ubah Matakuliah	69
Gambar 4. 13 Implementasi Antarmuka Data Pengawas.....	69
Gambar 4. 14 Implementasi Antarmuka Tambah Data Pengawas.	70
Gambar 4. 15 Implementasi Antarmuka Ubah Data Pengawas.....	70
Gambar 4. 16 Implementasi Antarmuka Tampilan Data Ruang.....	71
Gambar 4. 17 Implementasi Antarmuka Tambah Data Ruang	71
Gambar 4. 18 Implementasi Antarmuka Ubah Data Ruang	72

Gambar 4. 19 Implementasi Antarmuka Input Data Jam.....	72
Gambar 4. 20 Implementasi Antarmuka Tambah Data Sesi.....	73
Gambar 4. 21 Implementasi Antarmuka Ubah Data Sesi	73
Gambar 4. 22 Implementasi Antarmuka Tampilan Data Waktu.....	74
Gambar 4. 23 Implementasi Antarmuka Tambah Data Hari	74
Gambar 4. 24 Implementasi Antarmuka Ubah Data Hari.....	75
Gambar 4. 25 Implementasi Antarmuka Input Data Proses Penjadwalan	75
Gambar 4. 26 <i>script</i> program cek login	76
Gambar 4. 27 <i>script</i> Data Matakuliah	77
Gambar 4. 28 <i>script</i> Data Waktu.....	78
Gambar 4. 29 <i>script</i> Data Jam.....	80
Gambar 4. 30 <i>Script</i> Data Ruang	81
Gambar 4. 31 <i>Script</i> Data Pengawas.....	82
Gambar 4. 32 <i>Script</i> Data Proses Penjadwalan	89
Gambar 4. 33 <i>Script</i> Hasil Jadwal.....	90
Gambar 5. 1 Hasil Jadwal Uji ke satu dengan Aplikasi Algoritma Genetika	100
Gambar 5. 2 Hasil penjadwalan dengan jumlah kromosom 10 dan maksimal generasin 25.	101
Gambar 5. 3 Hasil Jadwal Uji ke dua dengan Aplikasi Algoritma Genetika.	101
Gambar 5. 4 Hasil penjadwalan dengan jumlah kromosom 20 dan maksimal generasin 35.	102
Gambar 5. 5 Hasil Jadwal Uji ke tiga dengan Aplikasi Algoritma Genetika	102
Gambar 5. 6 Hasil penjadwalan dengan jumlah kromosom 30 dan maksimal generasin 35.	103
Gambar 5. 7 Hasil Jadwal Uji ke empat dengan Aplikasi Algoritma Genetika.	103
Gambar 5. 8 Hasil penjadwalan dengan jumlah kromosom 50 dan maksimal generasin 100.	104

DAFTAR TABEL

Tabel 3. 1 Contoh Jadwal UAS Prodi Keperawatan S1	24
Tabel 3. 2 Tabel Jadwal UAS Prodi D3 Keperawatan.....	26
Tabel 3. 3 Tabel Lama Hari Pelaksanaan Ujian.....	26
Tabel 3. 4 Tabel Waktu Pelaksanaan UAS Prodi Keperawatan S1	27
Tabel 3. 5 Tabel Waktu Pelaksanaan UAS Prodi Keperawatan D3	27
Tabel 3. 6 Tabel Waktu Pelaksanaan UAS Prodi Keperawatan D3	28
Tabel 3. 7 Tabel Waktu Pelaksanaan UAS Prodi Keperawatan S1	29
Tabel 3. 8 Tabel Contoh Pengawas yang mengawasi dalam sesi yang sama dan berbeda ruang dalam satu sesi.....	34
Tabel 3. 9 Pengkodean Pegawai	37
Tabel 3. 10 Tabel Pengkodean Jam.....	37
Tabel 3. 11 Tabel Pengkodean Waktu	38
Tabel 3. 12 Tabel Pengkodean Ruang.....	38
Tabel 3. 13 Tabel Pengkodean Makul Keperawatan S1	39
Tabel 3. 14 Tabel Pengkodean Makul Keperawatan D3.....	40
Tabel 3. 15 Tabel Representasi Kromosom Pengawas Hari ke 1	42
Tabel 3. 16 Tabel Proses Evaluasi	44
Tabel 3. 17 Tabel Perhitungan Nilai Fitness pada child (offspring).....	45
Tabel 3. 18 Tabel Kumpulan Individu & offspring pada hari 1.....	45
Tabel 3. 19 Tabel Individu Terpilih pada Generasi Berikutnya.....	46
Tabel 3. 20 Tabel Entitas	50
Tabel 3. 21 Tabel Penentuan Entitas dan Atribut	50
Tabel 3. 22 Tabel Rancangan Tabel Matakuliah	55
Tabel 3. 23 Tabel Rancangan Tabel Pengawas.....	56
Tabel 3. 24 Tabel Rancangan Tabel Ruang	56
Tabel 3. 25 Tabel Waktu Ujian	57
Tabel 3. 26 Tabel Rancangan Tabel Jadwal.....	57
Tabel 5. 1 Tabel Penentuan Skor	106
Tabel 5. 2 Tabel Presentase Kriteria Skor.....	107
Tabel 5. 3 Tabel Hasil Kueisioner	107

BAB 1

PENDAHULUAN

A. Latar Belakang

Pembuatan jadwal ujian akhir semester adalah kegiatan rutin yang dilakukan oleh perguruan tinggi pada tiap semester. Di perguruan tinggi, program penjadwalan ujian merupakan salah satu hal penting dalam proses belajar mengajar, karena semua kegiatan ujian semester bergantung pada jadwal yang ada, sehingga harus disusun dengan benar dan diperbaiki setiap akan diadakan ujian pada tiap semester, sehingga nantinya tidak terjadi benturan jadwal ujian semester. Pembuatan jadwal Ujian membutuhkan banyak waktu untuk menghasilkan jadwal yang baik dan tepat sasaran.

Sistem penjadwalan pada sebuah Universitas terutama penjadwalan ujian semester, merupakan hal yang kritis dan perlu diperhitungkan dengan baik karena harus mempertimbangkan berbagai keterbatasan dan syarat. Jadwal ujian semester di universitas harus dibuat tiap semester, dan jadwal tersebut tidak akan dapat digunakan kembali pada semester yang sama ditahun berikutnya. Karena itulah, permasalahan penjadwalan mata kuliah merupakan permasalahan yang kompleks.

Pembuatan jadwal Ujian di Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang dirancang oleh ketua program studi. Penjadwalan ujian semester di Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang terdapat pada terbatasnya ruang ujian dan pengawas ujian, akan tetapi jumlah kelas perkuliahan sangat banyak. Pembuatan jadwal ujian selama ini masih menggunakan cara manual dengan menggunakan perangkat lunak Microsoft Excel yang rentan akan kesalahan. Penjadwalan yang manual menggunakan Microsoft Exel akan memakan waktu yang lama karena banyaknya factor yang harus dipertimbangkan.

Faktor yang menjadi pertimbangan dalam merancang jadwal ujian semester antara lain adalah jumlah mata kuliah, jumlah ruangan kelas, jumlah pengawas ujian dan waktu ujian. Permasalahan penjadwalan ujian dua prodi

yang memakan waktu lama karena masing-masing prodi membuat jadwal masing-masing sehingga salah satu prodi menunggu jadwal lainnya. Distribusi jadwal ujian diharapkan dapat merata setiap harinya untuk setiap kelas dan merata juga bagi pengawas ujian agar tidak menumpuk jadwal ujian dalam satu hari. Selain itu sebuah jadwal dirancang sedemikian rupa agar pengawas ujian tidak mengawasi ujian yang berbeda pada waktu yang bersamaan. Sebuah penjadwalan dikatakan baik apabila dapat memberikan solusi terhadap factor-faktor tersebut, tentu saja tidak semua sistem penjadwalan memiliki permasalahan yang sama, karena disesuaikan dengan kebutuhan instansi ataupun tempat yang akan dilakukan penjadwalan.

Permasalahan penjadwalan diatas sering terjadi di Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang. Permasalahan-permasalahan yang terjadi diharapkan mampu memecahkan setiap detail masalah yang muncul dalam proses penjadwalan di Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang. Banyak metode yang dapat digunakan dalam menyelesaikan masalah penjadwalan seperti metode *graph coloring*, metode *fuzzy logic*, metode algoritma genetika, metode *auto generate time table*, metode *tabu search*, dan *Partical swarm optimization*. Penelitian ini menggunakan metode Algoritma Genetika untuk memecahkan masalah penjadwalan, karena dengan metode algoritma genetika merupakan salah satu metode optimasi yang kuat dan bisa digunakan pada berbagaimacam studi kasus, baik kasus yang sederhana hingga kasus yang rumit karena menggunakan teori evolusi (Pradyana, Sunaryono, dan munif 2012)

Selain itu dalam Algoritma genetika semua modul berjalan dengan baik dan jadwal yang dihasilkan lebih presisi dengan pengalokasian jam ujian, ruang kelas dan pengawas ujian yang tepat. Berdasarkan hasil ini maka dapat disimpulkan bahwa system yang dibangun telah berjalan dengan baik sesuai dengan kebutuhan yang telah diidentifikasi diawal (Afrizal Nehemia Toscan, Rusdianto Roestam 2017). Penggunaan metode algoritma genetika pada penelitian ini diharapkan dapat membantu mengoptimalakan

dan memenuhi segala permasalahan dalam ujian semester. Selain itu dapat membuat system melakukan proses penjadwalan secara otomatis.

Berdasarkan latar belakang diatas, maka penelitian ini berusaha merancang sebuah system penjadwalan menggunakan metode algoritma genetika dan mengangkatnya menjadi sebuah penelitian yang berjudul Otomatisasi Sistem Penjadwalan Ujian Semester Menggunakan Metode Algoritma Genetika (Studi Kasus : Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang).

B. Perumusan Masalah

Berdasarkan latar belakang diatas maka dapat dirumuskan masalah sebagai berikut:

1. Bagaimana jadwal ujian dua prodi dibuat secara bersamaan menggunakan penjadwalan ujian otomatis?
2. Bagaimana penjadwalan ujian otomatis dapat menjamin seorang pengawas tidak mengawas ujian dalam waktu yang bersamaan?
3. Bagaimana penjadwalan ujian otomatis menjamin 1 matakuliah hanya ujian satukali dalam 1 hari?
4. Bagaimana algoritma genetika membantu menyelesaikan permasalahan yang ada dalam penjadwalan ujian otomatis?

C. Batasan Masalah

Dalam suatu penelitian dapat muncul permasalahan yang meluas. Agar permasalahan dalam sebuah penelitian tidak meluas maka dibutuhkan suatu batasan masalah antara lain :

1. Sampel data yang digunakan adalah Program Studi Keperawatan S1 dan Program Studi Keperawatan D3 Semester Genap tahun akademik 2018/2019.
2. Diasumsikan waktu ujian per-matakuliah adalah 75 sampai 90 menit.
3. Diasumsikan pengawas bersedia mengawasi di ruang manapun yang tersedia dan pada waktu yang telah ditentukan.

4. Sistem penjadwalan ini hanya untuk Program Studi dengan kelas regular
5. Metode yang digunakan adalah Algoritma Genetika.

D. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Membuat penjadwalan ujian otomatis dua prodi secara bersamaan dengan menjamin seorang pengawas tidak mengawasi ujian dalam waktu bersamaan.
2. Membuat sistem penjadwalan ujian otomatis dengan menerapkan metode algoritma genetika.

E. Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat membantu proses penjadwalan yang dilakukan di Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang mempermudah pembuatan jadwal ujian.

BAB II

TINJAUAN PUSTAKA

A. Penelitian Relevan

1. Penelitian yang dilakukan oleh Afrizal Nehemia Toscany, Rusdianto Roestam (2017) yang berjudul Pengembangan Sistem Penjadwalan Kuliah Menggunakan Algoritma Genetika (Studi Kasus : Pascasarjana Universitas Jambi) menyatakan bahwa Saat ini penjadwalan kuliah di Pascasarjana Universitas Jambi masih terdapat beberapa kendala seperti pembuatan jadwal yang relatif lama, jadwal yang bentrok dan jadwal kuliah yang tidak sesuai dengan waktu ketersediaan dosen. Oleh karena itu penulis memberikan solusi berupa sistem penjadwalan kuliah menggunakan algoritma genetik dengan bahasa pemrograman PHP dan DBMS MySQL. Untuk metode pengembangan sistem yang digunakan adalah waterfall. Dimana hasil jadwal kuliah yang dihasilkan dengan sistem yang telah dibangun ini menjadi lebih presisi dengan pengalokasian jam mengajar, ruang kelas dan dosen yang tepat. Dari hasil pengujian juga dapat disimpulkan bahwa implementasi algoritma genetik sudah sesuai dengan kebutuhan untuk mendukung proses penjadwalan, sehingga penyusunan jadwal bisa dilakukan dengan lebih cepat.
2. Penelitian yang dilakukan oleh Lian Aga Aditya, Windha Mega PD (2017) yang berjudul Implementasi Algoritma Genetika Untuk Penjadwalan Mata Pelajaran Pada LMS Getsmart menyatakan bahwa Penjadwalan dalam kegiatan belajar mengajar di institusi pendidikan merupakan hal yang rumit jika dilakukan secara manual. Karena untuk mengatur jadwal melibatkan banyak komponen seperti jumlah guru, kelas, dan ketersediaan waktu sesuai jadwal yang bisa digunakan. Dengan menganalisa permasalahan ini, sebuah sistem yang dapat membantu kita untuk mengoptimalkan dan secara otomatis menghasilkan jadwal kerja sangat dibutuhkan. Pendekatan untuk mengoptimalkan proses penentuan waktu adalah dengan menggunakan Algoritma Genetika. Algoritma ini merupakan pendekatan komputasi dari prinsip seleksi alam dari teori evolusi Charles Darwin dan teori pewarisan Mendel. Studi kasus penelitian

ini dilakukan di LMS GETSMART yang merupakan platform e-learning yang belum memiliki fitur penetapan waktu. Melalui metode algoritma genetika dan dengan menggunakan data kendala yang ada, sistem ini mampu menghasilkan jadwal yang paling optimal berdasarkan nilai *fitnessnya*, karena algoritma ini menggunakan kombinasi prinsip seleksi alam dan prinsip ketegaran untuk mendapatkan hasil seperti yang diharapkan.

3. Penelitian yang dilakukan oleh Ivan, Stephanus Raphael, Halim Agung (2018) yang berjudul Aplikasi Penjadwalan mata kuliah di SMAN 31 Menggunakan Algoritma Genetika Berbasis Web menyatakan bahwa Sekolah Menengah Atas Negeri 31 memiliki kelas sebanyak 29 kelas yang masing-masing memiliki 3 jurusan yaitu IPA, IPS dan Bahasa. Masing masing jurusan memiliki jumlah kelas yang cukup banyak. Sehingga pembuatan jadwal mata pelajaran oleh pihak sekolah memerlukan waktu yang cukup lama dan mengganggu kelancaran proses belajar dan mengajar. aplikasi penjadwalan mata pelajaran menggunakan algoritma genetika merupakan solusi yang tepat untuk SMAN31. Algoritma Genetika adalah Proses seleksi alamiah yang melibatkan perubahan gen yang terjadi pada individu melalui proses perkembang-biakan yang menjadi proses dasar dan menjadi perhatian utama. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah Metode Waterfall. Hasil dari penelitian ini adalah aplikasi penjadwalan SMAN 31 Jakarta yang dapat menyusun jadwal yang baik secara otomatis. Kesimpulan dari penelitian ini adalah algoritma genetika dapat diimplementasikan pada aplikasi penjadwalan dibuktikan dengan menguji 30 contoh jadwal yang tersusun oleh aplikasi tanpa ada yang bertabrakan satu sama lain.

Dari ketiga penelitian diatas dapat disimpulkan bahwa penyelesaian masalah penjadwalan yang masih manual dan begitu rumit dapat diselesaikan dengan Algoritma genetika. Masing-masing penerapan dari aplikasi ini menggunakan metode yang sama yaitu metode Algoritma Genetika. Penulis ingin membuat suatu sistem yang dapat mengatur penyusunan jadwal yang baik sesuai dengan sumber daya dan batasan

yang ada dengan menggunakan Algoritma Genetika. Metode ini digunakan pencarian *heuristic* yang didasarkan atas mekanisme evolusi biologis. Setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika. Algoritma ini adalah simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom. Metode yang dapat disebut algoritma genetika adalah metode yang setidaknya memiliki komponen seperti populasi dari kromosom, seleksi berdasarkan *fitness*, *crossover* untuk memproduksi *offspring* baru, dan random mutasi pada *offspring*. Dari cara tersebut yang nantinya dapat menghasilkan penyusunan jadwal yang cukup baik dan akurat yang bermanfaat buat Kaprodi.

B. Penjelasan Teoritis Masing – Masing Variabel Penelitian

1. Penjadwalan

Penjadwalan adalah aktivitas perencanaan untuk menentukan kapan dan di mana setiap operasi sebagai bagian dari pekerjaan secara keseluruhan harus dilakukan pada sumber daya yang terbatas. Menurut (Kenneth R. Baker, 1974) penjadwalan adalah proses untuk melakukan tugas dengan menggunakan sumber-sumber yang tersedia pada waktu yang telah ditetapkan.

Pada dasarnya penjadwalan mencakup pengurutan aktivitas, pengalokasian aktivitas pada fasilitas dan pemetaan aktivitas menurut urutan waktu. Menurut (Farida, 2008), Penjadwalan merupakan kumpulan kebijaksanaan dan mekanisme dalam system operasi yang berhubungan dengan urutan kerja yang dilakukan system computer.

Terdapat batasan atau persyaratan (*constraints*) dalam penyusunan penjadwalan ujian akhir semester. *Constraints* sendiri merukan suatu syarat tidak boleh terjadi pelanggaran terhadap kendala yang ditetapkan agar dapat menghasilkan susunan penjadwalan yang baik. Beberapa constraint tersebut , yaitu:

Pengawas tidak boleh dijadwalkan lebih dari satu kali pada waktu yang bersamaan.

Satu kelas dan ruang tidak boleh dijadwalkan lebih dari satu kali pada waktu yang bersamaan.

Jika terjadi pelanggaran terhadap kendala yang ditetapkan maka akan diberikan suatu nilai pinalti atau hukuman antara 0 sampai 1 untuk setiap pelanggaran. Semakin kecil jumlah pelanggaran yang terjadi solusi penjadwalan yang dihasilkan akan semakin baik.

2. Algoritma Genetika

Algoritma Genetika pertama kali dikembangkan tahun 1970-an oleh Jhon Holland. Tujuan yang ingin dicapai Holland saat itu adalah mengabstraksikan poses-proses evolusi yang terjadi di alam dan mendesain suatu software yang prinsip kerjanya meniru proses-proses evolusi. Hasilnya, algoritma genetika ternyata dapat menyelesaikan masalah-masalah yang tidak dapat diselesaikan dengan menggunakan perhitungan matematika biasa (Arkeman Yandra, Kudang Boro Seminar, 2012).

Algoritma genetika adalah suatu teknik pencarian dan teknik optimasi yang cara kerjanya meniru proses evolusi dan perubahan struktur genetic pada makhluk hidup (Arkeman Yandra, Kudang Boro Seminar, 2012). Algoritma ini sering digunakan untuk mencari solusi optimal baik kasus yang sederhana sampai kasus yang rumit. Algoritma genetik bekerja pada suatu populasi yang dibentuk oleh sebuah kasus yang direpresentasikan sebagai kromosom dan akan dievaluasi untuk memperoleh seberapa nilai optimalnya. Apabila pada generasi tersebut belum ditemukan kasus yang memiliki nilai paling optimal, maka akan dibentuk populasi baru dengan cara melakukan seleksi kepada individu yang nilai optimalnya kecil dan digantikan dengan individu baru melalui proses reproduksi .

Hal-hal yang dilakukan dalam menggunakan algoritma genetika adalah (Sitanggang, 2015):

Mendefinisikan individu, dimana individu menyatakan salah satu solusi (penyelesaian) yang mungkin dari permasalahan yang diangkat.

Mendefinisikan nilai *fitness*, yang merupakan ukuran baik-tidaknya sebuah individu atau baik tidaknya solusi yang didapatkan.

Menentukan proses pembangkitan populasi awal. Hal ini biasanya dilakukan dengan menggunakan pembangkitan acak seperti *random-walk*.

Menentukan proses seleksi yang akan digunakan.

Menentukan proses perkawinan silang (*cross-over*).

Mutasi gen yang akan digunakan.

3. Struktur Umum Algoritma Genetika

Algoritma genetika memberikan suatu pilihan bagi penentuan nilai parameter dengan meniru cara reproduksi genetik, pembentukan kromosom baru serta seleksi alami seperti terjadi pada makhluk hidup. Inisialisasi populasi awal dilakukan untuk menghasilkan solusi awal dari suatu permasalahan algoritma genetika. Inisialisasi ini dilakukan secara acak sebanyak jumlah kromosom yang diinginkan. Selanjutnya dihitung nilai *fitness* dan seterusnya dilakukan seleksi dengan menggunakan metode roda *roulette*, tournament atau ranking.

Kemudian dilakukan perkawinan silang (*crossover*) dan mutasi. Setelah melalui beberapa generasi maka algoritma ini akan berhenti sebanyak generasi yang diinginkan. Sebagaimana halnya proses evolusi di alam, suatu algoritma genetika yang sederhana umumnya terdiri dari tiga operator yaitu: operator reproduksi, operator *crossover* (persilangan) dan operator mutasi. Ada dua hal penting yang harus dilakukan pada awal proses Algoritma Genetika. Pertama, pendefinisian atau pengkodean kromosom yang merupakan solusi yang masih berbentuk simbol. Kedua, penentuan fungsi *fitness* atau fungsi obyektif. Dua hal ini berperan penting dalam algoritma genetika untuk menyelesaikan suatu masalah (Maharani, 2013).

4. Komponen – Komponen Algoritma Genetika

Ada beberapa komponen dalam algoritma genetika, yaitu :

a. Pengkodean

Teknik pengkodean adalah bagaimana mengkodekan gen dari kromosom, gen merupakan bagian dari kromosom. Satu gen akan mewakili satu variabel. Agar dapat diproses melalui algoritma genetika, maka alternatif solusi tersebut harus dikodekan terlebih dahulu kedalam bentuk kromosom. Masing-masing kromosom berisi sejumlah gen yang mengodekan informasi yang disimpan didalam individu atau kromosom.

Berdasarkan jenis simbol yang digunakan sebagai nilai suatu gen, metode pengkodean dapat diklasifikasikan sebagai berikut (Maharani, 2013):

- 1) Pengkodean biner merupakan cara pengkodean yang paling umum digunakan, karena pengkodean ini merupakan yang pertama kali digunakan dalam algoritma genetika oleh Holland (Cordon et al, 2001). Pengkodean biner dinyatakan dalam kromosom biner.
- 2) Pengkodean bilangan bulat. Pengkodean ini baik digunakan untuk masalah optimasi kombinatorial. Dengan pengkodean bilangan bulat, ukuran kromosom menjadi lebih sederhana dan tidak terlalu panjang.
- 3) Pengkodean bilangan riil. (Gen dan Cheng 2000) menyatakan bahwa pengkodean bilangan riil baik digunakan untuk masalah optimasi fungsi dan optimasi kendala.
- 4) Pengkodean struktur data adalah model pengkodean yang menggunakan struktur data. Pengkodean ini digunakan untuk masalah yang lebih kompleks seperti perencanaan trajektori robot dan masalah pewarnaan grap.

b. Inisialisasi Populasi Awal

Inisialisasi populasi awal merupakan suatu metode untuk menghasilkan kromosom-kromosom awal. Jumlah individu pada populasi awal merupakan masukan dari pengguna. Setelah jumlah individu pada populasi awal ditentukan, dilakukan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inisialisasi dilakukan secara acak, namun tetap memperhatikan domain solusi dan kendala permasalahan yang ada.

c. Fungsi Evaluasi (Fungsi *Fitness*)

Fungsi evaluasi dalam algoritma genetika merupakan sebuah fungsi yang memberikan penilaian kepada kromosom (*fitness value*) untuk dijadikan suatu acuan dalam mencapai nilai optimal pada algoritma genetika. Nilai *fitness* ini kemudian menjadi nilai bobot suatu kromosom. Ada dua hal yang harus dilakukan dalam melakukan evaluasi kromosom, yaitu: evaluasi fungsi objektif (fungsi tujuan) dan konversi fungsi objektif ke dalam fungsi *fitness*. Secara umum fungsi *fitness* ditentukan dari fungsi objektif dengan nilai yang tidak negatif, jika ternyata nilai dari fungsi objektif bernilai negatif maka perlu ditambahkan suatu konstanta x agar nilai *fitness* yang terbentuk tidak bernilai negatif (Putra, 2009).

Didalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai *fitness* rendah akan mati. Pada masalah optimasi, jika solusi yang akan dicari adalah memaksimalkan fungsi h (dikenal sebagai masalah maksimasi) sehingga nilai *fitness* yang digunakan adalah nilai dari fungsi h tersebut, yakni $f = h$ (dimana f adalah nilai *fitness*). Tetapi jika masalahnya adalah meminimalkan fungsi h (masalah minimasi), maka fungsi h tidak bisa digunakan secara langsung. Hal ini disebabkan adanya aturan bahwa individu yang memiliki nilai *fitness* tinggi lebih mampu bertahan hidup pada generasi berikutnya. Oleh karena itu nilai *fitness* yang bisa digunakan adalah $f = 1/h$, yang artinya semakin kecil nilai h , semakin besar nilai f . Tetapi hal ini akan menjadi masalah jika

h bisa bernilai 0, yang mengakibatkan f bisa bernilai tak hingga. Untuk mengatasinya, h perlu ditambah sebuah bilangan yang dianggap kecil [0-1] sehingga nilai *fitness*nya menjadi :

$$f = \frac{1}{h} + a$$

dengan a adalah bilangan yang kecil dan bervariasi [0-1] sesuai dengan masalah yang akan diselesaikan (Suyanto,2005). Oleh karena itu fungsi *fitness* menjadi masalah atau penentu utama keberhasilan algoritma genetika. Didalam penelitian ini batasan atau *constraint* dalam penyusunan jadwal proyek yang dijadikan fungsi objektifnya yaitu meminimumkan pelanggaran terhadap *constraint* yang telah ditentukan.

d. Seleksi

Seleksi merupakan proses pemilihan orang tua untuk reproduksi (biasanya didasarkan pada nilai *fitness*). Seleksi bertujuan untuk memberikan kesempatan reproduksi yang paling besar bagi anggota populasi yang paling baik. Ada beberapa metode yang bisa digunakan dalam tahap seleksi, diantaranya :

1) *Range Base Fitness*

Populasi diurutkan berdasarkan nilai objektifnya. Nilai *fitness* dari tiap-tiap individu hanya tergantung pada posisi individu tersebut dalam urutan dan tidak dipengaruhi oleh nilai objektifnya.

2) *Roulette Wheel Selection* (Seleksi Roda Roulette)

Individu-individu dipetakan dalam suatu segmen garis secara berurutan sedemikian hingga tiap-tiap segmen individu memiliki urutan yang sama dengan ukuran *fitness*. Sebuah bilangan acak dibangkitkan dan individu yang memiliki segmen dalam kawasan bilangan random tersebut akan terseleksi. Cara penyelesaian metode ini meniru permainan roda roulette, dimana

prosedur seleksi dimulai dengan memutar roda roulette sebanyak n , setiap waktu dipilih satu kromosom sebagai induk untuk menghasilkan kromosom baru. Metode roulette-wheel selection sangat mudah diimplementasikan dalam pemrograman. Pertama, dibuat interval nilai kumulatif (dalam interval $[0,1]$) dari nilai *fitness* masing-masing kromosom dibagi total nilai *fitness* dari semua kromosom. Sebuah kromosom akan terpilih jika bilangan random yang dibangkitkan berada dalam interval akumulatifnya. Metode ini merupakan salah satu yang paling umum digunakan (Goldsberg, 1989 dikutip dari Riza Arifudin,2011).

3) *Stochastic Universal Sampling*

Individu-individu dipetakan dalam suatu segmen garis secara berurutan sedemikian hingga hingga tiap-tiap segmen individu memiliki urutan yang sama dengan ukuran *fitness*. Kemudian diberikan sejumlah pointer sebanyak individu yang ingin diseleksi pada garis tersebut.

4) *Local Selection* (Seleksi Lokal)

Tiap individu yang berada pada konstrain tertentu disebut nama lingkungan lokal. Interaksi antar individu hanya dilakukan di dalam wilayah tersebut. Lingkaran tersebut ditetapkan sebagai struktur dimana populasi tersebut terdistribusi.

5) *Truncation Selection* (Seleksi dengan pemotongan)

Individu diurutkan berdasarkan *fitness*, hanya individu yang terbaik saja yang akan diseleksi sebagai induk. Parameter yang digunakan dalam metode ini adalah suatu nilai ambang (trunk) yang mengidentifikasi ukuran populasi yang akan diseleksi sebagai induk yang berkisar antara 50%-100%.

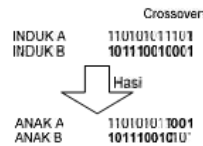
6) *Tournament Selection* (Seleksi dengan turnamen)

Dalam bentuk paling sederhana, metode ini mengambil dua kromosom secara random dan kemudian menyeleksi salah satu yang bernilai *fitness* paling tinggi untuk menjadi orang tua pertama. Cara yang sama dilakukan lagi untuk mendapatkan orang tua yang kedua. Metode tournament selection yang lebih rumit adalah dengan mengambil m kromosom secara random. Kemudian kromosom bernilai *fitness* tertinggi dipilih sebagai orang tua pertama jika bilangan random yang dibangkitkan kurang dari suatu nilai batas yang ditentukan p dalam interval $[0,1]$. Pemilihan orang tua akan dilakukan secara random dari $m - 1$ kromosom yang ada jika bilangan random yang dibangkitkan lebih dari atau sama dengan p . Pada tournament selection, variabel m adalah tournament size dan p adalah tournament probability. Biasanya m diset sebagai suatu nilai yang sangat kecil, misal 4 atau 5. Sedangkan p biasanya diset sekitar 0,75. (Suyanto, 2005)

e. Pindah Silang (*Crossover*)

Pindah silang atau *crossover* adalah sebuah proses yang membentuk kromosom baru dari dua kromosom induk dengan menggabungkan bagian informasi dari masing-masing kromosom. *Crossover* menghasilkan kromosom baru yang disebut kromosom anak (*offspring*). *Crossover* bertujuan untuk menambah keanekaragaman string dalam satu populasi dengan penyilangan antar string yang diperoleh dari reproduksi sebelumnya (Arifudin, 2011).

Crossover merupakan operator genetik utama, yang beroperasi pada dua kromosom dalam suatu waktu dan menghasilkan *offspring* dengan mengkombinasikan kedua fitur-fitur kromosom (Fadlisyah dkk, 2009). Sebuah kromosom yang mengarah pada solusi yang bagus bisa diperoleh dari proses memindah-silangkan dua buah kromosom. Contoh *crossover* dapat dilihat pada gambar berikut:



Gambar 2. 1 Proses Pindah Silang

Proses *crossover* dilakukan dengan cara memilih dua induk dengan kualitas yang baik, setelah itu dilakukan proses ekstraksi kromosom setiap induk. Titik potong ditentukan secara acak, kemudian dilakukan pertukaran bit-bit kromosom disebelah kanan titik kromosom sehingga terbentuk keturunan yaitu anak A dan B. Kromosom anak sebagian besar masih mewarisi kromosom induk tetapi sebagian lagi sudah terjadi pertukaran materi genetik antar kromosom.

Pindah silang bisa juga berakibat buruk jika ukuran populasinya sangat kecil. Dalam suatu populasi yang sangat kecil, suatu kromosom dengan gen-gen yang mengarah ke solusi akan sangat cepat menyebar ke kromosom-kromosom lainnya. Untuk mengatasi masalah ini digunakan suatu aturan bahwa pindah silang hanya bisa dilakukan dengan suatu probabilitas tertentu (probabilitas *crossover*). Artinya pindah silang bisa dilakukan hanya jika suatu bilangan random $[0,1]$ yang dibangkitkan kurang dari probabilitas *crossover* (P_c) yang ditentukan. Pada umumnya P_c diset mendekati 1, misalnya 0.8 (Suyanto,2005). Probabilitas *crossover* (P_c) bertujuan untuk mengendalikan operator *crossover*. Jika n adalah banyaknya string pada populasi, maka sebanyak $(P_c) \times n$ string akan mengalami *crossover*. Semakin besar nilai (P_c), semakin cepat pula string baru muncul dalam populasi. Dan juga jika (P_c) terlalu besar, string yang merupakan kandidat solusi terbaik mungkin dapat hilang lebih cepat pada generasi berikutnya.

Proses *crossover* memiliki nilai kemungkinan yang besar dalam satu siklus algoritma genetika karena tujuan utamanya adalah membentuk keragaman individu, semakin tinggi probabilitas

crossover maka semakin cepat keragaman terbentuk. *Crossover* dapat dilakukan dengan beberapa cara yang berbeda, diantaranya:

1) Penyilangan Satu Titik (*One-point Crossover*)

Metode *crossover* ini yang sering digunakan pada algoritma genetika. Pada penyilangan satu titik, posisi penyilangan k ($k=1,2,3,\dots,n$) dengan n = panjang kromosom yang diseleksi secara acak. Pada titik tersebut dilakukan pertukaran antar kromosom induk untuk menghasilkan anak.

2) Penyilangan Banyak Titik (*Multi-point Crossover*)

Untuk kromosom yang sangat panjang, misalkan 1000 gen, mungkin saja diperlukan beberapa titik potong. Penyilangan dilakukan sebanyak m ($m=1, 2, 3,\dots,n$) dengan posisi penyilangan k ($k=1,2,3,\dots,n$) yang ditentukan secara acak. Pada titik tersebut dilakukan pertukaran antar kromosom induk untuk menghasilkan anak.

3) Penyilangan Seragam (*Uniform Crossover*)

Dibentuk suatu kromosom sepanjang kromosom induk dengan bit-bit yang dipilih secara acak, kemudian penyilangan dilakukan sebanyak m ($m=1, 2, 3,\dots,n$) dengan posisi penyilangan k ($k=1,2,3,\dots,n$) yang ditentukan secara acak. Pada titik tersebut dilakukan pertukaran antar kromosom induk untuk menghasilkan anak.

f. Mutasi

Mutasi merupakan proses mengubah secara acak nilai dari satu atau beberapa gen dalam suatu kromosom (Haupt, 2004 dikutip dari Riza Arifudin, 2011). Mutasi adalah operator algoritma genetika

yang bertujuan untuk membentuk individu-individu yang baik atau memiliki kualitas diatas rata-rata.

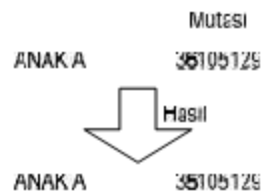
Selain itu mutasi dipergunakan untuk mengembalikan kerusakan materi genetic akibat proses *crossover*. Di dalam algoritma genetika, mutasi melakukan peran vital berikut (Fadlisyah dkk, 2009) :

- 1) Menempatkan kembali gen-gen yang hilang dari populasi sepanjang proses seleksi, agar mereka dapat dilibatkan kembali pada konteks yang berikutnya.
- 2) Menyediakan gen-gen yang tidak hadir pada populasi awal.

Pada mutasi terdapat satu parameter yang sangat penting, yaitu probabilitas mutasi (P_m) yang bertujuan untuk mengendalikan operator mutasi. Probabilitas mutasi didefinisikan sebagai persentasi dari jumlah total gen dalam populasi yang akan mengalami mutasi. Disetiap generasi diperkirakan terjadi mutasi sebanyak $(P_m) \times n$ sring. Pada seleksi alam murni, mutasi jarang sekali muncul sehingga probabilitas mutasi yang digunakan umumnya kecil, lebih kecil dari probabilitas *crossover*. P_m biasanya diset antara $[0-1]$, misalnya 0.1 (Suyanto,2005). Misalkan offspring yang terbentuk adalah 100 dengan jumlah gen setiap kromosom adalah 4 dan peluang mutasi adalah 0.10, maka diharapkan terdapat 40 kromosom dari 400 gen yang ada pada populasi tersebut akan mengalami mutasi.

Mutasi yang digunakan dalam penelitian ini adalah mutasi dengan pengkodean bulat. Mutasi gen ini dilakukan dengan cara pemilihan nilai secara acak. Suatu gen yang terpilih untuk dimutasi nilainya diganti dengan nilai baru yang dibangkitkan secara acak dalam interval nilai-nilai gen yang diizinkan. Misalnya, jika nilai-nilai gen berada dalam interval $[0.9]$, maka gen baru yang dibangkitkan secara acak juga berada dalam interval $[0.9]$. Nilai

gen baru yang dihasilkan bisa saja dibatasi dengan aturan berbeda dengan nilai lama.



Gambar 2. 2 Proses Mutasi

g. *Elitism*

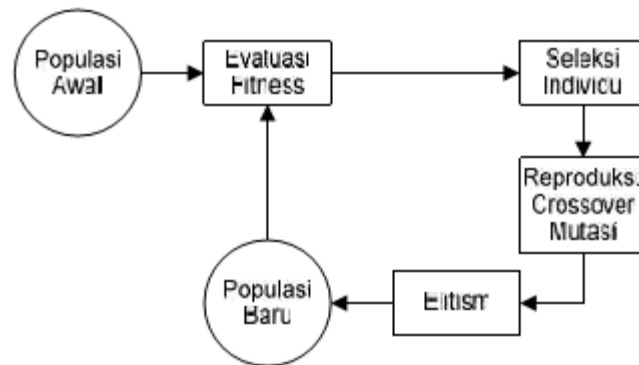
Karena seleksi dilakukan secara acak (random), maka tidak ada jaminan bahwa suatu individu bernilai *fitness* tertinggi akan selalu terpilih. Walaupun individu yang memiliki nilai *fitness* tertinggi terpilih, mungkin saja individu tersebut akan rusak (nilai *fitness* menurun) karena proses *crossover*. Untuk menjaga agar individu yang bernilai *fitness* tertinggi tidak hilang selama evolusi, maka perlu dibuat satu atau beberapa buah duplikatnya. Proses ini dikenal sebagai elitism.

5. Siklus Umum Algoritma Genetika

Langkah Umum pada algoritma genetika adalah sebagai berikut (Putra, 2009):

- a. Melakukan inisialisasi populasi kromosom dengan solusi secara acak (random).
- b. Melakukan evaluasi setiap kromosom dalam populasi menggunakan persamaan fungsi evaluasi (*fitness function*).
- c. Memilih sebagian anggota populasi sebagai solusi yang sesuai dengan induknya untuk generasi selanjutnya.
- d. Menciptakan solusi (keturunan) baru dengan mengawinkan solusi dari induknya dengan cara *crossover* dan mutasi.

- e. Membuang atau menghapus anggota populasi lama yang tidak produktif untuk membuat ruang solusi yang baru agar dapat masuk kedalam populasi.
- f. Jika aturan pemberhentian terpenuhi, berhenti dan keluarkan kromosom yang paling baik. Jika tidak kembali ke langkah 3.



Gambar 2. 3 Siklus Umum Algoritma Genetika.

6. PHP

Secara khusus PHP dirancang untuk membentuk aplikasi web dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, bisa menampilkan database ke halaman web. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun perlu diketahui bahwa PHP sebenarnya bisa dipakai secara command line. Artinya skrip PHP dapat dijalankan tanpa melibatkan web server maupun browser. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi tool yang disebut “Personal Home Page”. Paket inilah yang menjadi cikal-bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2. Pada versi inilah pemrogram dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga bisa

berkomunikasi dengan database dan melakukan perhitungan-perhitungan yang kompleks sambil jalan.

7. MySQL

MySQL adalah salah satu jenis database server yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. Selain itu, ia bersifat Open Source pada berbagai platform (kecuali untuk jenis Enterprise, yang bersifat komersil). MySQL termasuk jenis RDBMS (*Relational Database Management System*). Itulah sebabnya, istilah seperti tabel, baris, dan kolom digunakan pada MySQL. Pada MySQL, sebuah database mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

C. LANDASAN TEORI

Berdasarkan ketiga penelitian yang relevan diatas dan penjabaran variable-variabel yang berkaitan dengan penelitian ini, dapat diambil kesimpulan bahwa semua penelitian relevan terkait membahas tentang sistem penjadwalan secara otomatis yang dilakukan penulis, pada penelitian ini, penulis membuat sistem penjadwalan ujian semester otomatis yang bisa memunculkan jadwal ujian dan hasil akhir penjadwalan bisa dilihat lewat halaman web. Penulis membuat sistem ini bertujuan agar pihak Fakultas Ilmu Kesehatan Universitas Muhammadiyah Magelang dapat membuat jadwal secara otomatis dengan menggunakan metode yaitu Algoritma Genetika.

Untuk penentuan Penjadwalan yang secara otomatis ini, penulis memilih metode Algoritma Genetika dalam menentukan jadwal ujian semester. Dengan adanya sistem ini yang diharapkan akan mempermudah fakultas, Kaprodi, Universitas Muhammadiyah Magelang khususnya fakultas Ilmu Kesehatan.

BAB VI

KESIMPULAN DAN SARAN

Setelah melalui proses analisis perencanaan, implementasi, serta pengujian maka pada bab ini akan dibahas kesimpulan tentang hasil. Selain kesimpulan dari permasalahan yang diangkat juga akan disampaikan saran-saran yang dapat memberikan masukan dan catatan-catatan guna pengembangan sistem menjadi yang lebih baik.

A. KESIMPULAN

Dari hasil penelitian yang dilakukan, dapat diambil kesimpulan sebagai berikut :

1. Telah berhasil dirancang solusi penjadwalan ujian otomatis dua Prodi secara bersamaan dan menjamin seorang pengawas tidak mengawasi ujian dalam waktu yang bersamaan dengan menggunakan Algoritma Genetika
2. Algoritma genetika mampu menyelesaikan permasalahan penjadwalan ujian dua prodi secara bersamaan di mana algoritma genetika memperlihatkan hasil fitness akhir dengan nilai 1 yang mana menjadi solusi terbaik yang berarti tidak terjadi bentrokan jadwal pengawas.

B. SARAN

Sistem penjadwalan ini mungkin masih dapat dikembangkan lagi dengan menambahkan beberapa fungsi atau menu tambahan dan juga tampilan atau desain antarmuka dapat dibuat lebih menarik lagi, Penambahan tenaga pengawas juga sangat diperlukan agar keakuratan sistem dapat lebih meningkat dikarenakan semakin banyaknya jumlah pengawas mempengaruhi dengan tingkat keakuratan sistem dalam membuat ujian akhir semester.

DAFTAR PUSTAKA

- Aditya, L.A., & Mega W., 2017. Implementasi Algoritma Genetika untuk Penjadwalan Mata Pelajaran pada LMS GETSMART. Matik Penusa : Jurnal Manajemen dan Informasi Pelita Nusantara, Volume 21, Nomor 1, Juni 2017, halaman 65-70.
- Arifudin, Riza. 2011. Optimasi Penjadwalan Proyek dengan Penyeimbangan Biaya Menggunakan Kombinasi CPM dan Algoritma Genetika. Jurnal Masyarakat Informatika, Volume 2, Nomor 4, halaman 1-14.
- Arkeman, Kundang dan Hendra. 2012. *Algoritma Genetika Teori dan Aplikasinya untuk Bisnis dan Industri*. IPB Press, Bandung.
- Baker, Kenneth R.(1974), *Introduction To Sequencing and Scheduling*, Jhon Willey and Sons, Inc. New York.
- Fadlisyah, Arnawan, Faisal. 2009. *Algoritma Genetika*. Graha Ilmu, Yogyakarta.
- Farida, N. I. 2008. *Sistem Pendukung Keputusan Penjadwalan Pengajar Praktikum Laboratorium Komputer STIKOM Menggunakan Algoritma Genetika*. STIKOM Surabaya, Surabaya.
- Gata Windu, Gata Grace, 2013. *Sukses Membangun Aplikasi Penjualan dengan Java*. PT. Elex Media Komputindo Kelompok Gramedia, Anggota IKAPI, Jakarta 2013.
- Gen, M. & R. Cheng. 2000. *Genetic Algorithm and Engineering Optimization*. Jhon Wiley and Sons. Inc. Newyork.

- Goldberg, D. E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. United State of America : Addison-Westley.
- Haupt, R. L. & Haupt. 2004. *Practical Genetic Algorithms*. New Jersey: Jhon Wiley dan Sons, Inc.
- Ivan, Stephanus Raphael dan Halim Agung. 2018. Aplikasi Penjadwalan Mata Pelajaran di SMAN 31 Menggunakan Algoritma Genetika Berbasis WEB. *Jurnal SIMETRIS*, Volume 9, Nomor 1, April 2018, Halaman 641-656.
- Maghfira, Tusty Nadia dkk. Optimasi Penjadwalan Pengajar Menggunakan Algoritma Genetika Pada Royal English: Toefl & Toeic Center Malang. Skripsi Sarjana, Fakultas Ilmu komputer Universitas Brawijaya Malang, Malang, 2016.
- Maharani Febria. Sistem Penjadwalan Proyek Menggunakan Algoritma Genetika. Skripsi Sarjana, Fakultas Sains dan Teknologi Universitas Islam Negeri SUSKA, Riau, 2013
- Pradnyana, Nanda Bagus, Dwi Sunaryono, dan Abdul Munif. 2012. Perancangan dan Pembuatan Aplikasi Penjadwalan Perkuliahan Menggunakan Algoritma Genetik dan Teknologi Java API for XML Web Service pada Platform Android. *Jurnal Teknik Pomits* Volume 1, Nomor 1, (2012) 1-5.
- Putra, Y. Aplikasi Penjadwalan Perkuliahan Menggunakan Algoritma Genetika. Skripsi Sarjana, Fakultas Sains dan Teknologi Universitas Islam Negeri SUSKA, Riau, 2009
- Rahadi, D. R. (2014). Pengukuran Usability Sistem Menggunakan Use Questionnaire Pada Aplikasi Android Interface pengguna Android didasarkan pada manipulasi langsung menggunakan masukan sentuh yang serupa dengan tindakan di dunia nyata , seperti menggesek (swiping), mengetuk . 6(1), 661–671.

Sitanggang, D. 2015. Inisialisasi Populasi Pada Algoritma Genetika Menggunakan Simple Hill Climbing (SHC) Untuk Traveling Salesman Problem (TSP). *Jurnal Times*, Volume 4, Nomer 2, Halaman 40-44.

Suyanto. 2005. *Algoritma Genetika Dalam MATLAB*. Andi. Yogyakarta

Toscany, A., N., & Rusdianto, R. 2017. Pengembangan Sistem Penjadwalan Kuliah Menggunakan Algoritma Genetika. *Jurnal Manajemen Sistem Informasi*, Volume 2, Nomor 2, Juni 2017, Halaman 379-393.