# Scheduling Courses using Genetic Algorithms

Andysah Putera Utama Siahaan
Faculty of Computer Science
Universitas Pembangunan Panca Budi
Jl. Jend. Gatot Subroto Km. 4,5 Sei Sikambing,
20122, Medan, Sumatera Utara,
Indonesia

## ABSTRACT
Preparation of courses at every university is done by hand. This method has limitations that often cause collisions schedule. In lectures and lab scheduling frequent collision against the faculty member teaching schedule, collisions on the class schedule and student, college collision course with lab time, the allocation of the use of the rooms were not optimal. Heuristic method of genetic algorithm based on the mechanism of natural selection; it is a process of biological evolution. Genetic algorithms are used to obtain optimal schedule that consists of the initialization process of the population, fitness evaluation, selection, crossover, and mutation. Data used include the teaching of data, the data subjects, the room data and time data retrieved from the database of the Faculty of Computer Science, Universitas Pembangunan Panca Budi. The data in advance through the stages of the process of genetic algorithms to get optimal results The results of this study in the form of a schedule of courses has been optimized so that no error occurred and gaps.

## General Terms
Artificial Intelligence

## Keywords
Genetic Algorithm, Scheduling

## 1. INTRODUCTION
The process of teaching and learning has become a routine that is always carried out by the university to carry out the day-to-day activities. In the process of teaching and learning will involve courses that will be given to students who participate in it [5]. The number of courses each semester hobbled right in not a few. Also, there are compulsory courses, and there are also elective courses. The preparation of this course became a classic problem that until now has always been a dilemma of faculty in particular to the faculty staff. In setting the schedule of college faculty often encounter obstacles, so that the result of the preparation always has a discrepancy with hopes to achieve.

At this time only rely on the strength of the faculty manual calculations in the preparation of the schedule of courses, so there is still an error occurred and irregularities in the final product scheduled lectures per semester. Errors obtained such a schedule mutually collisions for each faculty; space is not optimal, the lecturer who was unable to attend because they do not fit the schedule, students entering different classes at the same time and so forth. While irregularities obtained as the not uniform amount charged class faculty against faculty, where there are lecturers who get the number of teaching hours, and there is also a lot to get the number of hours of teaching a little.

Genetic algorithms are one way to address the problem for the preparation courses. These algorithms can produce optimal schedules subjects [1][2][7]. Application of this method will reduce the workload of faculty to students assembles all the more so if the students owned by large numbers of faculty. From the financial point of view, this method plays an important role to save time and costs incurred to prepare a report schedule faculty courses. A genetic algorithm is a heuristic search algorithm based on the mechanism of natural selection, better known by a process of biological evolution. This algorithm can produce optimal results with a fast time and has a very large solution space.

## 2. RELATED WORK
Genetic algorithms work in many cases of scheduling problem [4][6][8]. Travelling salesmen problem is one of the problems. It is done to find the solution of all the nodes that pass and looking for the most optimal value or the shortest path passes on the track. The genetic algorithm can find the optimal solution in each generation were executed but the result is not a correct answer to find the desired distance. This research tried to combine the Knapsack and genetic algorithms. Knapsack Problem is a combinatorial optimization problem. For example, given a set of items by weight and value, then do the selection of these items to put in a bag with a limited capacity.

It can specify the desired weight in a container. Knapsack itself has at least two parameters as determining whether the fitness of a population approaching with a predetermined solution. The parameters used in this study is the number of nodes and weight range. The number of nodes is the number of point coordinates that will pass while the weighted distance is the distance between the accumulated number of nodes to go back to the origin node. In this algorithm is expected to achieve a solution that can generate fitness = 1 or at least closer to that number [3].

## 3. PROPOSED WORK
### 3.1 Courses Design
As noted in the Introduction, case studies taken in this study is the Faculty of Computer Science Universitas Pembangunan Panca Budi. For a schedule can be made correctly, there are some scheduling rules must be observed. The factors that influence the preparation of the schedule to include are:

**1. Lecturer**
A professor can not teach several courses at the same time. Also, sometimes a teacher can only teach at odd hours and certain days only, so it is necessary to know the specific schedules that can not be subject to another.

**2. Space**
Given the limited amount of space owned, it should be noted the available space so as not to interfere with the course of the lecture. The schedule should only occupy space there.

**3. Time**
Time is a time limit of lectures per subject, and there are certain hours where the lecture is limited to certain hours such as Friday schedule from 07.30 until 12.00 and resumed at 13.30.

**4. Course**
Given each course has a semester of courses that are taught, the need for rules that restrict the scheduling of courses, so that the courses were by the rules of scheduling.

## 3.2 Genetic Step

Model of Genetic Algorithm to be used for optimization is as follows:

**1. Selection**
In the selection, an assessment of the value of fitness. As a result, the fitness which has the best quality of chromosomes have a chance in the next generation. Selection used is the selection of the roulette wheel. In the implementation of this selection to consider the number of the population so that the population is not too much and take a long time, and the population is also not too little will result in chromosomal similarity.

**2. Crossover**
Crossover used was crossing one point with a permutation. Selection of chromosomes is determined by probability. Many genes are exchanged depends on the determination of the initial parameters. In doing a crossover, each of the two chromosomes will produce two new offspring as the best genes.

**3. Mutation**
Mutation is performed after the crossover operation is completed. This mutation technique is done by swapping genes randomly. In this process to consider the mutation rate and the degree of probability of a mutation. If the mutations are likely the best chromosome loss. However, if too little mutations, chromosome will long to find the optimal solution.

**4. Determination of Fitness**
Fitness determination is the provision of value that determines whether a process is achieved genetic algorithms What does this process is the process of making the schedule according to chromosome was selected by processing by calculating how close to the value of fitness.

## 3.3 Courses List

This sections shows the list of the disciplines offered by the Faculty of Computer Science.

| NO | CODE | SUBJECT | CRD | SEM. |
|---|---|---|---|---|
| **1** | MPK-370-101 | Pendidikan Pancasila dan Kewarganegaraan | 2 | |
| **2** | MPK-370-102 | Pendidikan Agama | 2 | |
| **3** | MBB-370-103 | Ilmu Sosial dan Budaya Dasar | 2 | **I** |
| **4** | MPK-370-104 | Bahasa Inggris I | 2 | |
| **5** | MPB-370-105 | Metafisika I | 2 | |
| **6** | MKK-370-106 | Matematika Diskrit | 2 | |
| **7** | MKK-370-107 | Pengantar Teknologi Informasi | 2 | |
| **8** | MKK-370-108 | Praktek Pengantar Teknologi Informasi | 1 | |
| **9** | MKK-370-109 | Algoritma & Pemrograman | 2 | |
| **10** | MKK-370-110 | Praktek Algoritma & Pemrograman | 1 | |
| **11** | MKB-370-111 | Pengantar Manajemen Umum | 2 | |
| **CREDIT = 20** | | | | |
| **12** | MPK-370-201 | Bahasa Inggris II | 2 | |
| 13 | MPB-370-202 | Metafisika II | 2 | |
| **14** | MKK-370-203 | Aljabar Linear & Matriks | 2 | |
| 15 | MPK-370-204 | Bahasa Indonesia (Tata bahasa ilmiah) | 2 | |
| **16** | MKK-370-205 | Sistem Operasi | 2 | |
| 17 | MKK-370-206 | Praktek Sistem operasi | 1 | **II** |
| **18** | MKK-370-207 | Pemrograman Berorientasi Objek I (.NET) | 3 | |
| 19 | MKK-370-208 | Praktek Pemrograman Berorientasi Objek I (.NET) | 1 | |
| **20** | MKK-370-209 | Statistik dan Probabilitas | 2 | |
| 21 | MPB-370-210 | Etika Profesi IT | 3 | |
| **CREDIT = 20** | | | | |
| 22 | MPB-370-301 | Metafisika III | 2 | |
| 23 | MKB-370-302 | Kewirausahaan Teknologi Informasi | 2 | |
| 24 | MKB-370-303 | Pemrograman Berorientasi Objek II (Java) | 3 | |
| 25 | MKB-370-304 | Praktek Pemrograman Berorientasi Objek II (Java) | 1 | |
| 26 | MKK-370-305 | Struktur Data | 3 | **III** |
| 27 | MKK-370-306 | Praktek Struktur Data | 1 | |
| 28 | MKK-370-307 | Elektronika Dasar | 3 | |
| 29 | MKK-370-308 | Praktek Elektronika Dasar | 1 | |
| 30 | MKK-370-309 | Sistem Basis Data | 3 | |
| 31 | MKK-370-310 | Komunikasi Data | 3 | |
| **CREDIT = 22** | | | | |
| 32 | MPB-370-401 | Troubleshooting dan Maintenance | 3 | |
| 33 | MPB-370-402 | Praktek Troubleshooting dan Maintenance | 1 | |
| 34 | MKB-370-403 | Komputer Grafik | 3 | |
| 35 | MKB-370-404 | Praktek Komputer Grafik | 1 | **IV** |
| 36 | MKK-370-405 | Organisasi & Arsitektur Komputer | 2 | |
| 37 | MPK-370-406 | Sistem Basis Data Lanjutan | 3 | |
| 38 | MPK-370-407 | Praktek Sistem Basis Data Lanjutan ((MySql / SQL Server / Oracle) | 1 | |
| 39 | MKK-370-408 | Jaringan Komputer | 3 | |

| | | | | |
|---|---|---|---|---|
| 40 | MKK-370-409 | Praktek Jaringan Komputer | 1 | |
| 41 | MPB-370-410 | Metode Penelitian | 2 | |
| **CREDIT = 20** | | | | |
| 42 | MKB-370-501 | Rekayasa Perangkat Lunak | 3 | |
| 43 | MKK-370-502 | Pemrograman Internet | 2 | |
| 44 | MKK-370-503 | Praktek Pemrograman Internet (HTML 5, CSS, Java Script/ Jquery) | 1 | |
| 45 | MKB-370-504 | Dasar Sistem Digital | 2 | |
| 46 | MKB-370-505 | Desain Berbasis Komputer | 2 | **V** |
| 47 | MKB-370-506 | Jaringan Komputer Lanjut (Router) | 3 | |
| 48 | MKB-370-507 | Praktek Jaringan Komputer Lanjut (Router) | 1 | |
| 49 | MKB-370-508 | Simulasi dan Pemodelan | 1 | |
| 50 | MKB-370-509 | Sistem Pendukung Keputusan | 3 | |
| **CREDIT = 18** | | | | |
| 51 | MKK-370-601 | Kecerdasan Buatan | 3 | |
| 52 | MKK-370-602 | Praktek Kecerdasan Buatan | 1 | |
| 53 | MKK-370-603 | Metode Numerik | 2 | |
| 54 | MKB-370-604 | Multimedia | 2 | |
| 55 | MKB-370-605 | Praktek Multimedia | 1 | |
| 56 | MKB-370-606 | Sistem Informasi Geografis | 3 | **VI** |
| 57 | MKB-370-607 | Praktek Sistem Informasi Geografis | 1 | |
| 58 | MBB-370-608 | Kerja Praktek | 2 | |
| 59 | MKB-370-609 | Robotika | 3 | |
| 60 | MKB-370-610 | Praktek Robotik | 1 | |
| **CREDIT = 19** | | | | |
| 61 | MKB-370-701 | Interface | 3 | |
| 62 | MKB-370-702 | Praktek Interface | 1 | |
| 63 | MKK-370-703 | Aplikasi Mobile (Android) | 2 | |
| 64 | MKB-370-704 | Image Processing | 2 | |
| 65 | MKB-370-705 | Keamanan Komputer | 2 | |
| 66 | MKB-370-706 | Proyek Teknologi Informasi | 2 | **VII** |
| 67 | MKK-370-707 | Interaksi Manusia dan Komputer | 2 | |
| 68 | MKB-370-708 | Embedded System | 3 | |
| 69 | MKB-370-709 | Praktek Embedded System | 1 | |
| 70 | MKB-370-710 | Analisa Kerja Sistem Komputer | 2 | |
| **CREDIT = 20** | | | | |
| 71 | MKB-370-801 | Kecakapan Antar Personal | 2 | |
| 72 | MBB-370-802 | Seminar | 2 | **VIII** |
| 73 | MBB-370-803 | Sidang | 4 | |
| **CREDIT = 8** | | | | |

# 4. RESULT AND DISCUSSION

This stage will report on the results of the scheduling of the courses. It develops a software system is the evaluation stage. Implementation and testing of this section will be done by the design that has been described in previous chapters. To determine whether a software implementation is successful or not, required testing. Here are the results of the implementation and testing of applications that have been built.

## 4.1 Genetic Process

This section describes the steps of the process of finding an optimal solution in the genetic algorithm. The process can be seen in the section below:

```
Indv [1]  --> D : 13  S : 0      K : 40     R : 75     J : 8
          Error : 136
Indv [2]  --> D : 15  S : 0      K : 45     R : 55     J : 11
          Error : 126
Indv [3]  --> D : 11  S : 0      K : 33     R : 79     J : 17
          Error : 140
Indv [4]  --> D : 18  S : 0      K : 39     R : 72     J : 16
          Error : 145
Indv [5]  --> D : 26  S : 0      K : 39     R : 64     J : 6
          Error : 135
Indv [6]  --> D : 18  S : 0      K : 42     R : 71     J : 16
          Error : 147
Indv [7]  --> D : 14  S : 0      K : 44     R : 68     J : 11
          Error : 137
Indv [8]  --> D : 14  S : 0      K : 31     R : 65     J : 8
          Error : 118
Indv [9]  --> D : 22  S : 0      K : 47     R : 74     J : 12
          Error : 155
Indv [10] --> D : 17  S : 0      K : 35     R : 72     J : 10
          Error : 134

==================== End of Generation 1
====================

Indv [1]  --> D : 4   S : 0      K : 20     R : 35     J : 0
          Error : 59
Indv [2]  --> D : 2   S : 0      K : 20     R : 30     J : 0
          Error : 52
Indv [3]  --> D : 14  S : 0      K : 31     R : 65     J : 8
          Error : 118
Indv [4]  --> D : 26  S : 0      K : 39     R : 64     J : 6
          Error : 135
Indv [5]  --> D : 18  S : 0      K : 42     R : 71     J : 16
          Error : 147
Indv [6]  --> D : 13  S : 0      K : 40     R : 75     J : 8
          Error : 136
Indv [7]  --> D : 13  S : 0      K : 40     R : 75     J : 8
          Error : 136
Indv [8]  --> D : 22  S : 0      K : 47     R : 74     J : 12
          Error : 155
Indv [9]  --> D : 17  S : 0      K : 35     R : 72     J : 10
          Error : 134
Indv [10] --> D : 13  S : 0      K : 40     R : 75     J : 8
          Error : 136

==================== End of Generation 2
====================

Indv [1]  --> D : 1   S : 0      K : 12     R : 15     J : 0
          Error : 28
Indv [2]  --> D : 6   S : 0      K : 27     R : 38     J : 0
          Error : 71
Indv [3]  --> D : 26  S : 0      K : 39     R : 64     J : 6
          Error : 135
Indv [4]  --> D : 13  S : 0      K : 40     R : 75     J : 8
          Error : 136
```

Indv [5] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [6] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [7] --> D : 13 S : 0 K : 40 R : 75 J : 8
Error : 136
Indv [8] --> D : 2 S : 0 K : 20 R : 30 J : 0
Error : 52
Indv [9] --> D : 17 S : 0 K : 35 R : 72 J : 10
Error : 134
Indv [10] --> D : 14 S : 0 K : 31 R : 65 J : 8
Error : 118

========== End of Generation 3
==========

Indv [1] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [2] --> D : 2 S : 0 K : 25 R : 35 J : 1
Error : 63
Indv [3] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [4] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [5] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [6] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [7] --> D : 26 S : 0 K : 39 R : 64 J : 6
Error : 135
Indv [8] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [9] --> D : 13 S : 0 K : 40 R : 75 J : 8
Error : 136
Indv [10] --> D : 13 S : 0 K : 40 R : 75 J : 8
Error : 136

========== End of Generation 4
==========

Indv [1] --> D : 2 S : 0 K : 9 R : 19 J : 0
Error : 30
Indv [2] --> D : 1 S : 0 K : 13 R : 14 J : 0
Error : 28
Indv [3] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [4] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [5] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [6] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [7] --> D : 2 S : 0 K : 25 R : 35 J : 1
Error : 63
Indv [8] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [9] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [10] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28

========== End of Generation 5
==========

Indv [1] --> D : 1 S : 0 K : 10 R : 13 J : 1
Error : 25
Indv [2] --> D : 1 S : 0 K : 6 R : 13 J : 0
Error : 20
Indv [3] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [4] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [5] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [6] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [7] --> D : 6 S : 0 K : 27 R : 38 J : 0
Error : 71
Indv [8] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [9] --> D : 2 S : 0 K : 9 R : 19 J : 0
Error : 30
Indv [10] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23

========== End of Generation 6
==========

Indv [1] --> D : 1 S : 0 K : 9 R : 8 J : 0
Error : 18
Indv [2] --> D : 1 S : 0 K : 7 R : 10 J : 0
Error : 18
Indv [3] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [4] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [5] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [6] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [7] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [8] --> D : 1 S : 0 K : 10 R : 13 J : 1
Error : 25
Indv [9] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [10] --> D : 1 S : 0 K : 10 R : 13 J : 1
Error : 25

========== End of Generation 7
==========

Indv [1] --> D : 0 S : 0 K : 3 R : 10 J : 0
Error : 13
Indv [2] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [3] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [4] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [5] --> D : 1 S : 0 K : 10 R : 13 J : 1
Error : 25
Indv [6] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [7] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [8] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [9] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [10] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23

========== End of Generation 8
==========

Indv [1] --> D : 1 S : 0 K : 5 R : 8 J : 0
Error : 14
Indv [2] --> D : 1 S : 0 K : 6 R : 9 J : 0
Error : 16
Indv [3] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [4] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [5] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [6] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10

Indv [7] --> D : 4 S : 0 K : 20 R : 35 J : 0
Error : 59
Indv [8] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [9] --> D : 0 S : 0 K : 3 R : 10 J : 0
Error : 13
Indv [10] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23

========== End of Generation 9
==========

Indv [1] --> D : 0 S : 0 K : 6 R : 9 J : 1
Error : 16
Indv [2] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [3] --> D : 1 S : 0 K : 6 R : 9 J : 0
Error : 16
Indv [4] --> D : 1 S : 0 K : 12 R : 15 J : 0
Error : 28
Indv [5] --> D : 2 S : 0 K : 10 R : 11 J : 0
Error : 23
Indv [6] --> D : 1 S : 0 K : 5 R : 8 J : 0
Error : 14
Indv [7] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [8] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [9] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [10] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10

========== End of Generation 10
==========

Indv [1] --> D : 0 S : 0 K : 8 R : 14 J : 0
Error : 22
Indv [2] --> D : 0 S : 0 K : 3 R : 5 J : 0
Error : 8
Indv [3] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [4] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [5] --> D : 1 S : 0 K : 6 R : 9 J : 0
Error : 16
Indv [6] --> D : 0 S : 0 K : 6 R : 9 J : 1
Error : 16
Indv [7] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [8] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [9] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [10] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10

========== End of Generation 11
==========

Indv [1] --> D : 0 S : 0 K : 1 R : 5 J : 0
Error : 6
Indv [2] --> D : 0 S : 0 K : 5 R : 2 J : 0
Error : 7
Indv [3] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [4] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [5] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [6] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [7] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [8] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [9] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [10] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4

========== End of Generation 12
==========

Indv [1] --> D : 0 S : 0 K : 0 R : 3 J : 0
Error : 3
Indv [2] --> D : 0 S : 0 K : 1 R : 1 J : 0
Error : 2
Indv [3] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [4] --> D : 1 S : 0 K : 5 R : 4 J : 0
Error : 10
Indv [5] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [6] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [7] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [8] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [9] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [10] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4

========== End of Generation 13
==========

Indv [1] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [2] --> D : 1 S : 0 K : 2 R : 3 J : 1
Error : 7
Indv [3] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [4] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [5] --> D : 0 S : 0 K : 1 R : 1 J : 0
Error : 2
Indv [6] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [7] --> D : 0 S : 0 K : 0 R : 3 J : 0
Error : 3
Indv [8] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [9] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [10] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4

========== End of Generation 14
==========

Indv [1] --> D : 0 S : 0 K : 0 R : 1 J : 0
Error : 1
Indv [2] --> D : 0 S : 0 K : 3 R : 2 J : 0
Error : 5
Indv [3] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [4] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [5] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [6] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [7] --> D : 0 S : 0 K : 2 R : 2 J : 0
Error : 4
Indv [8] --> D : 0 S : 0 K : 1 R : 1 J : 0
Error : 2

```
Indv [9]   --> D : 0   S : 0       K : 0       R : 3       J : 0
           Error : 3
Indv [10]  --> D : 0   S : 0       K : 2       R : 2       J : 0
           Error : 4

=================== End of Generation 15
===================

Indv [1]   --> D : 0   S : 0       K : 0       R : 0       J : 0
           Error : 0

=================== End of Generation 16
===================
```

## 4.2 Result

In the first generation, a population of 1 to 10 population has a value large error in which an error in each population is above 100. At the next generations, the error value will decrease gradually. If any value of the population still has an error value of more than 1, then the population is not yet optimal. The error value must be 0 to get the value Fitness = 1. After 16 generations, the population is worth error = 0 as shown below.

```
Indv [1]   --> D : 0   S : 0       K : 0       R : 0       J : 0
           Error : 0

=================== End of Generation 16
===================
```

# 5. CONCLUSION

Based on the analysis and testing conducted in the previous chapter, it can be concluded the genetic algorithm is a good algorithm in the process of optimization of scheduling courses. Genetic algorithms are very influenced by the random function, so it is not always the results obtained in the process of scheduling lectures and lab work to get the most optimal results. Selection of the genes in the process of crossover and mutation, can not be randomized because of differences in the range of each gene but directly elected. Based on testing with genetic input parameter values are the same or different, the scheduling process produces results and generation of different iterations this is because of the random function. Fingers and crossover and mutation probabilities used is 100%, because of all the chromosomes crossover and mutation. Genetic operators used in this study is Roulette wheel selection, crossover a cutting point, because it is more appropriate exchange mutation on chromosome representation and generate initialization population at the time of the initial population.

# 6. FUTURE SCOPE

The algorithm needs to be developed to gain more result. It needs to collaborate to other division to make the scheduling system more sophisticated.The Knapsack problem is the best method to limit the output. By combining this method with genetic algorithms, it makes the output better than before.

# 7. REFERENCES

[1] M. U. Siregar, "A New Approach to CPU Scheduling Algorithm: Genetic Round Robin," *International Journal of Computer Applications*, vol. 47, no. 19, pp. 18-25, 2012.

[2] Y. Li dan Y. Chen, "A Genetic Algorithm for Job-Shop Scheduling," *Journal of Software*, vol. 5, no. 3, pp. 269-273, 2010.

[3] A. P. U. Siahaan, "Adjustable Knapsack in Travelling Salesman Problem," *International Journal of Science & Technoledge*, vol. 4, no. 9, 2016.

[4] A. P. U. Siahaan, "Comparison Analysis of CPU Scheduling FCFS, SJF and Round Robin," *International Journal of Engineering Development and Research*, vol. 4, no. 3, pp. 124-132, 20 November 2016.

[5] U. Aickelin dan K. A. Dowsland, "An Indirect Genetic Algorithm for a Nurse Scheduling Problem," *Computers & Operations Research*, vol. 31, no. 5, pp. 761-778, 2004.

[6] M. Gupta dan S. Gupta, "Optimized Processor Scheduling Algorithms using Genetic Algorithm Approach," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 2, no. 6, pp. 2415-2417, 2013.

[7] F. A. Omara dan M. M. Arafa, "Genetic Algorithms for Task Scheduling Problem," *Journal of Parallel and Distributed Computing*, vol. 70, no. 1, pp. 13-22, 2010.

[8] H. Z. Jia, A. Y. C. Nee, J. Y. H. Fuh dan Y. F. Zhang, "A Modified Genetic Algorithm for Distributed Scheduling Problems," *Journal of Intelligent Manufacturing*, vol. 14, no. 3, p. 351, 2003.

# 8. AUTHOR PROFILE

**Andysah Putera Utama Siahaan** was born in Medan, Indonesia, in 1980. He received the S.Kom. degree in computer science from Universitas Pembangunan Panca Budi, Medan, Indonesia, in 2010, and the M.Kom. in computer science as well from the University of Sumatera Utara, Medan, Indonesia, in 2012. In 2010, he joined the Department of Engineering, Universitas Pembangunan Panca Budi, as a Lecturer, and in 2012 became a researcher. He is applying for his Ph.D. degree in 2016. He has written several international journals. He is now active in writing papers and joining conferences.