DECISION SUPPORT SYSTEM BASED ON GENETIC ALGORITHM FOR COURSE SCHEDULING PROBLEMS

by

Vania Berliana

A Thesis

Submitted to the Faculty of Nanjing Xiaozhuang University

In Partial Fulfillment of the Requirements for the degree of

Bachelor of Software Engineering

School of Information Engineering

Fangshan, Nanjing

May 2019

Approval Page

To: Dean Xiangjun Zhao
School of Information Engineering

This thesis, written by Vania Berliana, and entitled Decision Support System Based on Genetic Algorithm for Course Scheduling Problems, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Qing Li

Haiyong Wu

Wan Li Song

Date of Defense: June 2, 2020
The thesis of Vania Berliana is approved.

Dean Xiangjun Zhao
School of Information Engineering

Nanjing Xiaozhuang University, 2020

Dedication

To my beloved family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Scheduling lecturers is a complex problem that offer occurs on campus. Scheduling is needed to manage work time, so that the schedule is as efficient as possible. A scheduling will look easy if the number of scheduled components is relatively small, but solving the problem of scheduling lectures in very large numbers until now is still a complicated problem to be solved manually.

Several processes are needed so that the schedule will be formed optimally according to the rules given. The rule in making this course scheduling is that courses for the same class are not allowed to be held simultaneously, one space is only for one course, for practicum subjects are in the laboratory, the possibility that the lecturer will teach more than one subject, there is a possibility that the number of courses and the number of lecturers is not comparable, so solutions must be considered so that lecturers do not teach two different courses on the day and the same time. There is also a possibility of excessive teacher teaching hours. In addition, the availability of classrooms must also be considered so that lecturing activities can be carried out. For this reason, a method is needed to solve the complexity of scheduling lectures so that learning activities can be carried out optimally.

Genetic algorithms can be used as an alternative solution to solve the subject scheduling problem. This information system with genetic algorithms can process lecturer data, courses, rooms, and lecture time slots into optimal lecture schedules, which there are no more clashes in courses and the availability of efficient use of space for all existing courses.

Keywords: Genetic Algorithms, Course Scheduling, Method, Lecture, Optimally, Complexity

CHAPTER 1. INTRODUCTION

1.1 Research Background

The problem of scheduling for teaching received the attention of many researchers. Scheduling is needed to manage work time, so that the schedule is as efficient as possible. One of the scheduling cases at a university is course scheduling. A number of methods have been produced to get the optimum schedule. A course scheduling will look easy if the number of scheduled components is relatively small, but it will be complicated if the constituent components are in large numbers. The problem of scheduling this course will be very complex when involving dozens or even hundreds of lecturers , rooms and courses offered because more and more combinations of components may occur, where in the selection of combinations must be considered the rules set in making the schedule.

The course scheduling creation process at several tertiary institutions is still done manually. The schedule produced in this way requires a long process of time to find a solution, especially if the size of the problem is getting bigger with the increasing number of components and conditions determined by the institution where the schedule is used. Besides requiring very high accuracy and relatively little estimated time, this method also allows for errors. For example, considerations made for compiling a schedule need to pay attention to various components namely, Teachers, Students, Rooms and Subjects. In order to arrange a good schedule, correlations must be made between these components to avoid cases of collision schedules. Not only schedule collisions are taken into consideration. But also some other parameters, such as repeated repetition of the same schedule in one day, limited number of teacher teaching hours, number of hours for students that are adjusted to grade level. The resulting schedule must also meet the restrictions and conditions which aim to make the resulting schedule suitable when used. With so many problems, the man in charge of making a schedule is certainly likely to get into trouble.

Based on this difficulty, it is necessary to develop a right system for scheduling lectures that can accommodate various aspects considered above. Thus making it easy for scheduler in making course schedules. There are several methods and algorithms that are often used in solving scheduling problems. One such method and algorithm is the genetic algorithm which will be applied in this research.

1.2 Research Status Analysis

This research will discuss plans to develop a decision support system based on genetic algorithm for course scheduling problems. Genetic algorithm was chosen because genetic algorithm is different from other algorithms. The uniqueness of genetic algorithms is to follow the evolutionary patterns of living things and randomly generated numbers. Although numbers are generated randomly, they must be in accordance with established rules.

This genetic algorithm is proposed with one aim to facilitate the process of allocating time for each activity in a project according to certain rules or circumstances, because each problem in the form of adaptation can be formulated in genetic terminology and for projects with a combination of problems that are quite complex, pure mathematical concepts are not able to accommodate the search for solutions to problems, so it must use suitable methods to obtain optimal scheduling solutions (Haldurai, Madhubala, & Rajalkshmi, A Study on Genetic Algorithm and its Applications, 2016).

Genetic algorithm is a technique for finding solutions using the principle of natural selection. Genetic algorithm is used to solving problems that are modeled by the biological processes of evolution. This algorithm is one algorithm that is very appropriate to be used in solving complex optimization problems, which are difficult to do by conventional methods (Suwirmayanti, Sudarsana, & Darmayasa, Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran, 2016).

1.3 Significance of The Study

1.3.1 Formulation of Problems

Based on the background above, researcher identified the problems that occur in the field as follows:

1. How to implement the genetic algorithm to the course scheduling information system.
2. The feasibility of the genetic algorithm approach to the problem of scheduling school subjects.
3. In scheduling school subjects, many obstacles are found in terms of accuracy and optimization. For example the occurrence of a schedule collision due to the existence of compound classes with a limited number of teachers and space.
4. The timetable is limited, meaning that only groups of officers or teachers in the curriculum section are often involved in making schedules.

5. The scheduling model is still in the form of a manual model that does not yet have an optimization system.
6. The randomization of the requests from each lecturer/teaching staff. Such as teachers who can not teach on certain days.

1.3.2 Limitation of Problems

The scope of the research to be discussed are as follows :

1. Making course scheduling information system using a genetic algorithm just for one department / study program.
2. The number of students in one class is smaller or equal to the total capacity of lecture space or laboratory.
3. Solving the problem using the genetic algorithm approach.
4. Building a lecture scheduling system application.

1.3.3 Research Objectives

The purpose of this study are as follows:

1. Build a course scheduling system using genetic algorithms.
2. Generate output in the form of an optimal schedule.
3. Knowing how to implement the creation of course scheduling system using genetic algorithms.
4. Knowing the feasibility of the genetic algorithm approach to the problem of scheduling school subjects.
5. Complete the schedule crash due to many factors by minimizing errors and finding the best possible solution as effectively as possible.
6. Eliminate limited resource requirements for schedulers.
7. Quickly create accurate schedules that have been equipped with system optimization.
8. Accommodate the conditions from teaching staff who have barriers to teach on certain days and hours.

1.3.4 Benefits of Research

Research on the use of genetic algorithms for course scheduling information system is expected to be useful both theoretically and practically:

1. Theoretically

a. For the development of science, especially for the world of software engineering.

b. Discover how to implement genetic algorithms to the course scheduling information system.

2. Practically

a. For Researchers

i. Understanding the work process of genetic algorithms and the forms of their implementation.

ii. As a portfolio for researchers who are useful for the future.

b. For the University

i. As a reference material for future research.

ii. As an evaluation material for universities in developing science, in this case relating to genetic algorithm-based programs.

1.4 Schedule

Scheduling will be implemented in this thesis as follows:

No	Activities	Weeks								
		1	2	3	4	5	6	7	8	9
1	Literature review	■	■							
2	Analysis of needs/requirements			■	■					
3	Design				■					
4	Coding					■	■	■	■	
5	Testing					■	■	■	■	
6	documentation									■

The process of arranging this thesis takes approximately 9 weeks.

1.5 Organizational Structure of The Paper

As for the systematic structure, this thesis is divided into five chapters, with explanations for each chapter as follows:

CHAPTER 1 INTRODUCTION

This chapter, the author discusses the background of research, identification of problems, research objectives, limitations, and benefits of research, research methodology, and schedule.

CHAPTER 2 RELATED TECHNOLOGIES AND THEORIES

This chapter discusses the system that has been made before and the limitation

CHAPTER 3 SYSTEM REQUIREMENTS ANALYSIS

In this chapter, we discuss the basic theories that support and are related to this thesis research.

CHAPTER 4 SYSTEM DESIGN

This chapter discusses the methodology used in this research. Start from research design, data collection techniques, and conceptual framework.

CHAPTER 5 SYSTEM IMPLEMENTATION

This chapter explains the process of making and application results from designs that have been made in the previous chapter.

CHAPTER 6 SYSTEM TEST

This chapter contains tests on the system and how the tests are carried out, with what tools, and the results of the tests.

CHAPTER 7 CONCLUSION AND PROSPECT

This chapter contains conclusions from this thesis and suggestions that are used as references for future development.

CHAPTER 2. RELATED TECHNOLOGIES AND THEORIES

2.1 Technology Overview

2.1.1 Codeigniter

CodeIgniter is a framework created using the PHP language, which can be used for rapid web development. While the framework can be interpreted as the structure of libraries, classes and run-time infrastructure that can be used by programmers to develop web applications quickly. The purpose of using the framework is to make it easier for web developers to develop robust web applications quickly without approval. Design patterns in web development with CodeIgniter using MVC (Models-View-Controller). Where applications are made that will connect between business and presentation, thus allowing web programmers and web designers to work with one another. In order to develop the web with CodeIgniter, it is necessary to first concept the MVC and the directory structure of CodeIgniter. (Destinigrum & Adrian, 2017)

2.2 Application Components

2.2.1 General Scheduling

General scheduling is the division of time based on a work order arrangement plan; a list or table of activities or planned activities with a detailed division of implementation time. While the definition of scheduling is the process, manner, scheduling or entering into a schedule.

Another notion of scheduling is the process of assigning it to a set of power sources. This is an important concept in various areas such as computing and production processes.

Automatic planning and scheduling is a branch of artificial intelligence that refers to the realization of strategies or work sequences, specifically for the implementation of intelligent agents, automated robots, and unmanned vehicles. Unlike classical control and classification problems, the solution is complex, unknown, and must be found and optimized in a multidimensional space.

In mathematics, scheduling problems are often solved as an optimization problem, with the aim of maximizing the quality of scheduling. For example, an airline wants to reduce the number of airport gates needed for its aircraft in order to reduce its operating costs. (Ridwan, 2016).

2.2.2 Course Scheduling

Course scheduling (lecture timetable) is a matter of placing time and space on a number of courses, tutorials, and similar academic activities, taking into account a number of rules relating to the capacity and location of available space, free time required and a number of other rules relating to tolerance for lecturers, and the relationship between elective courses.

The essence of course scheduling is how to schedule a number of components consisting of students, lecturers, space, and time with a number of specific rules and constraints.

The subject matter scheduling is a very complex problem, because it involves dozens of lecturers, rooms and courses offered. The more components there are, the more combinations of components that might occur. And, the most important thing in choosing a combination must be considered the rules that have been set.

At present there are several methods to solve course scheduling problems, namely using several methods, such as intelligent search techniques, graph methods, and genetic algorithm.

On the process of solving course scheduling problem there are obstacles that must be met or must not be violated. These constraints are a measure of the quality of scheduling courses, so that an optimal course schedule can be formed. Constraints that must be met in scheduling courses in general are the obstacles that occur at a particular campus or university. These constraints are:

a) Lecturers can teach more than one course and may not collide with the lecturer.
b) One course can be taught by 2 or more lecturers. There are certain courses that use laboratory rooms that must be scheduled in the laboratory room.
c) Students can take courses before and after the class and may not occur collisions in courses that have been taken.
d) Availability of sufficient space for all courses.

(Ridwan, 2016).

2.2.3 Factors that Affect Course Scheduling

In the manufacturing schedule for course there are some components major, such as:

a) Lecturer

A lecturer is not able to teach several subjects at the the same time. Besides that, a lecturer can be ordered to teach only at the time, at a time that is desired.

b) Space / Room

Availability of rooms is also very much needed in making class schedules. Space capacity must also be adjusted to the number of students who take classes.

c) Time

Time that is provided is also limited, so schedules should be made optimal as possible.

d) Subject

All subjects had a semester of courses that are offered, it is necessary to have scheduling boundaries.
(Jing, 2018).

2.2.4 Course Scheduling Rules

Scheduling made by rules that have been set, there are rules of general course scheduling, such as:

a) The class schedule is a combination of the main components, namely lecturer, students, subjects/courses, space/room/, and time. If one components is not complete, the schedule cannot be used.
b) Different class subjects cannot be located in the same time and space (room).
c) There is no lecturer who teach more than one subject at the same time.
d) There are limitations on the availability of space and time.

In addition to rules of general course scheduling , there are also some rules specifically, such as:

a) Lecturers can teach according to their time availability.

b) Compulsory and elective courses in one semester may not collide with their culinary hours.
c) The room that used must meet the capacity of students who follow courses. (Jing, 2018).

2.2.5 Genetic Algorithms

Genetic algorithms are inspired by genetic science, so the terms used in genetic algorithms are adapted from that science. The genetica algorithm is one algorithm that is very appropriate to be used in solving complex optimization problmes, which are difficult to do by conventional methods (Tabassum & Mathew). The nature of genetic algorithm is to look for possibilities from prospective solutions to get the optimal solution for the problems. (Sugeha, Inkiriwang, & Pratasis, 2019). Genetic algorithms provide a choice for determining parameter values by mimicking the way of genetic reproduction, the formation of new chromosomes and natural selection as occurs in living things (Kumar).

The encoding used in course scheduling is the value encoding. Before the chromosome model is made, simplifying the representation variables forming the chromosome model, which is the main component subject scheduling. It is simply not possible for a teacher to teach the same or different subjects in a given time. Similarly, this can be changed to all classes taught by the same teacher with different itme allocations. The remaining components are space and clock. So the genetic model used in the process of scheduling these subjects consists of space and time. Iteration in the scheduling process with the genetic model above will be very complex so it requires a long time. This is because the genetic algorithm works using a random approach so that the resulting values are also random (Affenzeller, Wagner, Winkler, & Beham, 2009). In this process, we need a value to get the right combination of teacher variables, space, and hours to avoid conflicts. Subjects are arranged sequentially on a chromosome because all subjects must be allocated hours and space. The order is based on the teacher taking it with the aim of making it easier to check the rules relating to the teacher. (Massalesse, 2011).

There are two important things that must be done at the beginning of the genetic algorithm process. First the definition or coding of chromosomes which are solutions that are still in the form of symbols. Second, determining the fitness function or objective function. These two things play an important role in genetic algorithms to solve a problem. (Abdy, Wahyuni, & Ilmi, 2016)

The basic concept that inspired the emergence of genetic algorithms was the theory of natural evolution put forward by Charles Darwin. In the theory it is explained that in the natural evolutionary process, each individual must adapt to the surrounding environment in order to survive. (Priambodo, Nhita, & Aditsania, 20166).

A little look back, actually the forerunner to the development of this Genetic Algorithm originated from the initiatives of several Biologists. The biologists use digital computers to work on simulations of genetic systems. These experts include Baricelli, NA who in 1957 conducted research on symbiogenetic evolution and in 1962 put forward the theory of evolution and its numerical analysis. Besides Baricelli, another name that initiated the Genetic Algorithm was Fraser, who in 1960 simulated a genetic system with a computer. The research carried out by biologists gave the idea for Jhon Holland and his student David Goldberg to apply this genetic process to an artificial system. (Worzel, Yu, Almal, & Chinnaiyan, 2009).

2.2.6 Genetic Model

The coding that will be used in scheduling this course is the value coding. The choice of value coding because if it is done binary coding it will make the sequence bits more complex, so also the longer sequence bits need to be recorded to get the true value of the rules represented. Students are not included in the chromosome coding, because students have the right to freedom to choose the courses offered. Giving lecture teaching assignments to lecturers has been done before making a schedule. Simply put, if there are no lecturers teaching the course then this course is not offered. The remaining scheduling components are the course, space, and time. Time has two components, namely days and hours. So the genetic model of scheduling consists of days, hours, and space.

Genetic algorithm works by using a random approach, so that the resulting values are random values. In the case of scheduling with a genetic model consisting of space, day, and hour, there will be many iterations. That is because an appropriate value is needed in order to get the right combination between variable lecturers, time, and space that do not conflict with each other. The more iterations are performed, the longer it will take. Therefore, the resolution of the subject matter scheduling problem will be resolved through two stages. The first step is to place class courses in the available time slots. In the first stage it is solved using genetic algorithms. This stage is completed first because of the many rules relating to time variables. The second step is to place the course class which has got a time slot in the appropriate room and at this stage is completed with ordinary search programming. (Mahmudy & Mawaddah, 2006).

Another reason for the division of completion into two stages is because the lecturer did not ask for the room to be used for teaching and the number of rooms did not change much. The division of genetic models into two stages is expected to be able to solve the scheduling problem faster because the search for combinations that do not conflict with each other between space and time are not done together but at different stages. In the first stage, courses are arranged sequentially on chromosomes, because each course must be allocated time. Sorting is done in accordance with the order of lecturers who teach, it is done to facilitate checking the time of willingness of lecturers. After all class courses occupy time slots, the next step is in the second stage, namely allocating space in accordance with student capacity. The following are examples of stages to be worked on:

Stage 1 Model

Lecturer 1		Lecturer 2		Lecturer 3	
C a	C b	C c	C d	C e	C f
T1	T3	T2	T1	T4	T3

Table 2.1 Stage 1 of Genetic Algorithm Model Table

Stage 2 Model

T1		T2		T3	T4
C a	C b	C c	C d	C e	C f
R1	R4	R3	R2	R4	R1

Table 2.2 Stage 2 of Genetic Algorithm Model Table

Information:

Lecturer : lecturer who teaches

C : taught subjects / course

T : teaching time (days, time)

example:

T1 (2,3) = Tuesday, session 3

T2 (4,4) = Thursday session 4

T4 (1,5) = Monday session 5

T3 (5,2) = Friday session 2

R : teaching room

Example: R1: C1, R2: C2,

R3: C3, R4: C4

In the stage 1 model can be seen the first stage, each lecturer teaches several courses as examples:

- lecturer 1 teaches subject a and subject b,
- lecturer 2 teaches subject c,
- lecturer 3 teaches subjects d, subjects e, and subjects f.

Then stage 1 is carried out using genetic algorithms, resulting in a time slot for each course (T1, T3, … Tn).

In the stage 2 model is the second stage process. In the second stage the division of the room is carried out using the usual search algorithm. In 1 time slot there are several courses taught. For example, in time slot T1 there are subjects a and b. R1 states the room in accordance with the number of participants required by a course, as well as R4. (Ridwan, 2016).

2.2.7 Course Scheduling Using Genetic Algorithms

At present there are already many scheduling courses using genetic algorithms. Basically the whole process is almost the same. But the difference is in the selection process, cross marriage, and mutation. Also important is the existence of scheduling rules that must be as optimal as possible to be applied. (Kachitvichyanukul, 2012).

The initial process of constructing a course schedule using genetic algorithms is modeling or representing in the form of chromosomes. Commonly used encoding is value coding. The value coding is chosen because it will be easier to use, which is directly representing the problem directly.

Some coding examples in the scheduling case are represented as matrices $(R)_{ij}$ ($1 \leq i \leq m$, and $1 \leq j \leq n$), where each row represents the lecturer and each column represents time; an element of the matrix R is the class course (r_{ij} ε $\{C_1, ..., C_k\}$) [2] .

The coding has been chosen, then the next process is selection. In the selection process will largely determine the individual chosen to be the parent. Individuals who have good fitness values, will have more possibilities to be selected as the parent. Selection which is commonly used is the roulette wheel, ranking method, and tournament method. The selection of the selection method depends on the existing problems, so it is necessary to do some experiments to find the suitable selection method used.

If several parents have been chosen, then the next process is to conduct cross-marriages between the parents. Most cross marriages used are two point cross marriages and recombination cross marriages.

To prevent premature convergence, mutations can be made. Mutations can be done in various ways. In matrix coding, mutations can be done by taking two elements from the matrix R, then exchanging them. (Jing, 2018).

2.2.8 Chromosome Initialization

Chromosome initialization is represented in an array with a data record type that contains data that supports the scheduling process. The length of a chromosome is as many genes as there are, in which each gene represents the course offered. (Janah, Brayudi, & Maria, 2016).

Each chromosome is a sequence of genes consisting of two values, namely the value of days and hours. Day values are expressed in bytes 1 through 5 to represent Monday through Friday and gene value are obtain by randomly generating numbers. While the time value is raised value 1 to 5 which represents 5 available time slots in one day. So that in one week there are 25 slots available time. (Suwirmayanti, Sudarsana, & Darmayasa, Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran, 2016)

Illustration of chromosome initialization can be seen in the following table:

Lecture ID	1	1	2	3	3	4
Course ID	1	2	3	4	5	6
Chromosome 1	1,2	2,4	3,5	4,4	2,1	3,1
Chromosome 2	2,2	1,4	3,3	1,1	5,4	4,2
…..	…	…	…	…	…	…
Chromosome n	1,5	4,5	3,4	2,2	4,5	1,1

Table 2.3 Chromosome Initialization Table

The table above is an illustration of chromosome initialization. After the genetic process occurs, chromosomes are made up of several genes (according to the number of subjects) that contain time slots (days and hours) for each course. (Suwirmayanti, Sudarsana, & Darmayasa, Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran, 2016).

2.2.9 Fitness Function

Every rule that is used in scheduling subjects is given a penalty value, where the more mandatory to be carried out, the greater the penalty value given. The calculations are done by giving a penalty for each rule used in scheduling.

Here are the rules for calculating the fitness function:

$f(g) = 1 / (1 + \sum P_i v_i (g))$;

where P_i is the penalty given for rule i, and $v_i (g) = 1$ if schedule g violates rule i, is 0 if otherwise.

From the calculation of fitness values it can be seen that the fewer rules that are violated, the greater the fitness value. A perfect schedule will have a fitness value of 1, because the total value of a rule that is broken is 0. (Suwirmayanti, Sudarsana, & Darmayasa, Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran, 2016).

2.2.10 Selection

Assuming all chromosomes are placed on a roulette wheel, the magnitude of the possibility for each chromosome is dependent on the fitness value as in the following example:

Chromosome	Fitness
A	15
B	5
C	10
D	5
E	5

Table 2.4 Roulette Wheel Table

Then the probability of a chromosome in the roulette wheel can be described as follows:

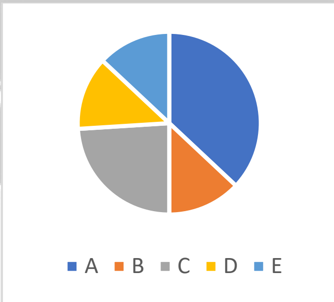


Figure 2.1 Roulette Wheel Figure

In the picture above is an example of a population consisting of five chromosomes. Each chromosome has a different fitness value. From the table above it can be seen the probability of each chromosome being chosen to become the parent. Chromosome A has a fitness value of 15 and the value is the highest fitness value in the population. So chromosome A has the greatest probability of being selected as the parent. (Budhi, Gunadi, & Wibowo, 2011).

2.2.11 Crossover

This crossover process is a process to increase string diversity in a population. Cross shifting operators have the most impotant role in genetic algorithms because there is a process of mating (crossing) genes between two individuals (parents) that produce two new individuals (offspring) in the next generation. The gene exchange must also be checked whether new individuals are formed in accordance with applicable regulations. The resulting child chromosome is a combination of genes owned by the parent chromosome. The parent chromosomes are randomly selected as many as the crossover rate that was set at the beginning. For example there are 10chromosomes with a CR (crossover rate) of 70% then the parent used is 7. Of these7 parents, two will be paired to produce 1 new individual. Parent pairs that occur are 7 pairs namely parent 1 and parent 2, parent 2 and parent 3, and so on until parent 7 with parent 1. Each parent will pair twice. (Hassanat, et al., 2019).

2.2.12 Mutation

Mutations are carried out to prevent premature convergence. Mutations can be done in two ways, namely the random method and the method of swap or exchange. The first way to mutate is to determine the two genes to be mutated. After that the value of the two genes is randomized again to get a new value. In the second way is to exchange values directly from genes. The choice of the mutation method is done randomly. (Suwirmayanti, Sudarsana, & Darmayasa, Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran, 2016).

CHAPTER 3. SYSTEM REQUIREMENTS ANALYSIS

3.1 Enterprise Background

The system that will be developed for scheduling this course uses genetic algorithms, where the scheduling process is done automatically. In the event of a scheduled crash, the developed program will search automatically and continuously for the most optimal schedule.

This application was created with the aim of accommodating the making class schedules per semester while the data can change according to semester and campus policies. The process of planning a lecture scheduling system using genetic algorithm methods consists of chromosome formation, evaluation of fitness values, population formation, completion process, crossover, mutation process, checking rules and acceptance, changing new population, and testing (if the machine is finally fulfilled and found a solution) and the condition is complete.

3.2 Business Relationship Analysis

According to (Sukrianto, 2017), The ERD model contains entity components and sets of relations, each of which is equipped with attributes that present all the facts reviewed so that the relationship between existing entities and their attributes can be identified. It also can describe the relationships that exist in data processing. The following symbols from ERD:

Table 3.1 ERD

No	Image	Information
1.	Entity	An entity or rectangular shape is a data object.
2.	Relation	Relationship between entities that generally given names with basic verbs.
3.	Attribute	An elliptical attribute or form is something that explains what an entity or relationship actually means and represents the attributes of each entity.
4.	――――――	Lines are links between entities

3.3 Functional Requirements Analysis

The project scheduling system requires several functions to be used properly by the user, and provides optimal results. The functions that users need are as follows:

a. Data input function, consisting of project data, activity data, data for the genetic algorithm process, and dictionary data.
b. Process functions, consisting of project scheduling processes.
c. Report printing function, used to display and print report in the form of a project work calendar produced by the system.

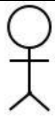
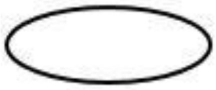
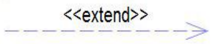
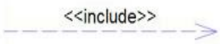
While the functions needed by the system are as follows:

1. Initial population initialization function
   This function is used for inputting and initial population initialization by randomizing all data when the activity is carried out to become chromosomes.
2. The function of chromosome evaluation and selection
   This function is used to assess whether or not a chromosome is good which can be a solution to the problem.
3. Crossover function
   This function is used to perform a crossover between two chromosomes that have been previously evaluated.
4. Mutation function
   This function is used to change the composition of genes contained in chromosomes by changing the value of genes.
5. Setting function
   In this function there is a facility to determine parameters in genetic algorithms, such as crossover probability and mutation probability. However, this parameter also has a default value to anticipate if changes in parameter values result in unsatisfactory performance.

3.3.1 Personal Office Demand Analysis

Use Case describes an interaction between one or more actors with information systems to be created. Use case diagram symbols can be seen in table below.

Table 3.2 Use Case Diagram

No	Image	Name	Information
1.		Actor	people, processes, or other systems that interact with the information system to be created.
2.		Use Case	functionalities provided by the system as units exchanging messages between units or actors.
3.		Association	communication between actors and use cases or use cases that have interactions with actors.
4.		Extend	an additional use case relation to a use case where the added use case can stand alone even without the additional use case.
5.		Include	shows that a whole use case is functionality from other use cases.
6.		Generalization	demonstrates specialization actor to be able to participate with use case.

3.3.2 Document Management Demand Analysis

The research material is in the form of scientific works presented in journals, papers, seminar proceedings, bulletins, and books relating to the problem being studied that can support as a reference in this study. Literature studies needed are about genetic algorithms, databases, UML, etc.

3.3.3 Workflow Requirements Analysis

According to (K, Irfan, & Nurpianti, 2013), Flowchart is used to show the workflow or what is being done in the system as a whole and explain the sequence of procedures that exist in the system. The following are the symbols of the flowchart.

Symbol	Name	Function
	Process	Indicates any type of internal operation inside the Processor or Memory
	input/output	Used for any Input / Output (I/O) operation. Indicates that the computer is to obtain data or output results
	Decision	Used to ask a question that can be answered in a binary format (Yes/No, True/False)
	Connector	Allows the flowchart to be drawn without intersecting lines or without a reverse flow.
	Predefined Process	Used to invoke a subroutine or an Interrupt program.
	Terminal	Indicates the starting or ending of the program, process, or interrupt program
	Flow Lines	Shows direction of flow.

Table 3.3 Flowchart Symbols

3.3.4 Address Book Management Demand Analysis

Data collection is done to get information about what needs to be done when building a system. In making this information system there are several things that must be done to build a system, including the following:

1. Literature (Reading)

   Activities to collect data are carried out by reading journal references or sources related to research in the form of soft-copy and hard-copy so that they get the right reference.

2. Observation

   Data collection is done by making direct observations that occur in the community and what is needed to be able to provide important data that may affect the system.

3. System Development Study

   Aims to determine the methodology of software development used with a structured approach.

3.3.5 Problem Base Demand Analysis

This stage is the stage of analysis of the system running, then an analysis of the input, output and system processes. The analysis aims to further recognize whether the system used so far is still appropriate to use. In a running system (the old system) the possibility of an error rate that occurs is still very high. Data processing system that is done manually causes a large error. In terms of data storage is done in a file without a good database so it is likely to occur. File damage is getting higher. This can hamper the scheduling process.

3.4 Non Functional Requirements

Non-functional Requirements or non-functional requirements are descriptions of the features, characteristics and other limitations that define a satisfactory system. Non-functional requirements in information system course scheduling as follows:

1. The system can be run by several web browser software including Internet Explore, Google Chrome and, Safari, and Mozilla Firefox.
2. The process of the user opening an article / document to be read until the system issues / displays the article, takes no more than 10 seconds.
3. The system must be able to ensure that the data used in the system must be protected from unauthorized access.
4. The system has a display (between faces) that is easy to understand.
5. Login to the system requires a login using a username and password that can only be done by admin password used for login is encrypted (can not be seen and read directly).
6. This system uses information when the username, password, and data entered are incorrect in the form of an alert / warning containing an error message.

CHAPTER 6. SYSTEM TEST

6.1 Test Methods

After the implementation phase is finished, it is followed by testing of the implementation that has been made. System testing is carried out with the aim to ensure the system is built in accordance with the results of analysis and design so that a final conclusion can be made. Testing system functionality is done by the blackbox method.

6.2 Test Environment

6.2.1 Unit Test Tools

One method of testing this type is known as blackbox testing. The results of this test can be seen in table 6.1.

Table 6.1 System Testing Results by the Blackbox Method

No	Test Object	Expected Results	Results
1.	Save and Cancel data buttons on the project submenu add and edit lecturer	Save: save project lecturer data that has been entered into the database. Cancel: cancel to add and edit project lecturer data that has been entered into the database.	success
2.	Add, Edit, and Delete buttons on the project lecturer menu.	Add: displays the added activity page. Edit: change activity data. Delete: delete activity data.	success
3.	Search and Clear buttons on the project lecturer menu	Search: to find data with certain keywords. Clear: to clear the search.	success
4.	Save and Cancel data buttons on the project submenu add and edit course	Save project course data that has been entered into the database. Cancel: cancel to add and edit project course data that has been entered into the database.	success
5.	Add, Edit, and Delete buttons on the project course menu	Add: displays the added activity page. Edit: change activity data. Delete: delete activity data.	success

6.	Search and Clear buttons on the project course menu	Search: to find data with certain keywords. Clear: to clear the search.	success
7.	Save and Cancel data buttons on the project submenu add and edit supporting lecture	Save: save project supporting lecture data that has been entered into the database. Cancel: cancel to add and edit project supporting lecture data that has been entered into the database.	success
10.	Add button on the project supporting lecture menu	displays the added activity page.	success
11.	Search and Clear buttons on the project supporting lecture menu	Search: to find data with certain keywords. Clear: to clear the search.	success
12.	Save and Cancel data buttons on the project submenu add and edit classroom	Save: save project classroom data that has been entered into the database. Cancel: cancel to add and edit project classroom data that has been entered into the database.	success
13.	Add, Edit, and Delete buttons on the project classroom menu	Add: displays the added activity page. Edit: change activity data. Delete: delete activity data.	success
14.	Save and Cancel data buttons on the project submenu add and edit time	Save: save project time data that has been entered into the database. Cancel: cancel to add and edit project time data that has been entered into the database.	success
15.	Add, Edit, and Delete buttons on the project time menu	Add: displays the added activity page. Edit: change activity data. Delete: delete activity data.	success
16.	Save and Cancel data buttons on the project submenu add and edit days	Save: save project days data that has been entered into the database.	success

		Cancel: cancel to add and edit project days data that has been entered into the database.	
17.	Add, Edit, and Delete buttons on the project days menu	Add: displays the added activity page. Edit: change activity data. Delete: delete activity data.	success
18.	Save data button on the project unavailable time menu	save project unavailable time data that has been entered into the database.	success
19.	Process data button on the project schedulling menu	processing data to produce a schedule.	success
20.	Export to Excel button on the project scheduling menu	to import report to excel and print report result.	success

From the above blackbox testing it can be concluded that the features contained in the system can run well and in accordance with the expected results.

The result of evaluation per menu in the application can be seen in the table testing below:

Table 6.2 Table Testing

No.	Menu Name	Result
1	Login Menu	Run Successfully
2	Lecturer Input Data Menu	Run Successfully
3	Course Input Data Menu	Run Successfully
4	Room Input Data Menu	Run Successfully
5	Time Input Data Menu	Run Successfully
6	Supporting Lecturer Input Menu	Run Successfully
7	Unavailable Time Input Menu	Run Successfully
8	Generate Schedule Menu	Run Successfully
11	Logout	Run Successfully

6.2.2 Performance Test Process

Performance testing is a test conducted 10 times using default parameter values and input parameter values from the user on the course scheduling system. This test is carried out to find out whether the course scheduling application using genetic algorithms successfully produces a course schedule that meets all constraints. The results of this test can be seen in table 6.2.

1. Performance Testing Using Default Parameter Values This test uses default parameter values, namely:
   a. Crossover Probability (Pc) = 0.9
   b. Mutation Probability (Pm) = 0.1

Following are the test results of the course scheduling process using genetic algorithms with 10 generations, can be explained in the table below.

Table 6.3 Performance Testing Using Default Parameter Values

No	Best Chromosome	Generation	Solution / Chromosome contents	Test Result
1	Chromosome 2	7th	1 2 3 4 7 8 8 9 9 10 10 11 11 12	success
2	Chromosome 1	3rd	1 2 3 4 5 8 8 9 9 10 10 11 11 12	success
3	Chromosome 4	1st	1 2 2 5 8 9 9 10 10 11 11 12 12 13	success
4	Chromosome 7	5th	1 2 3 4 5 8 8 9 9 10 10 11 11 12	success
5	Chromosome 8	3rd	1 2 3 4 7 8 8 9 9 10 10 11 11 12	success
6	Chromosome 3	8th	1 2 2 5 8 9 9 10 10 11 11 12 12 13	success
7	Chromosome 2	6th	1 2 3 4 5 8 8 9 9 10 10 11 11 12	success
8	Chromosome 1	7th	1 2 2 5 8 9 9 10 10 11 11 12 12 13	success
9	Chromosome 6	2nd	1 2 3 4 5 8 8 9 9 10 10 11 11 12	success
10	Chromosome 9	1st	1 2 3 4 7 8 8 9 9 10 10 11 11 12	success

Based on Table 6.2 it can be explained that the first test resulted in chromosome 2 in the 7th generation as a course scheduling solution that did not have a violation of the constraints. Likewise, the second test up to the tenth test.

2. Performance Testing Using User Input Parameter Value

   This test uses the parameter value inputted by the user of the system. Test results are explained in the table below.

Table 6.4 Performance Testing Using User Input Parameter Value

No	Pc	Pm	Gen	Chro	Solution	Time	Result
1	0.25	0.1	20	3	1 2 2 5 8 9 9 10 10 11 11 12 12 13	1 min 58 sec	success
2	0.25	0.9	25	1	1 2 3 4 5 8 8 9 9 10 10 11 11 12	2 min 10 sec	success
3	0.5	0.8	30	2	1 2 3 4 7 8 8 9 9 10 10 11 11 12	2 min 22 sec	success
4	0.5	0.7	40	1	1 2 3 4 5 9 9 10 10 11 11 12 12 13	2 min 46 sec	success

5	0.75	0.6	50	6	1 2 2 5 8 9 9 10 10 11 11 12 12 13	3 min 10 sec	success
6	0.75	0.5	70	5	1 2 3 4 5 8 8 9 9 10 10 11 11 12	3 min 58 sec	success
7	0.8	0.4	80	4	1 2 3 4 7 8 8 9 9 10 10 11 11 12	4 min 22 sec	success
8	0.8	0.3	90	8	1 2 2 5 8 9 9 10 10 11 11 12 12 13	4 min 46 sec	success
9	0.9	0.2	100	7	1 2 3 4 5 8 8 9 9 10 10 11 11 12	5 min 56 sec	success
10	0.9	0.1	120	3	1 2 3 4 7 8 8 9 9 10 10 11 11 12	7 min 40 sec	success

Based on Table 6.3 it can be explained that the first test with a crossover probability value (Pc) = 0.25 and a mutation probability (Pm) = 0.1 of 20 generations produces chromosome 3 as a scheduling solution for subjects that meet all the constraints which takes 1 minute 58 seconds to generate. Likewise, the second test up to the tenth test.

6.2.3 Unit Test Case Analysis

Unit Test Case Analysis is a test carried out with requesting approval from the user for the output generated by the project scheduling system. Respondents who tested were 10 people. The complete unit test case analysis can be seen in in the table below.

Table 6.5 Results of Unit Test Case Analysis

No	Test Item	Yes	No	Respondents
1.	Use of a previous project scheduling system by respondents		√	10 people
2.	Respondents have seen the project scheduling system using genetic algorithms before		√	10 people
3.	User interface looks boring or not attractive		√	10 people
4.	Menus in the system are difficult to use		√	10 people
5.	The system can help search for project schedule solutions	√		10 people
6.	The colors and images used by the system are match and suitable	√		10 people
7.	The system can provide project scheduling solutions	√		10 people
8.	The results of the system are satisfying	√		10 people
9.	This system remains in use for a long period and for the future	√		10 people

Based on Table 6.1 it can be explained that 10 respondents agreed that the project scheduling system using genetic algorithms can run well as expected and is not difficult to use.

6.3 Test Conclusion

From the system testing that has been done, the project scheduling system using genetic algorithms can produce an optimal project schedule, several conclusions can be drawn, as follows:

1. The Blackbox test shows that the components of the system are functioning as expected.
2. Performance testing conducted using the default parameter values and input from the user worked well.
3. Based on performance testing, it can also be concluded that the genetic algorithm takes a long time if the iteration value is large, because in the process this algorithm carries out the process of generation.
4. Testing conducted on 10 respondents who have used the project scheduling system using genetic algorithms shows that the output generated by the system can be a solution in scheduling work or activities in a project.

CHAPTER 7. CONCLUSION AND PROSPECT

7.1 Summary

From the results of the analysis carried out in the last chapter, several conclusion points can be drawn as follows:

1. Genetic algorithm can be used to optimize the creation of course schedules.
2. In optimizing the preparation of course schedules four parameters are needed, there are lecturer, room, days, and hours. The size of each parameter greatly affects the resulting course schedule.
3. The course scheduling application with genetic algorithm can schedule optimally means there is no conflicting schedule.
4. The course scheduling application with genetic algorithm is able to make a schedule quickly.

7.2 Further Outlook

This college scheduling application is not perfect and can still be improved for more complex scheduling and wider scope. To develop a system that is created, it is necessary to provide suggestions that can help the process, including:

1. Making course scheduling information system using a genetic algorithm for all the department in university.
2. It is expected that this system will be able to do multilevel login in the future where there are more than one actor with different roles.
3. Scheduling systems can be developed by combining with academic information systems.

LIST OF REFERENCES

Haldurai, L., Madhubala, T., & Rajalkshmi, R. (2016, Oct 31). A Study on Genetic Algorithm and its Applications. *International Journal of Computer Sciences and Engineering, 4*(10).

Suwirmayanti, N. G., Sudarsana, I. M., & Darmayasa, S. (2016). Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran. *Journal of Applied Intelligent System*.

Ridwan, M. (2016). Prototype Sistem Pendukung Keputusan untuk Penetapan Jadwal Kuliah Menggunakan Algoritma Genetika. *2*(2).

Jing, W. W. (2018). Improved Adaptive Genetic Algorithm for Course Scheduling in Colleges and Universities. *International Journal of Emerging Technologies in Learning, 13*(6).

Tabassum, M., & Mathew, K. (n.d.). A Genetic Algorithm Analysis towards Optimization solutions. *International Journal of Digital Information and Wireless Communications (IJDIWC)*.

Sugeha, I. H., Inkiriwang, R. L., & Pratasis, P. A. (2019). Optimasi Penjadwalan Menggunakan Metode Algoritma Genetika Pada Proyek Rehabilitasi Puskesmas Minanga. *Jurnal Sipil Statik, 7*(12).

Kumar, M. (n.d.). Genetic Algorithm - an Approach to Solve Global Optimization Problems. *Journal of Computer Science and Engineering*.

Affenzeller, M., Wagner, S., Winkler, S., & Beham, A. (2009). *Genetic Algorithms and Genetic Programming Modern Concepts and Pratical Applications.* USA: CRC Press.

Massalesse, J. (2011). Pendekatan Algoritma Genetika pada Peminimalan Fungsi Ackley menggunakan Representasi Biner. *Jurnal Matematika, Statistika, & Komputasi, 7*(2).

Abdy, M., Wahyuni, M. S., & Ilmi, N. (2016). Algoritma Genetika dan Penerapannya dalam Mencari Akar Persamaan Polinomial. *Jurnal Saintifik*.

Priambodo, M. A., Nhita, F., & Aditsania, A. (20166). Penjadwalan Mata Kuliah Menggunakan Metode Hybrid Algoritma Genetika Dan Algoritma Koloni Semut. *e-Proceeding of Engineering, XX*(2).

Worzel, W. P., Yu, J., Almal, A. A., & Chinnaiyan, A. M. (2009, February). Applications of Genetic Programming in Cancer Research. *The International Journal of Biochemistry & Cell Biology, 41*(2).

Mahmudy, W. F., & Mawaddah, N. K. (2006). Optimasi Penjadwalan Ujian Menggunakan Algoritma Genetika. *2*(2).

Kachitvichyanukul, V. (2012, August 18). Comparison of Three Evolutionary Algorithms: GA, PSO, and DE. *Industrial Engineering & Management Systems, 11*(3).

Janah, N. Z., Brayudi, I., & Maria. (2016). Sistem Pendukung Keputusan Pembagian Beban Mengajar Dosen dengan Metode Algoritma Genetika Studi Kasus: Politeknik Negeri Batam. *Seminar Nasional Teknologi Informasi dan Multimedia*.

Suwirmayanti, N. G., Sudarsana, I. M., & Darmayasa, S. (2016). Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran. *Journal of Applied Intelligent System, 1*(3).

Haldurai, L., Madhubala, T., & Rajalakshmi, R. (2016, Oct 31). A Study on Genetic Algorithm and its Applications. *International Journal of Computer Sciences and Engineering, 4*(10).

Budhi, G. S., Gunadi, K., & Wibowo, D. A. (2011). Genetic Algorithm for Scheduling Courses.

Hassanat, A., Almohammadi, K., Alkafaween, E., Abunawas, E., Hammouri, A., & Prasath, V. S. (2019, December 10). Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach.

Fridawati, A., Wiwiet, & Arief. (2019, June). Analysis of Sales Infromation System. *Journal of Electrical Engineering and Computer Sciences, 4*(1).

Seidl, M., Scholz, M., Huemer, C., & Kappel, G. (2014). *UML@Classroom An Introduction to Object-Oriented Modeling.* German: Springer.

Rob, P., Coronel, C., & Crockett, K. (2008). *Database Systems: Design, Implementation & Management.* London: Gaynor Redvers-Mutton.

Putri, N. E., Marwan, S., & Hariyono, T. (n.d.). Aplikasi Berbasis Multimedia Untuk Pembelajaran Hardware Komputer. *Jurnal Edik Informatika*.

Kalsum, T. U., & Siswanto. (2012). Sistem Antrian pada Puskesmas Sukamerindu. *Jurnal Media Infotama*.

Pramana, Y. A., Supraptono, E., & Pribadi, F. S. (2013). Aplikasi Microsoft Office Excel 2010 untuk Menganalisis Butir Soal Pilihan Ganda. *Jurnal Teknik Elektro*.

Umar, U. (n.d.). Perancangan Sistem Informasi Administrasi Kesiswaan Pada Madrasah Aliyah Miftahul Jannah dengan Metodologi Berorientasi Objek. *Jurnal SIFOM*.

Palit, R. V., Rindengan, Y. D., & Lumenta, A. S. (2015). Rancangan Sistem Informasi Keuangan Gereja Berbasis Web Di Jemaat GMIM Bukit Moria Malalayang. *E-Journal Teknik Elektro dan Komputer*.

Davison, A., & Hinkley, D. V. (2008). *Bootstrap methods and their application.* United States of America: CAMBRIDGE UNIVERSITY PRESS.

Destinigrum, M., & Adrian, Q. J. (2017). Sistem Informasi Penjadwalan Dokter Berbasis Web dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre). *Jurnal TEKNOINFO*.

Syifani, D., & Dores, A. (2018). Aplikasi Sistem Rekam Medis di Puskesmas Kelurahan Gunung. *Jurnal Sistem Informasi, Teknologi Informatika dan Komputer, 9*(1).

Pradiatininggtyas, D., & Suparwanto. (2017). E-Learning Sebagai Media Pembelajaran Berbasis Web Pada SMK N 4 Purworejo. *Indonesian Journal on Networking and Security, 7*(2).

Pahlevi, O., Mulyani, A., & Khoir, M. (2018). Sistem Informasi Inventori Barang Menggunakan Metode Object Oriented di PT. Livaza Teknologi Indonesia Jakarta. *Jurnal PROSISKO, 5*(1).

Sibagariang, S. (2016). Penerapan Web Service Pada Perpustakaan Berbasis Android. *Jurnal Mahajana Informasi, 1*(1).

Swara, G. Y., & Pebriadi, Y. (2016). Rekayasa Perangkat Lunak Pemesanan Tiket Bioskop Berbasis Web. *TEKNOIF, 4*(2).

Sukrianto, D. (2017). Penerapan Teknologi Barcode Pada Pengolahan Data Pembayaran Sumbangan Pembinaan Pendidikan (SPP). *Intra-Tech, 1*(2).

K, T. O., Irfan, M., & Nurpianti, A. (2013). Pembuatan Aplikasi Anbiyapedia Ensiklopedi Muslim Anak Berbasis Web. *VII*(1).

Muftikhali, Q. E., Danar, A., Kusumawati, A., & Hidayat, S. (2017). Algoritma Genetik Dalam Menentukan Rute Optimal Topologi Cincin Pada WAN. *JTIIK, 4*(1), 62-68.