

**IE206 Project Report**

**2021-2022 Spring**

***Scheduling for a Local Hospital***

*Academic integrity is expected of all students of METU at all times, whether in the presence or absence of members of the faculty.*

*Understanding this, we declare that we shall not give, use, or receive unauthorized aid in this project.*

*Student ID Number Full Name Signature*

**2444974 Cafer Selli** 

**2587848 Zeynep Beril Şahin** metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Table of Contents

[Introduction 1](#_Toc107695530)

[Objective 1 1](#_Toc107695531)

[Flowchart of the Algorithm 3](#_Toc107695532)

[Objective 2 7](#_Toc107695533)

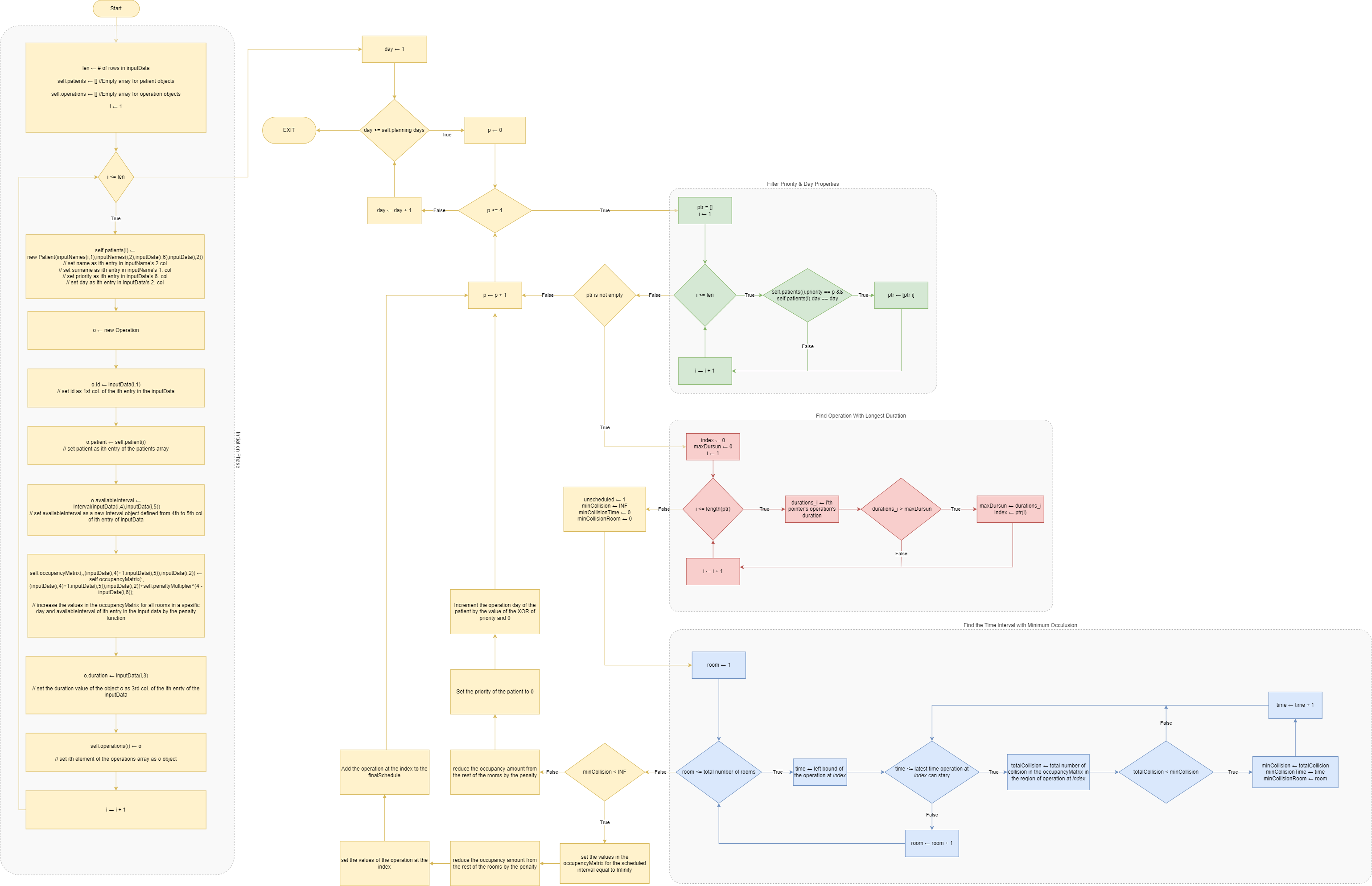
[Flowchart of the Algorithm 7](#_Toc107695534)

[Conclusion 9](#_Toc107695535)

Introduction

Solving scheduling problems of institutions enables decision-makers to develop and implement more efficient policies that directly serve the needs and expectations. By implementing these policies, resources are used in the most efficient and purposeful way.

This project embraces two different heuristics to solve the scheduling problem of operations in a hospital with a given number of rooms and a given time interval. For the given two different objective functions, which may differ in real life, the heuristics are transformed into algorithms to adjust the scheduling of operations expediently.

Objective 1

**Figure.1.1** Flowchart of Schedule Objective 1

In this objective, our purpose is to maximize the number of operations completed within the initial available time interval. In order to fulfill this objective, the designed algorithm considers each person one by one and arranges them to the interval where there is a minimum number of collisions occur. In terms of collision, how many of the patients are available at that time interval is considered. By counting the number of patients available at that time, we calculated the collision matrix. Using this calculation, a new matrix called *occupancyMatrix* is created. This matrix stores the count of the patient available at that time interval for each day and each room.

In order to start scheduling, we started from day one and found all of the patients that have the same day property. After patients are filtered, individuals who have the highest priority among them are chosen. In this array of chosen patients, one with the highest operation duration is selected. Our algorithm works in a way that, if the operation with the longest duration is scheduled first, the operations with shorter durations could be arranged to fit between them and achieve the desired highest scheduled patient number.

Patient with the highest priority and longest operation duration is shifted along with their available interval until it finds an interval that minimizes the number of collisions with other available time intervals. If there is no interval to fit, in other words, the total number of collisions is equal to the infinity; the next room is tested for the same procedure. In the end, if there is no room available, the operation is postponed unless it is already postponed. If it is scheduled, it sets the parameters of the operation and sets the start of the operation as the appointed starting time. Before starting to schedule another patient, the values in the *occupancyMatrix* at that day, selected room, and scheduled interval are set to infinity, and values in the rest of the matrix on that day and at unselected rooms in the available interval are reduced by 1.

With this algorithm, each operation is scheduled at the interval where there is a minimum available interval. With this heuristic total of 85 patient is scheduled, and 75 of the operations are scheduled to their initial available time interval.

Utilization of the rooms is not our priority in objective function value; however, in order to do as much operation as we can, we have also kept our utilization high. The values for the utilization can be seen in *Table.1.1.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  | **DAY 1** | **DAY 2** | **DAY 3** | **DAY 4** | **DAY 5** | **ROOM AVG** | | **ROOM 1** | 0.8750 | 0.8333 | 1.0000 | 0.8333 | 0.9167 | 0.8917 | | **ROOM 2** | 0.8750 | 0.8333 | 0.7083 | 0.7083 | 0.9167 | 0.8083 | | **ROOM 3** | 0.7500 | 0.8333 | 0.6667 | 0.5000 | 0.8333 | 0.7167 | | **DAY AVG** | 0.8333 | 0.8333 | 0.7917 | 0.6806 | 0.8889 | 0.8056 | |

Table.1.1 Utilization of the rooms each day in heuristic 1

On average, each room is utilized at least 70%. For each day, the average utilization is between 68% to 89%. Each day, utilization is maximized for room number 1 since each scheduling procedure starts from there. In the same manner, room three is utilized the least for five days since there is no patient left to fulfill this room.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **PRIORITY 1** | **PRIORITY 2** | **PRIORITY 3** | **PRIORITY 4** |
| **NUMBER OF OPERATIONS SHIFTED** | 20 | 18 | 10 | 6 |

Table.1.2 The number of operations shifted in each priority level

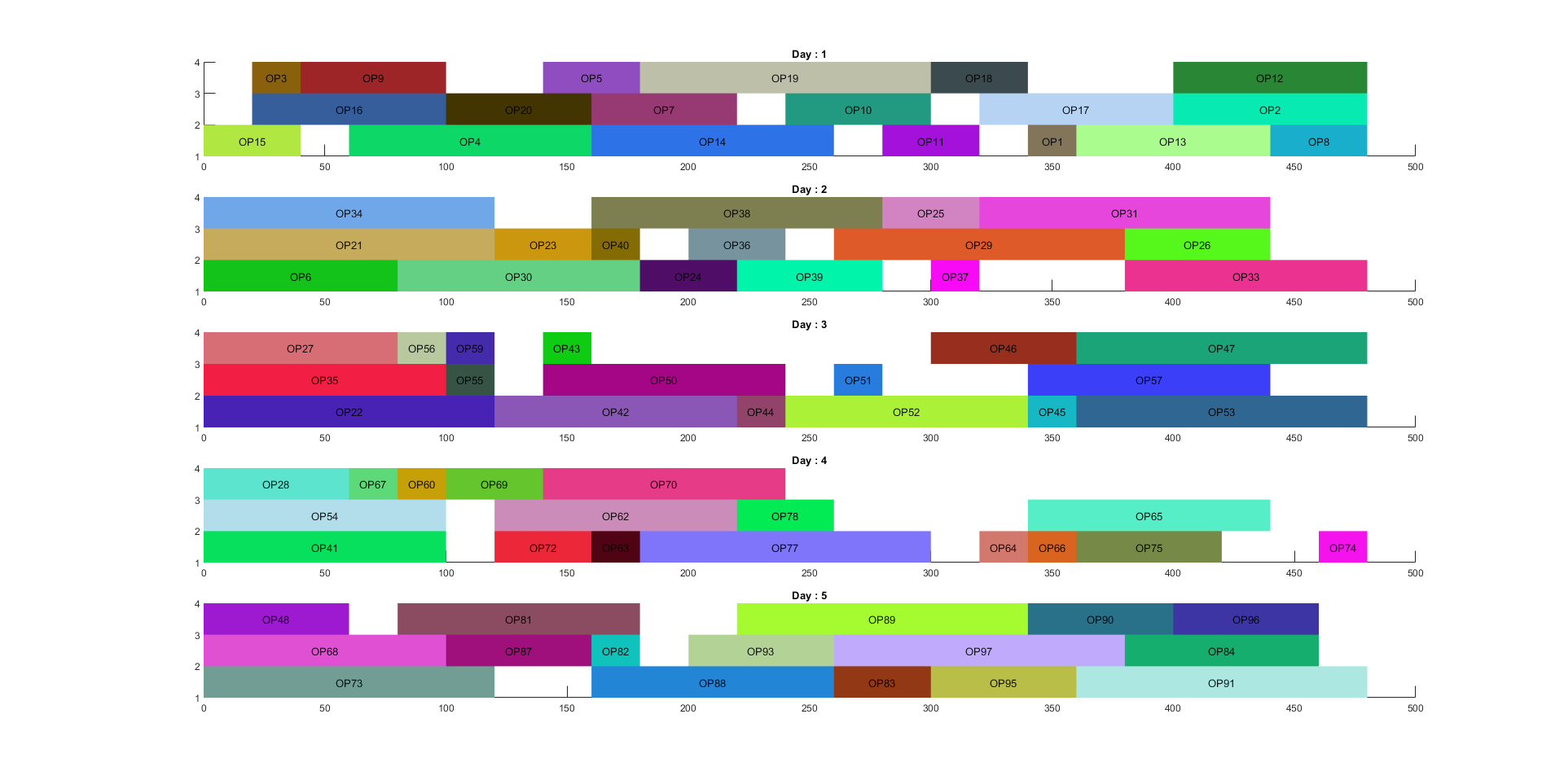
 For each priority, number of operations shifted is stated in *Table.1.2*. As can be seen on the table, as the priority gets higher, more of the patients are shifted. Since patients with higher priority are scheduled before the rest of the other patients, they are fitted into the least occupied interval. This enables other low-priority patients to fit in that spaces to maximize the number of patients scheduled.

Figure 1.2 Gantt Chart of Schedule Objective 1

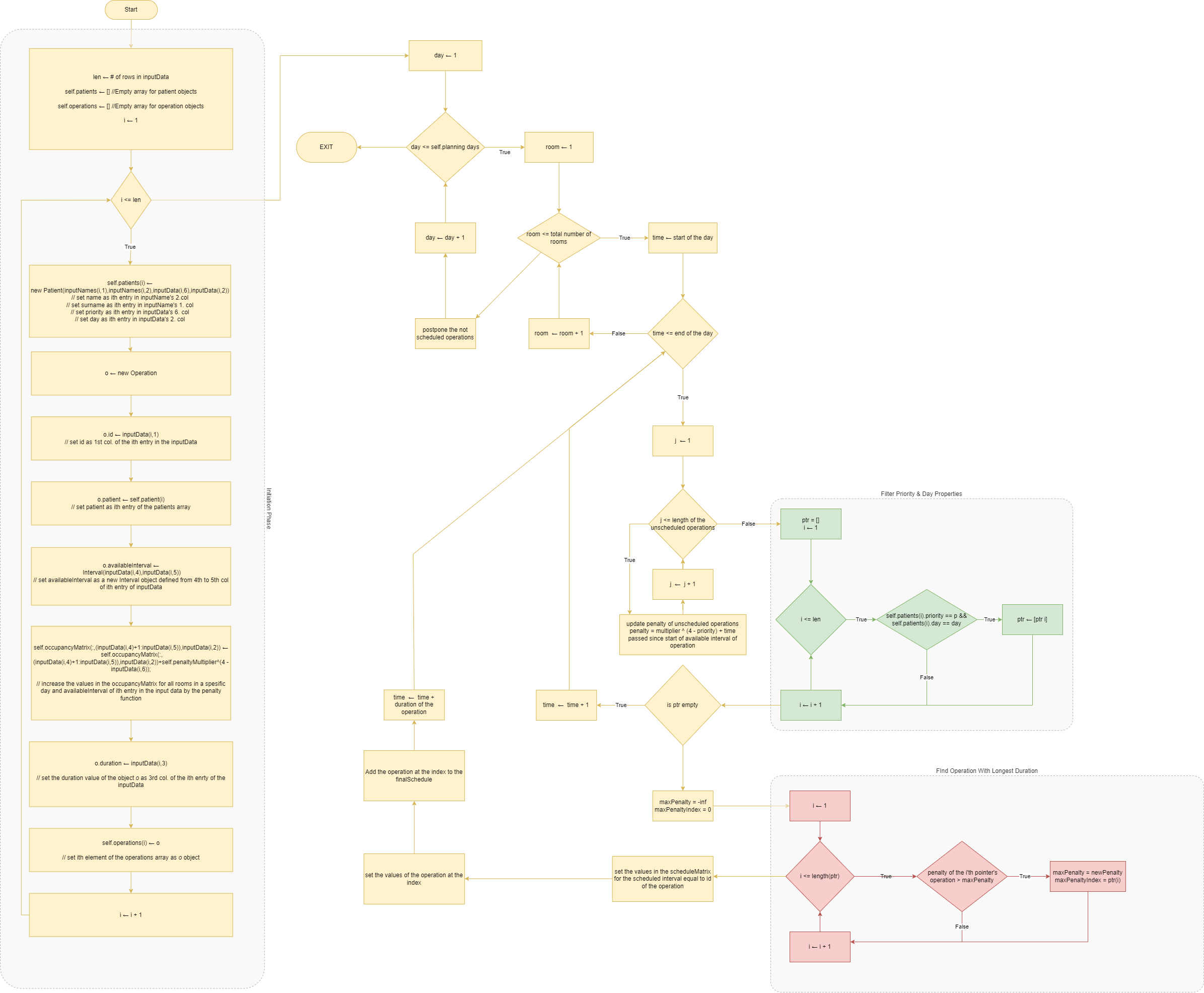
Objective 2

Figure 2.1 Flowchart of Schedule Objective 2

In this objective, our purpose is to maximize the utilization of the surgery rooms. In order to fulfill this objective, the designed algorithm starts scheduling from the start of the first minute of the first day. For time 0 and day 1, finds all of the operations that could start from this time and filter them out. Among filtered operations, the penalty of each of them is calculated, and operations are scheduled according to this penalty value in descending order. This penalty is equal to the equation bellow

With this penalty function, operations with higher priority will have the biggest penalty among all the operations. So, these operations will be scheduled first. Also, operations with the same priority value will be scheduled according to their delay so that patients will be shifted by a minimum amount. After filling the schedule in room 1, the rest of the operations are scheduled in room 2, and so on. After completing all of the rooms, the rest of them are postponed to the next day.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **DAY 1** | **DAY 2** | **DAY 3** | **DAY 4** | **DAY 5** | **ROOM AVG** |
| **ROOM 1** | 0.9583 | 0.9583 | 1.0000 | 0.9583 | 1.0000 | 0.9750 |
| **ROOM 2** | 0.9583 | 0.8750 | 0.7083 | 0.7917 | 0.9167 | 0.8500 |
| **ROOM 3** | 0.7083 | 0.6250 | 0.7500 | 0.5417 | 0.9583 | 0.7167 |
| **DAY AVG** | 0.8750 | 0.8194 | 0.8194 | 0.7639 | 0.9583 | 0.8472 |

**Table 2.1** Utilization of the rooms each day in heuristic 2

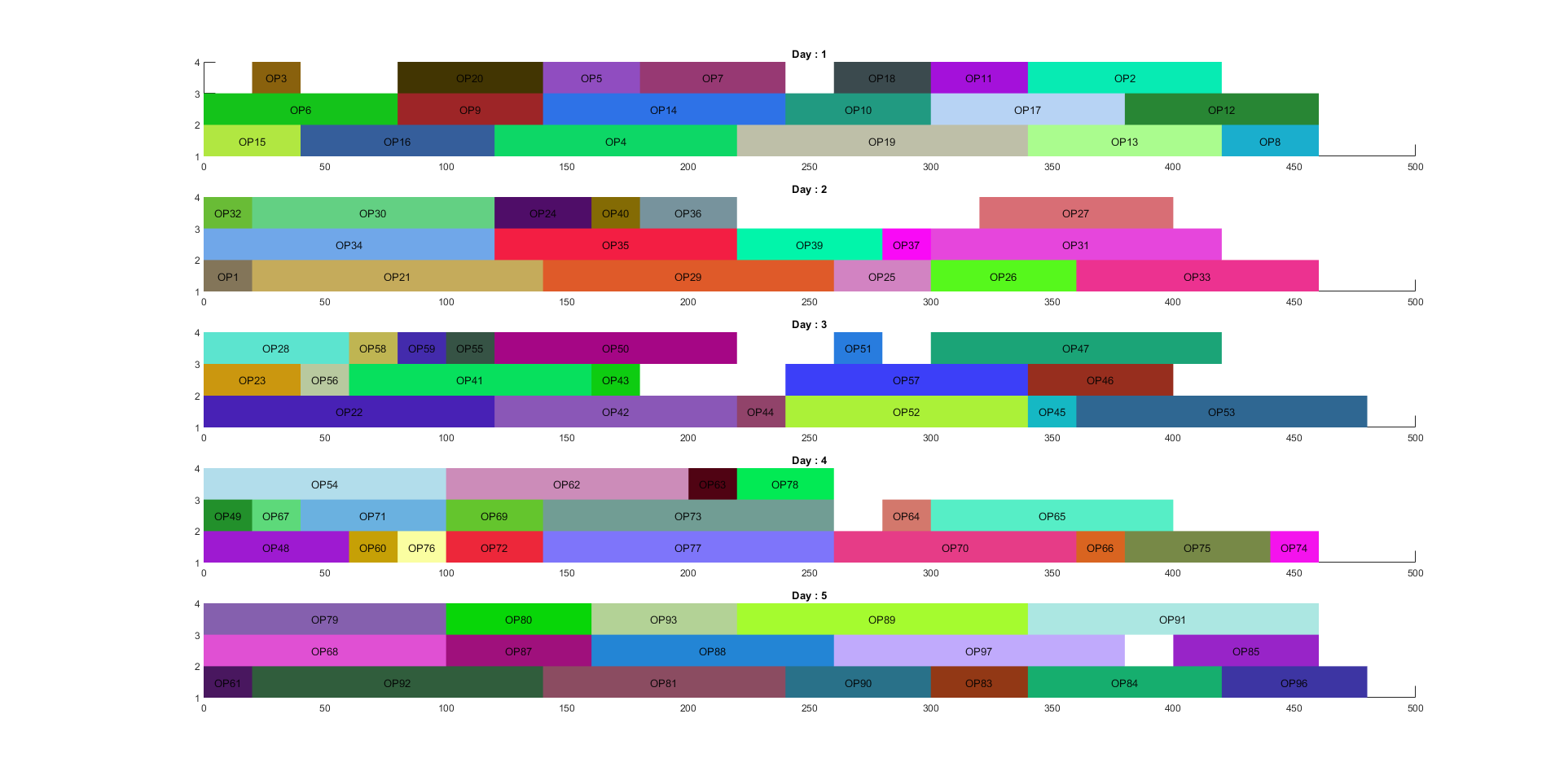
On average, each room is utilized by at least 72%. For each day, the average utilization is between 76% to 95%. Each day, utilization is maximized for room number 1 since each scheduling procedure starts from there. In the same manner, room three is utilized the least for five days since there is no patient left to fulfill this room. Compared with heuristic 1, the average utilization of rooms is increased significantly. The total average room utilization was 80.5% in heuristic 1, but with heuristic 2, this value is increased to 84.7%. This change corresponds to better usage of the surgery rooms and more efficient use of the hospital’s resources. With this algorithm, the number of patients postponed to the next day is also decreased.

A total of 14 patients are postponed to the next day, and the priority of those patients is not lower than 3. Due to the better usage of the time, fewer people are postponed, and more of the operations are done in their available intervals.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **PRIORITY 1** | **PRIORITY 2** | **PRIORITY 3** | **PRIORITY 4** |
| **NUMBER OF OPERATIONS SHIFTED** | 19 | 17 | 9 | 9 |

**Table.2.2** The number of operations shifted in each priority level

For each priority, the number of operations shifted is stated in *Table.2.2.* As can be seen on the table, as the priority gets higher, more of the patients are shifted. Since patients with higher priority have a bigger penalty, the algorithm must schedule them first. When they are getting scheduled, in order to prevent inefficient use of other rooms, the algorithm decides to schedule an operation that needs a more significant shift in its available interval. Therefore, operations with higher priority shifted more than operations with lower priority.

Figure 2.2 Gantt Chart of Schedule Objective 2

Futher Examination

Conclusion

In this project, we designed algorithms based on heuristics for different objective functions and used these algorithms to schedule operations in a three-room local hospital. Based on approaches and scheduling obtained in this project, decision-makers of the local hospital could make their decision regarding the use ofs resources.

One of the findings of this project is that……

………..