Ministry of Education and Science of the Republic of Kazakhstan
Suleyman Demirel University

UDC 50.49

On manuscript rights

Babur Rustauletov

The Automation of Course Scheduling in Higher Education Institutions

THESIS

Presented in Partial Fulfillment for the
Degree of Master of Science in Computing Systems and Software
(degree code: 6M070400)
Department of Computer Sciences
Faculty of Engineering and Natural Sciences

Supervisor: **Dr., Professor B. Aitchanov**

Kaskelen, 2020

Abstract

The formation and management of schedules is repetitive and troublesome for any class, university and organization. Limited resources in terms of people, time and locations. Institutions needs solutions that will meet all of these constraints of scheduling. The system for scheduling is proposed to take in control the scheduling process that takes time by automatically allocating time slots for teaching and creating course schedule. And easily can be integrated with their systems without need to do some extra work like adding data for scheduling or exporting them. One of the benefits of this system are reducing the time spend on creating schedules and minimizing errors that made by people, so it is expected that the schedules will be generated more quickly.

Аңдатпа

Кестені құру және басқару кез-келген сынып, университет және ұйым үшін қайталанатын және қиын. Адамдар, уақыт және орын тұрғысынан шектеулі ресурстар. Мекемелерге жоспарлаудың барлық шектеулеріне сәйкес келетін шешімдер қажет. Жоспарлау жүйесінде сабақ уақытының кестесін автоматты түрде бөлу және уақыт кестесін құру арқылы уақытты талап ететін кесте процесін бақылауға алу ұсынылады. Оларды жоспарлау немесе экспорттау үшін деректерді қосу сияқты қосымша жұмыстарды қажет етусіз олардың жүйелерімен оңай біріктіруге болады. Бұл жүйенің артықшылықтарының бірі - кесте құруға кететін уақытты азайту және адамдар жіберетін қателіктерді азайту, сондықтан кестелер тезірек құрылады деп күтілуде.

Аннотация

Формирование и управление расписаниями является повторяющимся и хлопотным для любого класса, университета и организации. Ограниченные ресурсы с точки зрения людей, времени и места. Учреждениям нужны решения, которые будут соответствовать всем этим ограничениям планирования. Система планирования предлагается взять под контроль процесс планирования, который занимает время, автоматически распределяя временные интервалы для обучения и создавая расписание курса. И легко может быть интегрирован с их системами без необходимости выполнять дополнительную работу, такую как добавление данных для планирования или их экспорт. Одним из преимуществ этой системы является сокращение времени, затрачиваемого на создание расписаний, и минимизация ошибок, допущенных людьми, поэтому ожидается, что расписания будут создаваться быстрее.

Acknowledgements

I would like to express my sincere gratitude to the Suleyman Demirel University for letting me fullfill my dream of being a student here. I would also like to thank the Computer Science Department for giving me the opportunity to write thesis. To committee, Cemil Turan, Andrey Bogdanchikov, I am extremely grateful for assistance and suggestions throughout my project. To the staff at the Automation Department, thank you for always listening and giving me words of encouragement. To all my friends and family for helping me survive all the stress from this year and not letting me give up. Most of all, I am fully indebted to Aitchanov Bekmurza, my advisor and to Baimuratov Olimzhon, my coach in research, for their understanding, wisdom, patience, enthusiasm, and encouragement and for pushing me farther than I thought I could go.

Contents

1 Introduction **7**

1.1 Motivation 7

1.2 Aims and Objectives 12

1.3 Thesis Outline 13

2 Analysis of the scheduling problem **14**

2.1 General Characteristics of the Task 14

2.2 Overview of Existing Solutions 18

2.3 Overview of Solution Methods 22

2.3.1 Mathematical Optimization 23

2.3.2 Combinatorial Approach 25

2.3.3 Heuristic and Probabilistic Methods 27

3 Formalization of the problem **30**

3.1 General Task of the Model 30

3.2 Creating Schedule Elements 31

4 Development of a method for solving the problem **35**

4.1 Sequential scheduling method 35

5 Development of Automated System **41**

5.1 Development of the Structure of an Automated System for Scheduling Courses 41

5.2 Hardware Components 42

5.3 Software Components 43

5.4 Development of the Logical Structure 44

5.5 Development of a Scheduling Algorithm 45

5.6 Development Web Interface 49

6 Analysis of results **54**

6.1 Results of Testing System 54

7 Conclusion **58**

1. Introduction

1.1 Motivation

In recent years, the automation process has taken wide steps. Medicine, agriculture, aircraft manufacturing, engineering and other areas of activity have moved to a new level thanks to her.

One of the most difficult and important with practical the point of view of the problems of system analysis is the planning of joint work of the elements of a complex system, taking into account their interconnection and various constraints due to it many elements of a complex system are found in many areas of traffic management, industrial planning enterprise or fulfillment of orders in the design organization, organization of work of institutions of the social-economic sphere, education and many others.

An analysis of the range of tasks of the class under consideration shows that, from the point of view of dimension, as well as the variety and severity of restrictions, the task of scheduling for higher education institutions, which is chosen to illustrate the main thesis provisions

In most educational institutions, you can see the use of information technology. One of the main details of the educational process in the educational institution is the course schedule, which are made in manual form for hundreds of years by the workers of the educational institution who spent days and even weeks on it. And only in the last decade we can see the use of some computer programs that are specially developed for scheduling. But why not in all institution we have such systems? What prevents institutions from using such programs? Which problems comes in using such system?

In the last 10-15 years, higher education in Kazakhstan has been characterized by a process of constant changes, and the most significant changes can be expected with the planned transition to a two-stage credit-modular system of higher education. Higher education institutions should be able to quickly and flexibly respond to possible changes.This opportunity is given by automation of planning the educational process of the university. With the growing diversity of educational programs, an acute shortage of budgetary funding and classrooms, the tasks of rational allocation of resources are becoming of paramount importance. Scheduling classes in new conditions without automation of this process is not possible [1]

An important problem is the staffing of the educational process. It is known that the insufficient level of remuneration of teachers in state educational institutions stimulates their desire to work part-time in other universities and enterprises.On the other hand, the modern scientific and technical level of educational programs is traditionally ensured by involving specialists from scientific and industrial organizations in the educational process. As a result, the vast majority of teachers require an individual work schedule.So in SDU at the moment more than one-third teachers submit individual wishes for the schedule.

A growing variety of forms of training, with different requirements for restrictions on the duration of classes, both for training forms and for groups.

Recently, various forms of additional education have been actively developing, requiring a special training schedule for groups, teachers and occupying the resources of the classroom fund.

When working with the classroom fund, it is necessary to take into account the rules of work of individual classrooms, the capacity of classrooms, the availability of specific equipment, the quality of the preparedness of the classrooms to ensure the educational process, the audience belonging to a university or department classroom fund, and the division of classrooms into ordinary classrooms and laboratories.

The presence of a large number of educational programs and curricula on the one hand and compliance with the requirements of unification of the educational process on the other hand leads to the need for flexible formation of flows of study groups for conducting various disciplines, with the union of groups of various specialties, courses and forms of training, which greatly complicates the process of scheduling.

The formulated problems, as well as the presence of a large number of diverse specific requirements for the organization of the educational process imposed by the administration of the educational institution and the graduating departments, make it impossible to apply old approaches and methods to the scheduling process based on manual labor. A huge variety of curricula and requirements, as well as a significantly increased volume of tasks and limited volume classroom fund, make the task of scheduling too complicated and inaccessible to the team of dispatchers. Due to the increase in diversity and the increased connectedness of the schedule of groups of various specialties and forms of training, an increase in the number of dispatcher employees will not bring results, since all dispatchers need to coordinate actions among themselves in order to avoid mistakes and overlays. A person is no longer able to hold in his head a huge volume of information and perform many routine operations and checks. In such a situation, only an automated system is able to provide timely and flexible planning of the educational process, performing centralized work with all the data, performing a huge number of labor-intensive routine operations and providing the user with only services of the highest level of abstraction. The task of automating the scheduling of classes in a modern university is more relevant than ever.

Similar tasks are found in the transport schedule, traffic flow control, in the production and assembly of various products, the tasks of organizing work at enterprises.

From a mathematical point of view, the task of scheduling is an ordering problem and is characterized by a very high dimension. Mathematical methods for solving such problems are considered in the framework of the Theory Timetables (TT) Or Job Shop Scheduling [2].

The "temporary" nature of the problems of scheduling theory distinguishes them in a special class, which differs significantly from the "voluminous" economic problems. If in the latter it is required to answer the question of what and how much to produce, then in the tasks of the TT it is necessary to determine when and in what sequence to perform work. This difference in the essence of the problems determines the difference in methods and possibilities for solving them. For tasks of a voluminous nature, a sufficiently powerful apparatus has been developed, mainly mathematical programming, which makes it possible to successfully solve them. For TR problems, the decisive apparatus is developed to a much lesser extent [3]. The search for the optimal or near optimal schedule is carried out using one of 3 approaches:

- mathematical programming [4];
- combinatorial [5];
- heuristic [6];

The task of scheduling can be formulated in terms of linear integer programming. A system of operations consisting of operations and machines performing specific operations is described. In the form of inequalities, a system of constraints is presented. Additional integer variables are also introduced to describe "or-or" constraints that cannot be described as part of conventional linear programming. Next, the system itself is recorded and the objective function is formed. Target functions may vary. The specific form of the objective function depends on what we need to minimize [7].

When applying mathematical programming methods to solve the TR problem under consideration, a significant increase in the solution time is inevitable tasks depending on its dimension due to the inevitable use of methods of enumerating options:

$$T = C \times k^{mn}$$

where C is a certain constant, k is the number of pairs per week, m is the average number of types of classes conducted by the teacher, n is the number of teachers.

The most common methods of reducing enumeration are those based on the branch-and-bound method or the implicit enumeration method, which consist of constructing "partial solutions" that allow cutting off unpromising solutions. However, even the perfect methods of reducing the search do not allow to escape from the exponential growth of labor intensity.

The combinatorial approach is reduced to a purposeful permutation of pairs of jobs in a certain initial sequence until an optimal (close to optimal) solution. The combinatorial approach uses concepts such as class P problems, efficient algorithms, and NP-complete problems [8].

By "effective algorithm" is meant an algorithm for which the number of required steps grows as a polynomial in the size of the input problem. Tasks having effective (polynomial) solution algorithms belong to the class of P-problems.

The class of NP-tasks has the following properties:

- no NP-complete problem can be solved by any known polynomial algorithms;
- if there is a polynomial algorithm for some NP-complete problem, then there are polynomial algorithms for all NP-complete problem.

The practical meaning of the concept of NP-completeness is as follows: such problems are essentially difficult to solve from a computational point of view, they are not amenable to an efficient algorithmic solution, and for an algorithm that correctly solves NP-complete problem, an exponential amount of time is required in the worst case, and therefore it will not be applicable in practice to any, except for very small, tasks [9].

The unsatisfactory state of development of exact methods for solving NP problems has led to the development of approximate methods that allow obtaining acceptable solutions at a relatively low cost of time and money. Conditionally approximate methods are divided into heuristic and probabilistic [9].

Heuristic algorithms are based on a technique called a technique for reducing requirements. It consists in refusing to search for the optimal solution in an acceptable time. Heuristic algorithms use various reasonable images without rigorous justification.

The so-called local search method is widely used. Moreover, a pre-selected set of permutations is used to consistently improve the initial solution until such an improvement is possible, otherwise a local extremum is reached.

Another area of heuristic methods for solving TT problems is the formation of rules or preference functions (priorities). For each And i job from the set of work awaiting completion, the value of the preference function f_i is calculated and that work is selected for which f_i reaches a maximum or a minimum.

The task of scheduling can be presented in the form of an ordering problem, where the disciplines from the curriculum are operations, and the groups, teachers and classrooms are service devices. In this case, it is convenient to represent the model in the form of a network whose arcs represent operations. Each arc corresponds to a certain weight coefficient, which is the value of the objective function that determines the degree of preference for performing this operation. The vertices of the network, called events and denoted by E_i, can be interpreted as the results of individual private work. Thus, the task of scheduling is reduced to building a network taking into account all the required conditions. In the case where several networks can be built, it is necessary to calculate the critical path for each of them, and choose the network with the smallest critical path. This method is well suited for the classical task of streamlining, and to solve the task of compiling a training schedule, its application is problematic, since the training schedule does not just need to streamline the operations, but also select the time of their implementation, which is not provided for by this model and will lead to the need to build discrete level network with forbidden states.

When applying all these methods, an exponentially increase in the program run time with an increase in the task volume is also inevitable.

Thus, the known methods do not provide a solution to the problem of automated scheduling of classes, taking into account its characteristic limitations with acceptable labor intensity. The solution requires a search for a new approach, which defines the purpose and objectives of the dissertation research.

1.2 Aims and Objectives

The aim of the work is to study the features of automatic scheduling in Higher Institutions and designing software (information system) for automatic scheduling.

To achieve this goal, it is necessary to solve the following subtasks:

- Examine the criteria for automatic scheduling and existing programs in this area. The result of this subtask will be a list of requirements for automatic scheduling and programs in this area, as well as a list of popular programs and wishes for the functionality and capabilities of the software obtained by conducting an online survey among administrations of educational institutions.

- Analyze automatic scheduling programs. Initial information for solving this problem are screenshots of these programs and information from official websites of developers. The result is a list of the pros and cons of existing programs.

- To analyze existing algorithms for automatic scheduling. The result of this task is a selected package of algorithms for automatic scheduling in educational institutions.

- Develop a project for an automatic scheduling system. The initial information for solving this problem is the results of all the above tasks. The result of this work is a draft program with the described database structure, interface and scheduling algorithm.

1.3 Thesis Outline

The first chapter is Introduction chapter. It is this one that you are currently reading. It gives insight into the work done. In Chapter 1 we review and analysis of existing programs in area of automated course scheduling . Chapter 2 is describing about automatic scheduling algorithms and which is will be good to use in the system. Chapter 3 is about development of a method for solving the problem. Chapter 4 is searching for best technologies to use in development such systems. Last Chapter 5 is about Analysis of results. And in Conclusion chapter is conclusion of our work done.

2. Analysis of the scheduling problem

2.1 General Characteristics of the Task

The task of scheduling theory is considered given if defined [10]:

1. requiring work and operations;
2. criteria for evaluating the result.

Depending on the nature of the receipt of work, two types of tasks are distinguished: static and dynamic [11]. In static problems, a certain number of jobs simultaneously enter a free system. After this, new works are not received and the schedule is drawn up for a well-defined and well-known number of works. In dynamic problems, the work enters the system at some points in time, which can only be predicted in a statistical sense. Therefore, the points of future earnings are not defined. Streamlining in dynamic and static problems requires different solution methods [3].

Depending on the order in which the machines perform operations that are part of the work, they distinguish between machine systems [12].

1. conveyor;
2. with a random order of work;
3. arbitrary type;

The task of scheduling occurs in a wide variety of areas: in the transport schedule, traffic management, production and assembly of various products, organization of work at enterprises.

The task of scheduling classes is characterized by discrete start and end times and requires several machines of different types to process the operation.

The task of compiling a schedule of classes at a university is characterized by high, laboriousness and has a number of features; hindering its formalization and automated solution. A variety of forms of training and curricula; on the one hand, a limited number of classrooms and restrictions associated with the mode of work of individual teachers, on the other hand, make it necessary to take into account a large number of requirements, sometimes mutually exclusive. The main difficulty in scheduling is to take into account the totality of the requirements.

Modern practice shows that the task of scheduling in the statement considered above is solved "manually" - only automation of work with documents is provided. Taking into account various individual requirements is the prerogative of experienced dispatchers using intuition and professional techniques worked out over many years. Such a process takes considerable time, and as the volume and variety of requirements increase, it becomes less and less successful. So in the SDU, by the time automation work began in 2019, the duration of the scheduling process for a semester approached three months.

The analysis of the problem under consideration by the example allows us to offer the following classification of the basic requirements for the lesson schedule.

1. Group related requirements:

    (a) taking into account working curriculums of groups;

    (b) the presence of several forms of training with characteristic time restrictions for classes: full-time, part-time (evening), etc .;

    (c) association of groups in training flows for the lesson (one group may be included in various flows);

    (d) accounting of the number of groups and educational flows;

    (e) the ability to select the time of classes on the days of the week;

    (f) inadmissibility of "windows" in the group schedule;

    (g) at least three lessons per day;

    (h) no more than four lessons per day (in exceptional situations, five lessons);

   (i) uniform distribution of lectures, practical and laboratory classes by day of the week;
   (j) the availability of free day at senior courses;
   (k) the ability to conduct several classes on the same course in a row;

2. Requirements related to teachers:
   (a) the appointment of teachers for specific classes;
   (b) the workload of the teacher;
   (c) the possibility of excluding simultaneous classes for some teachers;
   (d) taking into account the individual wishes of teachers on the day of the week and the time of the classes;
   (e) taking into account the maximum number of busy days for a teacher;
   (f) taking into account the maximum number of lessons per day for the teacher;
   (g) taking into account the maximum number of lectures per day for the teacher;
   (h) taking into account the maximum number of practical lessons per day for the teacher;
   (i) taking into account the maximum number of laboratory lessons per day for the teacher;
   (j) providing a compact timetable for teachers;
   (k) ensuring an even distribution of classes by day of the week;
   (l) consideration of wishes for the building and floors.

3. Rooms requirements:
   (a) limited number of rooms;
   (b) taking into account for the capacity of rooms;
   (c) taking into account the specialization of rooms;
   (d) taking into account the rooms study-time.

(e) taking into account the affiliation of rooms (departments, university and cathedral-university)

(f) the possibility of eliminating the simultaneous conduct of classes in some classrooms.

4. Requirements related to the types of sections:

   (a) the presence of lectures, practical and laboratory classes;

   (b) division of groups into subgroups for laboratory work;

   (c) compliance with the sequence of some classes in the schedule.

It should be noted that the listed requirements are very diverse, some of them are difficult to formalize. Practice shows that the main reasons that complicate the scheduling of classes are the lack of classrooms and the large number of diverse wishes of teachers.

We formulate the statement of the applied problem.

An automated system for scheduling classes should, at an acceptable time without the participation of the operator (automatically) or with minimal participation (in automated mode), arrange all classes in the schedule grid and assign classrooms for them from a limited list in accordance with the requirements listed above.

It should be noted that in some cases for universities the question of a limited number of classrooms arises most sharply. To compile a convenient schedule, both the number of general audiences and individual laboratories are insufficient.Therefore, the process of scheduling can be understood to some extent as a question of the efficient allocation of shared resources. There should be no overlap on the simultaneous ownership of shared resources. The lack of shared resources leads to a significant loss in the quality of the schedule, the need to abandon certain requirements, or even the inability to schedule under given conditions.

It is necessary to develop a model to take into account all the necessary requirements.

2.2 Overview of Existing Solutions

To date, a number of software products are known that make it possible to schedule classes, taking into account a certain set of requirements. Most of these software products are focused on their use in schools. The school schedule is much simpler than in universities, and the dimension of the task is much smaller.

All considered software products for a set of functionality can be divided into the following groups:

- scheduling systems for secondary education institutions,
- scheduling systems of medium functionality at a university,
- scheduling systems of high functionality at the university.

Schedule systems for secondary education

The software products of this group are mainly designed for scheduling classes in schools and technical schools.

The following programs belong to this group:

- Timetable PRO [13]
- Express TimeTable [14]
- Nika software [15]

Some programs of this group allow you to create a schedule for training in two shifts, to divide groups into subgroups. School orientation makes unnecessary such important requirements for universities as taking into account the forms of group learning, scheduling a two-week training cycle. Most programs support manual and automated modes. For programs of this group the assignment of teachers to classrooms is characteristic.

The programs of this group are designed to solve the problem of small volume; they do not satisfy the following requirements:

- taking into account the wishes of teachers,
- taking into account restrictions on the time of classes for groups and forms of training,

- taking into account the mode of work of audiences,
- grouping groups into classes
- division of groups into subgroups for laboratory work,
- various options for ordering classrooms for classes.

The programs of this group as a whole make it possible to automate the scheduling process in schools and technical schools, but are not suitable for universities because of the small dimension of the problem being solved and the lack of taking into account a large number of university-specific requirements.

Schedule systems of medium functionality at a university

The software products of this group have more functionality than the programs of the previous group and allow you to schedule more.

The following programs can be attributed to this group:

- Rector 3 [15]
- Cyber Matrix Class Scheduler [16]

This group of programs is positioned by developers as programs for scheduling at a university. However, individual programs give the impression of incompleteness and ill-conceived ideology, a very weak acquaintance of developers with the specifics of the educational process in a higher educational institution.

Key features of this group of programs:

- support the operating mode of the university,
- support two-shift training,
- the ability to reserve audiences for specific academic disciplines,
- support a two-week schedule,
- take into account the transition time between audiences.
- support automatic and manual modes,
- support wishes for the duration of classes for teachers,

- the ability to set the mode of operation of audiences,
- support audience types.

Some pros:

- there is no concept of a study group with an individual curriculum,
- there is no concept of forms of learning,
- there is no possibility to set restrictions on the time of classes for groups and forms of training,
- there is no possibility of flexible adjustment of the educational institution's operating mode,
- do not support the ability to indicate the affiliation of classrooms to departments,
- does not support the possibility of conducting one lesson by several teachers,
- do not support types of classes (lectures, practices, laboratory),
- do not allow in various combinations to reserve audiences for specific occupation.
- lack of transparent logic for working with the system, inconvenient user interface.

Despite the identified shortcomings, it should be noted a fairly wide range of functionalities of the program "Rector". Attempts have been made to take into account a number of requirements specific to universities (two-week schedule, types of classrooms, etc.). However, these programs do not allow solving the problem in a complete statement, since they do not take into account some very essential requirements. The ideology of the programs does not allow them to be used for scheduling in large educational institutions.

High Functionality Scheduling Systems at a University This group includes the most powerful scheduling programs with the widest range of functionality.

The following programs can be attributed to this group:

- 1C University [17]
- AscTimeTables [18]
- FET [19]
- UniTime [20]

Capabilities of this group:

- allows you to conveniently enter and edit curricula,
- supports various types of activities,
- allows you to store a wide range of information about the teacher,
- the teacher can be assigned disciplines that he can teach and audiences in which he can teach,
- it is possible to set the mode of work of the teacher,
- it is possible to set the mode of operation of the educational institution.
- taking into account the required range of days / hours for groups, teachers and classrooms;
- taking into account the wishes of teachers;
- adjustment of the class schedule.

Identified deficiencies

- a teacher can only teach in one department,
- creating flows
- there is no form of training for the group,
- there is no possibility to set time limits for conducting classes for a specific group.
- there is no way to record the number of busy days for groups and teachers;

- it doesn't allow you to reserve audiences for various classes in various combinations.
- ability to integrate programs with ERP modules.

The main module "Schedule" - is designed for semi-automatic scheduling of students. Partial automation consists in the fact that the schedule is not compiled automatically, and the user is given the opportunity to choose one of the available teachers, disciplines and classrooms for each cell of the schedule.

Unfortunately, there is no data on the principle of constructing the basic algorithms and, accordingly, the dependence of the solution time on the volume of the problem. Analysis of the available data allows us to conclude about the use, as in most other specified software tools, of enumeration of options and, therefore, about the nonlinear dependence of the solution time on the volume of the problem.

The software products of this group can be used to compile fairly complex timetables at universities, for the educational process of which the identified shortcomings are not critical. The question remains about the time of scheduling for a fairly large educational institution.

2.3 Overview of Solution Methods

The lack of automated systems on the market that can solve the problem in a complete statement forces us to develop a new information system that can schedule classes based on the full system of restrictions.

Let us analyze the well-known methods traditionally applied to scheduling problems.

The task of scheduling classes belongs to the class of tasks of the theory of schedules (ToS), which is a section of the study of operations [21]. The formation of ToS as an independent direction of applied mathematics took place at a time when the linear programming apparatus was sufficiently developed, which naturally found the application of this apparatus to solve problems scheduling. Use of linear models in real situations is possible only in a few cases. In this case, the need arises for solving large-scale linear programming problems [22].

In some cases, optimal schedules can be compiled as a result of simple considerations regarding changes in the characteristics of the schedule during some of its elementary transformations. A combination of techniques of this kind forms the basis of combinatorial methods of scheduling theory [9].

Among the approximate methods of scheduling theory, an extensive group of heuristic methods should be noted. In these methods, to some extent, the experience gained as a result of multiple scheduling in practically the same conditions is reflected. The search for the best schedule is artificially narrowed, often to one specific schedule.

The search for the optimal or near optimal schedule is carried out using one of 3 approaches[3]:

- mathematical optimization;
- combinatorial;
- heuristic and probabilistic.

2.3.1 Mathematical Optimization

The basics of scheduling theory developed at a time when mathematical models began to be applied to solve economic problems. Attempts were made to build mathematical models for the problems of ToS. At the same time faced difficulty of the following kind. In the mathematical model, the system of restrictions reflects the state of things that a certain set of conditions must be fulfilled together. In the ToS problem, a number of conditions must be fulfilled alternatively: either the i-th work starts earlier than the y-th one, or vice versa.

Let us consider the formulation of the general problem of scheduling in terms of linear programming.

Suppose we have a system of n jobs and m machines. Each work consists of g. operations. Each operation has three indexes: i is number of work containing this operation; y is the operation number inside the work, i = 1, ..., g. ', the k-number of the machine on which the operation is to be performed.

The restrictions on the time and order of operations by machines are as follows:

- each machine performs at the same time no more than one operation;

- operations are performed in the specified sequence;
- no two operations related to the same work are performed simultaneously.

Classical ToS considers sequential processing of an operation by machines [88]. In the training schedule for the operation, several machines are required simultaneously (groups, teachers and classrooms, are like machines, and training sessions are like operations). This fact does not allow us to attribute the considered problem to the class of simple problems of scheduling theory and to use the methods developed at present. If the simultaneous execution of an operation by several machines is permissible.

As you can see, taking into account the need for simultaneous operation of several machines leads to nonlinear constraints and does not allow the use of linear programming methods. The task of creating a curriculum is discrete, since classes can only be scheduled at strictly specified time intervals.

In addition, curriculum development is a multi-criteria task. For example: minimizing the number of windows in the schedule of groups and teachers, minimizing the number of transitions from the building to the building for groups and teachers, maximum compliance with the informal wishes of teachers, optimization (using various criteria) of using the classroom fund. The combination of all these criteria into one complex criterion is unacceptable [96]. For an objective analysis of the results obtained, it is necessary to consider several criteria, which makes it impossible to directly use mathematical programming methods.

When applying the methods of mathematical programming for solving TP problems, an inevitable exponential increase in the time of solving the problem depending on the dimension of the problem [23].

The most widely used techniques for reducing enumeration include those based on the branch and bound method or the implicit enumeration method. These techniques consist in constructing "partial solutions", presented in the form tree search and application of methods for constructing estimates, allowing to cut off unpromising partial solutions. However, even perfect methods of reducing enumeration do not allow one to escape from the exponential growth of labor.

2.3.2 Combinatorial Approach

The combinatorial approach is reduced to a purposeful permutation of pairs of jobs in a certain initial sequence until an optimal (often close to optimal) solution [9].

The combinatorial approach uses concepts such as class P problems, efficient algorithms, and NP-complete problems [8].

By "effective algorithm" is meant an algorithm for which the number of required steps grows as a polynomial in the size of the input problem. Problems with efficient (polynomial) solution algorithms belong to the class of P-problems.

The class of NP-tasks has the following properties:

- no NP-complete problem can be solved by any known polynomial algorithms;
- if there is a polynomial algorithm for some NP-complete problem, then there are polynomial algorithms for all NP-complete problems.
-

The practical meaning of the concept of NP-completeness is as follows: such problems are essentially difficult to solve from a computational point of view, they do not lend themselves to an efficient algorithmic solution, and for an algorithm, I correctly solve If an NP-complete problem is solved, an exponential amount of time will be required in the worst case and, therefore, it will not be applicable in practice to any, except for very small, problems [82].

Johnson's problem (polynomially solvable problem) [24] [25]. We have a conveyor of two machines: m = 2. Each work consists of two operations with durations a_i and b_i the total time for servicing the work is minimized.

The sequence that minimizes the total operating time is as follows: first, jobs are started for which $a_i <= b_i$ in non-decreasing order a_i, then all jobs for which $a_i > b_i$ are in non-increasing order b_i (thereby minimizing the downtime of the 2nd machine due to that the first has not yet managed to process any work). The optimality of such a sequence is proved. However, the results do not apply to the case n > 2.

The NP completeness of the problem is a weighty argument in justifying the need to build approximate or heuristic algorithms for solving it, using schemes of directional enumeration of options (such as the method of sequential construction, analysis and screening of options (generalization of the branch and bound method)), and also when substantiating the need for research special cases of the problem.

We will evaluate the complexity of compiling the curriculum. Suppose that there is a list of n training sessions that need to be scheduled. We will take classes from the list in the order of their placement, and put each lesson in the best place for him. The complexity in this case will be approximately as follows:

$$T \cong n \times C$$

where C is the time required to find the best place to practice.

The above formula does not take into account all possible options for the schedule for a given sequence of classes, as it does not take into account all the options for arranging classes (the lesson is put in the best place).

Note that the above formula gives an underestimated estimate of the complexity, as it does not take into account all the options for arranging classes, but reflects only options for the best placement of classes.

The schedule of training sessions depends significantly on the order of its preparation, as previously assigned classes limit the choice of time for the remaining ones. Thus, with a different sequence of classes in the list, various schedule options are obtained. Given the various options for following the classes in the list, we obtain the following approximate estimate of the complexity:

$$T \cong C \times n \times n!$$

The above formula does not take into account all possible options, limiting itself to arranging classes at the most preferred time intervals. It already contains the truncation of search options.

If, by some heuristic method, it is possible to optimally organize classes in the list, then it will be possible to reduce the complexity of the problem in n! time. The assessment of the complexity in this case will look as follows:

$$T \cong n \times C$$

In this case, the search remains in the task of choosing the best time for a particular lesson.

2.3.3 Heuristic and Probabilistic Methods

The tasks of the scheduling theory as a whole and the task of compiling the curriculum in particular belong to the class of NP-complete problems and cannot be solved using exact algorithms except for problems of very small volume [26].

The unsatisfactory state of development of exact methods for solving NP problems has led to the development of approximate methods that allow obtaining acceptable solutions at a relatively small cost of time and money. Conditionally approximate methods are divided into *heuristic* and *probabilistic*.

Heuristic algorithms are based on a technique called *a technique for reducing requirements*. It consists in refusing to search for the optimal solution in an acceptable time. Heuristic algorithms use various reasonable considerations without rigorous justification.

The so-called local search method is widely used. Moreover, a pre-selected set of permutations is used to consistently improve the initial solution until such an improvement is possible, in In the real case, a local optimum is reached.

Another area of heuristic methods for solving NP problems is the formation of rules or preference functions (priorities). For each i-th job from the set of jobs awaiting completion, the value of the f_i preference function is calculated and that job is selected for which f_i reaches a maximum or a minimum.

The advantage of heuristic methods is the convenience of implementing them on a computer, even when solving cumbersome tasks.

The disadvantages of heuristic methods are the difficulty in assessing the proximity of the obtained schedules to the optimal one. In addition, for each preference function, there are tasks for which the use of this function leads to poor results. One way to improve the method of preference functions is to bind them to task classes.

Probabilistic methods are associated with k-time modeling of schedules. The selection of jobs from the set of pending execution is carried out randomly. After playing back, the best schedule is selected, which is taken as the solution to the problem. At the same time, they distinguish:

1. non-directional random search;

2. directed random search without self-learning;

3. self-directed guided random search.

The heuristic method, traditionally used by the human controller, is as follows. From experience it is known that the schedule must begin to be drawn up with the most loaded classrooms and teachers. Wherein, making a schedule, the dispatcher at any stage can make changes to the process based on personal experience and intuition. Direct simulation of a dispatcher's machine is complex and ineffective. Many dispatcher actions are difficult to formalize and difficult for machine implementation. Thus, the dispatcher begins to schedule the classes that are the most difficult to set up. It is advisable to organize classes in accordance with a certain coefficient that evaluates the complexity of setting a lesson, we call its complexity coefficient (CC).

The heuristic method can be used to determine the optimal sequence of classes in the list for placement, which will allow to avoid an exponential increase in labor intensity from the dimension of the problem inherent in the class of NP-complete problems. In the process of scheduling, the allowable time for operations for individual specialized machines changes, the workload of universal machines grows, which ultimately can affect the order of execution of the remaining operations. Since previously assigned operations limit the options for choosing the time for the remaining ones, you should arrange operations as the number of classes that affect the arrangement of a given one decreases. To smooth the subjectivity of the heuristic method of forming the optimal sequence of arrangement of classes, it is advisable to make a correction the sequence of operations in the scheduling process.

As conclusion for this section.

1. An analysis of the well-known methods of scheduling theory showed the absence of a universal method for solving the scheduling problem with processing an operation by several machines that have an individual operating mode. The high dimensionality of real scheduling problems does not allow the practical application of existing methods.

2. It is advisable to look for a new empirical method that provides for a partial application of the combinatorial approach with the rejection of a complete enumeration of options by streamlining the list of operations.

3. As a specific applied problem to illustrate the main provisions of the dissertation, the task of compiling a class schedule for a higher educational institution is selected, which is of the greatest interest both in terms of complexity and taking into account practical needs.

4. The analysis of the automated systems for scheduling classes existing on the software market, made in this section, showed that they do not provide a solution to this problem, taking into account the complete system restrictions typical of a modern institution of higher education. Therefore, it is relevant to develop a methodology for scheduling classes and an automated system that implements this technique.

3. Formalization of the problem

This section presents the formalization of the task of the theory of schedules, selected as the main one for the dissertation research. To take into account the practical requirements and limitations characteristic of the problem under consideration, we used the apparatus of set theory. This approach is convenient and effective for constructing both a mathematical model and algorithms for solving the problem.

3.1 General Task of the Model

Consider an average university with 10,000 classes, 1,000 teachers, 500 groups, 300 classrooms, the study cycle consists of two weeks (78 time intervals). Let us estimate the dimension of the system.

The number of variables in the group (2.1) is 780,000, the groups (2.2) 3,000,000, the groups (2.3) - 1,170,000,000, the groups (2.4) - 234,000,000. The number of restrictions (2.8) - 10,000, (2.9) - 117 000, (2.10) - 23 400. We see that the task has a very high dimension, the number of restrictions is measured in hundreds of thousands. In practice, not one of the known methods is able to cope with such a high dimension, not even speaking about the criteria of optimality.

It is necessary to look for a heuristic method based on the principle of ordering operations and truncating search options by finding the best time to complete the operation. Let's move on to the development of the required method based on the chosen practical task - scheduling training sessions at the university.

3.2 Creating Schedule Elements

It is convenient to present the whole set of possible schedules in the form of a five-dimensional space, the dimensions of which are discrete (Figure 3.2). The first dimension is teachers, the second is academic lessons, the third is study groups, the fourth is time, and the fifth is classrooms.

All dimensions are divided into segments, each of which is associated with an element of the set. The "teachers" dimension corresponds to many teachers. The dimension of "lessons" corresponds to many academic disciplines. The dimension of a "group" corresponds to many groups. The "time" dimension corresponds to the set of all pairs per week. The "rooms" dimension corresponds to many audiences. Part of this five-dimensional space is the schedule cell. Thus, choosing the elements of this space according to the required rules, we will have a ready schedule.

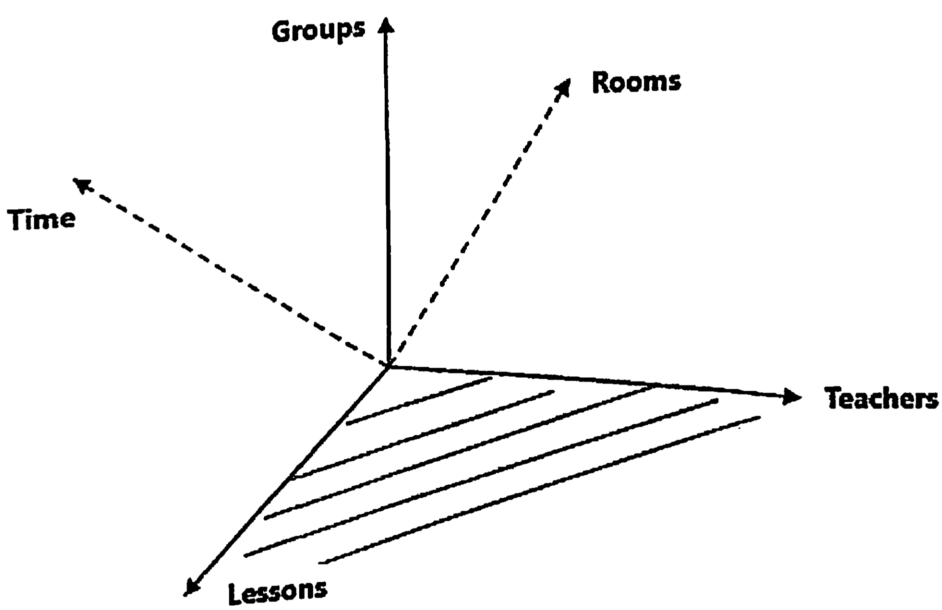

Figure 3.1: Schedule Space

This five-dimensional space can be decomposed into ten two-dimensional bases. By filling each of them with the observance of the condition of not contradicting each other, it will be possible to unambiguously restore the required space. This space will be the desired schedule. The first three dimensions are static. They do not change during the scheduling process. The three-dimensional space obtained on the basis of these measurements is the result of processing the source data. In fact, these are the restrictions that were set from the very beginning.

The hatched plane Teachers and Lessons reflects the possibility of conducting classes by the teacher in specific disciplines.

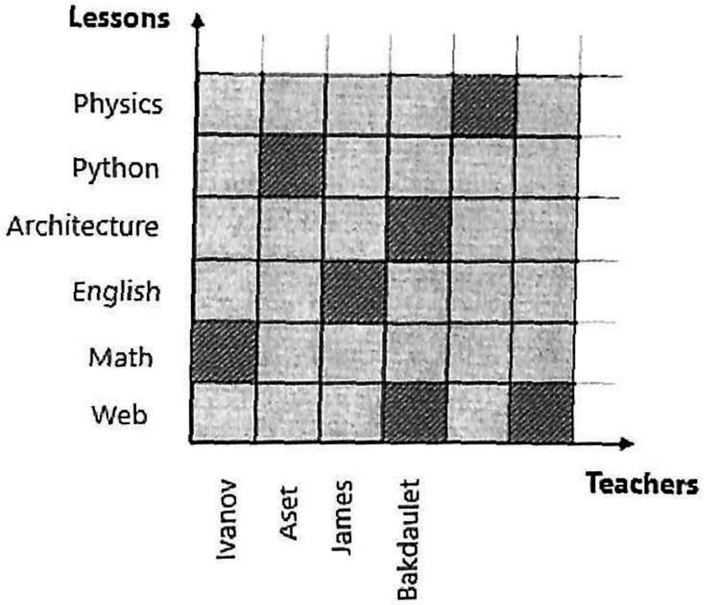

Figure 3.2: 2D Lessons - Teachers

Shaded boxes indicate acceptable combinations of teachers and lessons. Lets look for 2D model of Lessons and Groups (Figure 3.2)

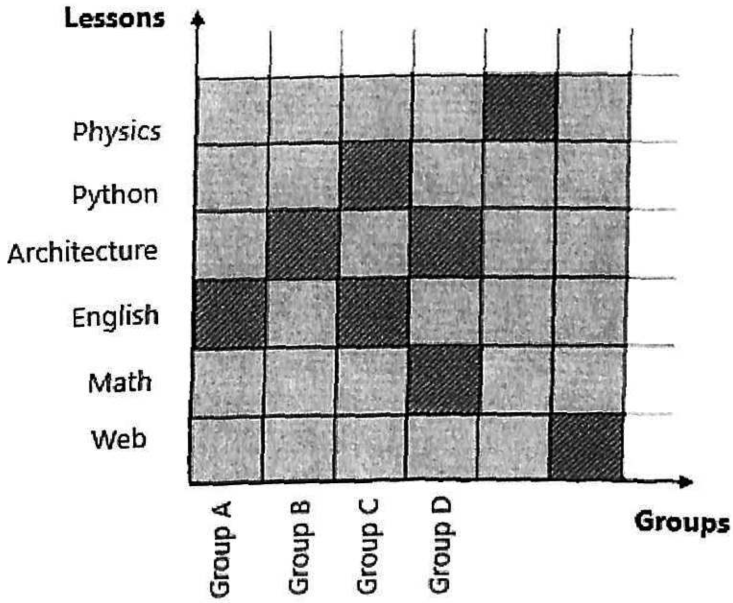

Figure 3.3: 2D Lessons - Groups

Shaded cells represent units of a study groups curriculum.
Lets look for 2D model of Groups and Teachers (Figure 3.2)

Lets look for 2D model of Groups and Teachers (Figure 3.2). Crosses mark unacceptable time intervals for teachers.

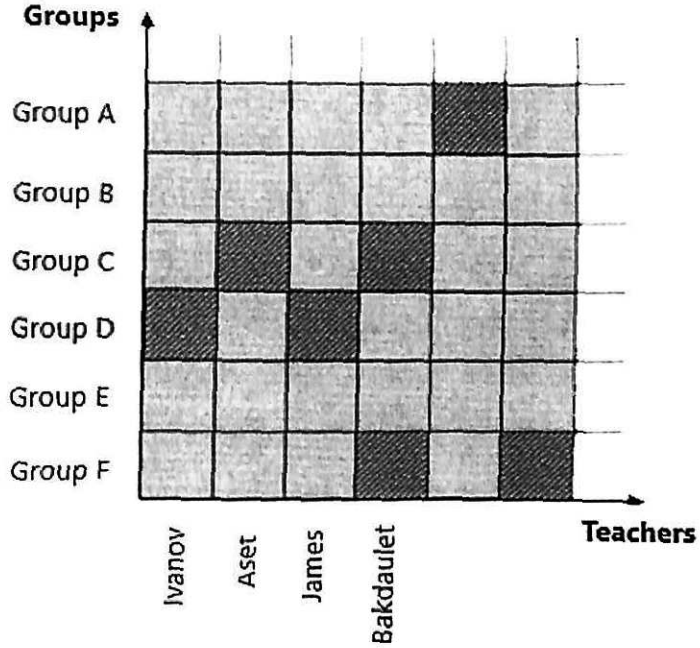

Figure 3.4: 2D Teachers - Groups

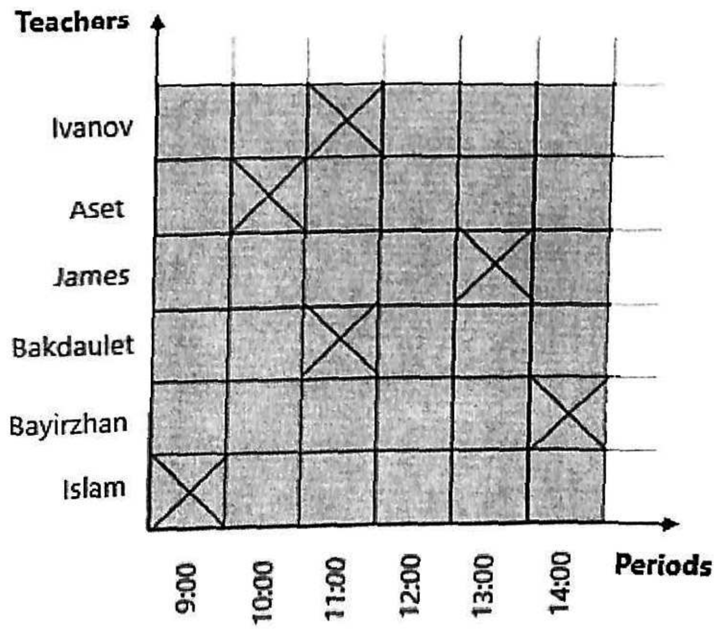

Figure 3.5: 2D Teachers - time

Similarly, in accordance with the initial conditions, the bases of the Group - Time and Room - Time are filled.

Filled three-dimensional space Group — Lesson — Teacher and wishes of time for groups, teachers and classrooms marked in the corresponding bases set the initial conditions. The task of compiling the curriculum consists in consistently filling the five-dimensional space of the schedule according to the required rules, taking into account all the requirements. The schedule space element is a schedule cell.

Let us estimate the dimension of this space using the example of SDU. In the fall semester of the 2019/2020 academic year, 450 teachers were involved in the educational process, 200 groups, which conducted a total of 1696 academic lessons, the number of classrooms 95, time intervals 9. Thus, the axis of the teachers contains 450 segments, the axis of the groups is 200 segments, the lessons axis is 1696 segments, the audience axis is 95 segments and the time axis is 9 segments. Therefore, the schedule space consists of 450 * 200 * 1696 * 95 * 9 = 130.507.200.000 cells, which must be filled in according to the required rules.

The model of the task of scheduling, taking into account the use of several machines of various classes for performing operations, with discrete processing time, and individual operation modes of machines, can be constructed in the form traditional for the linear discrete programming problem. Moreover, the need to take into account these restrictions leads to a significant increase in the dimension of the problem, which does not allow the use of known methods to solve it. It is necessary to look for a heuristic method based on the principle of ordering operations and truncating search options by finding the best time to complete the operation.

4. Development of a method for solving the problem

To solve the problem of scheduling classes, an approach is proposed consisting of two main stages:

1. analysis of the source data,
2. scheduling.

The principles of analysis of the source data are described in the second section. At the stage of scheduling, classes are directly arranged at the most suitable places.

4.1 Sequential scheduling method

The proposed method of scheduling classes provides for an alternate arrangement in the class schedule as decreasing pre-calculated difficulty factors. Each next lesson is assigned to the most suitable place.

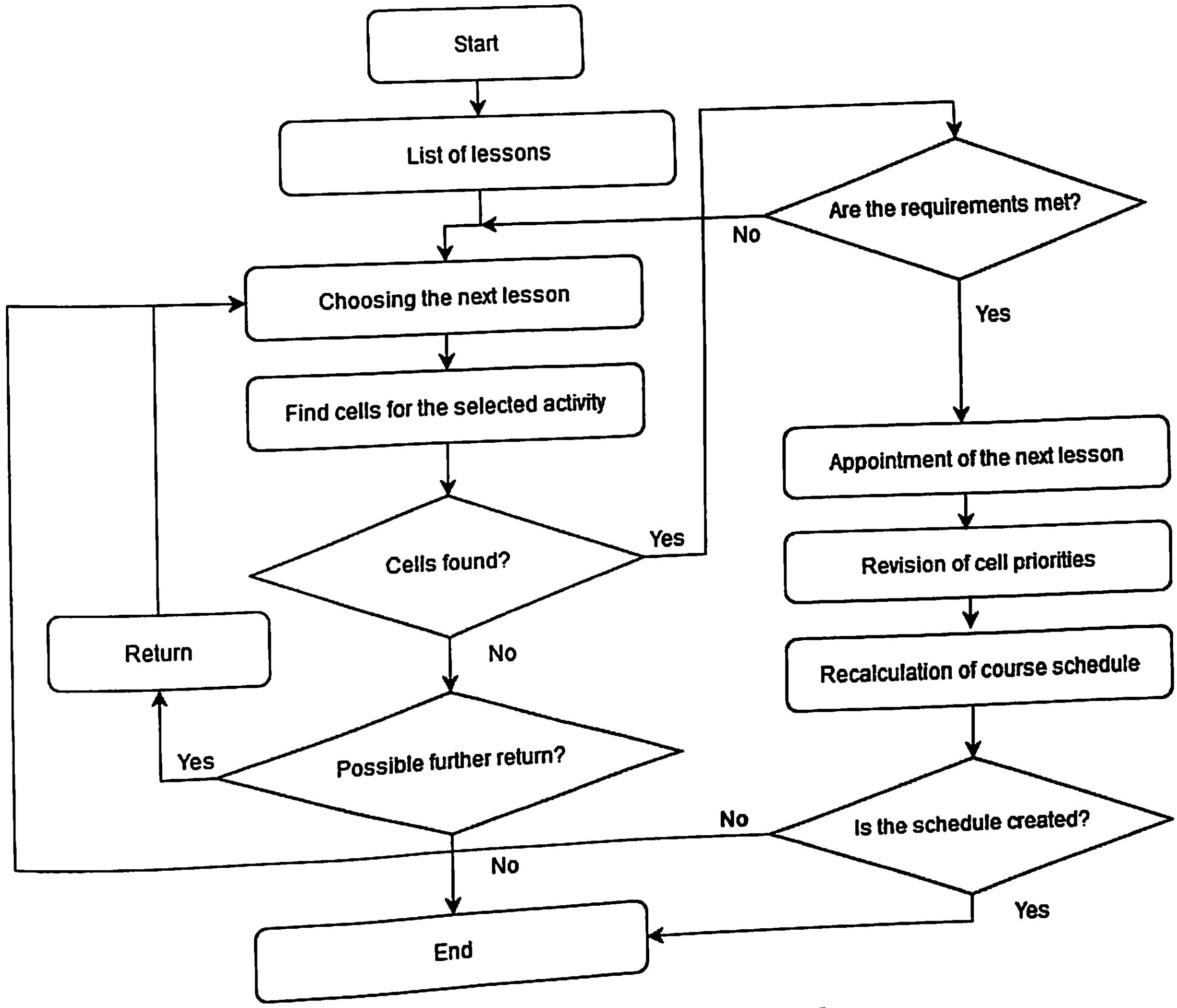

Figure 4.1: Workflow of algorithm

After each class is assigned, the priorities of the schedule cells are recalculated. In addition, after a number of assigned classes, we recalculate the complexity factors of the remaining classes in the queue. A generalized block diagram of the proposed algorithm is presented in Figure 4.1.

By assigning classes, we fill the schedule space. It is convenient to interpret this process as building a tree, the node of which will be a schedule cell. Each node has a list of possible directions of disclosure. This list corresponds to the list of all possible time intervals (segments on the time axis).The number of levels in the tree is equal to the total number of classes for placement. Passing from the root of this tree to any leaf, we get the university schedule. Each route from the root of the tree to the leaf will give a new version of the schedule. The challenge is to build this tree. A method is proposed to bypass this tree with a return when a deadlock occurs.

In this paper, the principle of constructing a drop-down tree is applied [27]. This means that not all of the tree is built, but the vertices are revealed as they are built.The next lesson is selected from an ordered list. From the list of possible directions of disclosure, the highest priority is selected and the disclosure of this direction is carried out, that is, the statement at the given time of the selected lesson. Then, the next lesson is assigned at the next level of the tree, for which the highest priority direction is also selected and revealed.Along the way, all the required checks for the possibility of disclosure (which include the whole variety of heterogeneous restrictions) of one direction or another are carried out. Directions are disclosed until all academic disciplines are assigned or at the next level there is not a single possible disclosure option. In this case, you need to perform a return - cancel the last setting and try to open another vertex. If all disclosure options at this level are exhausted, then it is necessary to return one more level up. If all disclosure options at the first level are exhausted, then under the given conditions it is impossible to draw up a schedule. If it is possible to expand all the nodes of one branch, then this branch will be one of the options for the schedule.The tree traversal order is shown in Figure 4.1. It should be noted that the use of the return mechanism leads to partial enumeration in the optimistic version and complete enumeration in the pessimistic version. Therefore, in this paper, it is proposed not to apply the return procedure, but to leave the classes unassigned, since it is most likely that in order to be able to arrange them together with the existing system of restrictions, it will be necessary to weaken the restrictions that impede the placement of these classes, which only a person can decide on.

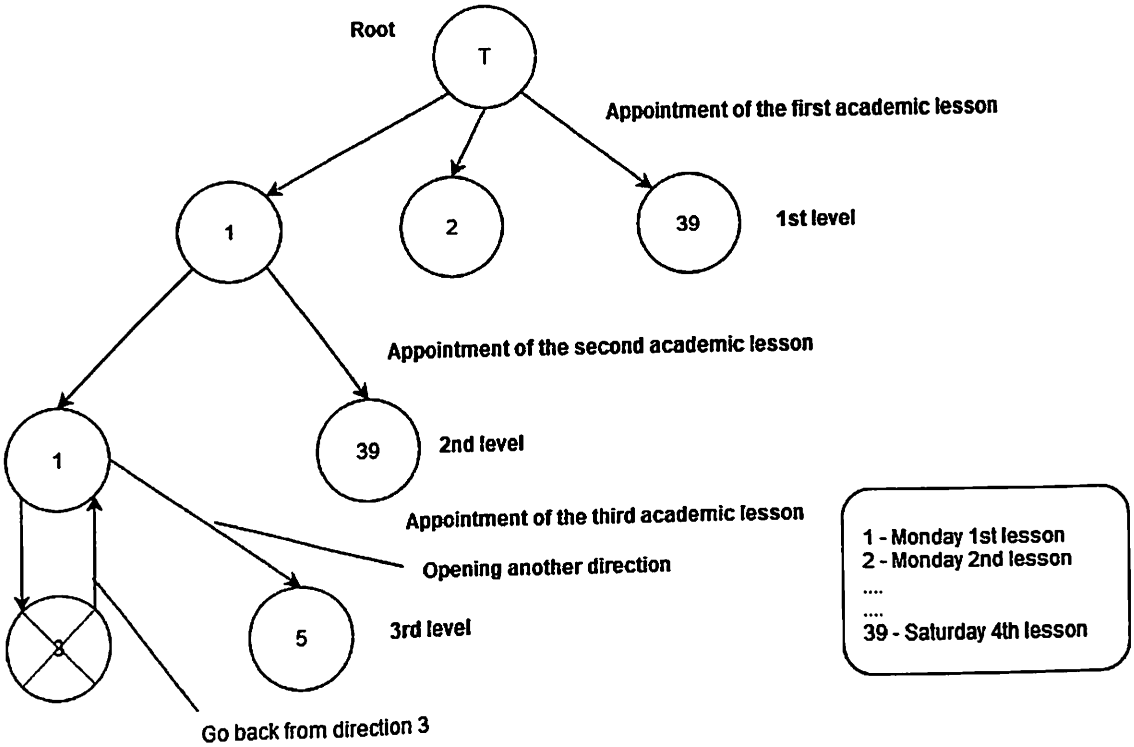

Figure 4.2: Tree schedule

If it was not possible to build the tree, it is necessary to analyze the initial conditions, remove the restrictions that impede the appointment of the classes left unassigned, and try again to draw up a schedule. To minimize the possibility of such situations occurring within the framework of the described procedure, it is advisable to solve the following tasks when assigning another lesson:

- selecting the best time for the lesson,
- selection of the best rooms for conducting classes at a specific time,
- recalculation of priorities of schedule cells.

Selection of the best rooms for the lesson

When scheduling, it is necessary to solve the problem of selecting an room to conduct a specific lesson in a specific period of time. Lecture ny and practical exercises are required to be carried out in general classrooms, and laboratory work in specialized classrooms. Since the list of possible classrooms for the laboratory classes is predefined, the selection of the classrooms is not difficult. Also, when choosing an classroom, it is necessary to take into account the advantages of departments for conducting classes in individual classrooms. According to the rights of use, classrooms are divided into the following groups:

university classrooms, cathedral auditoriums, cathedral-university classrooms. It is permissible to assign classes to all departments without exception to university classrooms. The cathedral room can be occupied for conducting classes only for those departments that are allowed to use this room. The cathedral-university room is allowed to be involved only if there are currently no free classrooms of another affiliation. Assigning a lesson to the room, it is necessary to take into account its capacity (number of seats). Teachers have wishes for buildings in the classrooms of which it is convenient for them to conduct classes. In drawing up Schedules must take into account the wishes of teachers to the corps, as well as the transitions of study groups and teachers from corps to corps. It is quite possible that from the point of view of saving the classroom fund it is desirable to choose one room, and from the point of view of the wishes of the teachers another.

If we do not consider the task of minimizing the transitions from room to room inside the building, then it is permissible to choose from each building only one audience that is currently free, which is most suitable for a particular lesson in terms of capacity. Then the final choice must be made by choosing from a limited list. The task of choosing an room can be solved in two stages:

1. formation of a list of the most optimal room of the required type in terms of capacity not occupied at a given time, taking into account affiliation in one of each building
2. selecting an audience from the list based on capacity, teacher wishes and minimizing the number of transitions from case to case for groups and teachers

The task of choosing the time of classes can be considered as fighting of several competing parties for common resources. Parties are: groups, teachers, classrooms.

A consistent method for solving the problem is proposed that provides a linear dependence of the solution time on the volume of the problem. The method is based on the arrangement of classes for placement and their purposeful placement in the most suitable places. Classes are sorted by decreasing complexity factor, which is calculated for each class at the analysis stage, the principles of which are described in section 2. When arranging classes, the following tasks: choosing the most suitable audience for conducting classes at a given time, choosing the most suitable time for classes, managing a custom priority system.

5. Development of Automated System

An automated system should have a hierarchical structure and consist of subsystems. In the case of an automated scheduling system, you must first develop the structure of the system, and then its main subsystems.

5.1 Development of the Structure of an Automated System for Scheduling Courses

The system should be multi-user with differentiation of access rights and functionality. The process of working with the system should be regulated by governing documents, user manuals. User actions should be coordinated among themselves organizationally and synchronized technically. Simultaneous editing of work curricula and work with the schedule, even if these actions are performed by different users from different client places, should be synchronized with each other in order to avoid loss of logical data integrity. During synchronization, it is necessary to carry out a number of checks according to the business processes of the subject area. For example, you cannot create a study flow for a discipline if this discipline is saved in the schedule for at least one group. This does not violate referential integrity of the data, but leads to a loss of logical integrity.

The system should have a hierarchical structure and consist of the following subsystems: hardware, software and methodological. The structure of the automated system is shown in Figure 5.1.

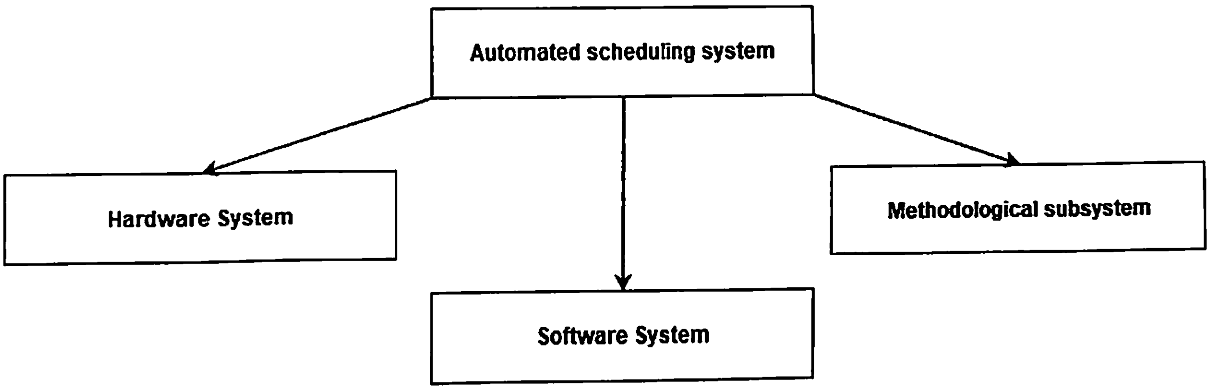

Figure 5.1: Structure of system

5.2 Hardware Components

The hardware subsystem provides the functioning of an automated system at the equipment level, providing communication and computing capabilities as well as data storage facilities.

Computing capabilities are provided by an automated scheduling server. The server also stores all the necessary data and processes user requests for access to the necessary data. As a server, it is desirable to use a high-performance computer with broad computing capabilities. The server must provide reliable data storage with the possibility of their backup and quick access to both read and write. To ensure reliable operation, it is required to provide reliable and stable power to the server and equip it with an uninterruptible power supply. For fast data processing, the server must have a high-performance processor system with one or more central processors and a sufficient amount of high-speed RAM. The amount of RAM and the speed of access to it very significantly affect the performance of the entire system as a whole [28]. To enable network operation, the server computer must be equipped with network equipment.

On the client side, personal computers are installed that carry out a dialogue with the user and organize a request to the server at the hardware level. Also, a number of non-critical checks are performed on the client side, which are required for the correct display of information and the correct organization of requests. There are no special requirements for the computing capabilities of client computers. All client machines must be equipped with network equipment.

The general structure of the hardware subsystem is shown in Figure 5.2.

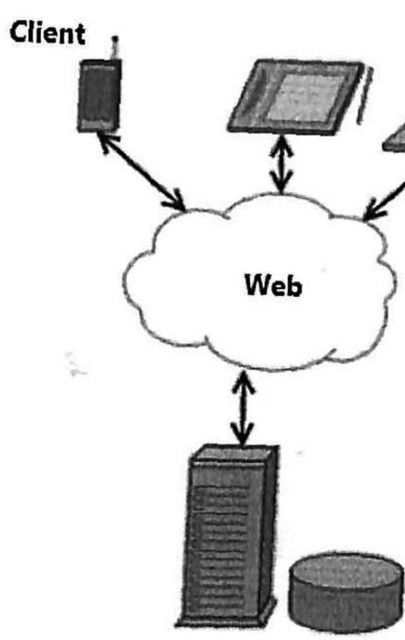

Figure 5.2: Hardware Architecture

It is advisable to allocate the schedule department to a separate network that does not have access to an external network, with the ability to access only the server and other computers of the department. This will provide additional protection against viruses and external attacks and ultimately the stability of training schedules.

5.3 Software Components

The software subsystem is a set of software modules that work in conjunction with each other, aimed at implementing the required functionality of the system [29].

The requirements for the subsystem can be classified as follows.

Key Stored Data:

- groups indicating the form of training and the number of people;
- group curricula;
- faculty;
- classroom fund;

- wishes for the schedule;
- work schedule of the university, classrooms, teachers;
- learning forms;

Main functions:

- adding and adjustment of curricula and disciplines;
- load calculation of teachers;
- formation of work plans of groups from curricula;
- the formation of educational assignments of departments;
- scheduling classes;
- scheduling exams;
- view schedule;
- input and editing of wishes of teachers;
- educational process control;

The software subsystem of the schedule department must have a common database stored on the server, provide multi-user access with the provision of data and functionality in accordance with the rights user.

5.4 Development of the Logical Structure

The logical architecture of any information system can be represented in the form of several layers [30]. Most often, the three most common layers are distinguished.

1. The presentation layer is the user interface and that part of the application that somehow supports its operation. Moreover, this layer can be represented by different clients with a different user interface;

2. Business logic layer - is engaged in processing data obtained from the data access layer;

3. Data access layer - organizes database queries and returns responses to these queries.

The model of the layers' logical architecture of the information system is presented in Figure 5.4.

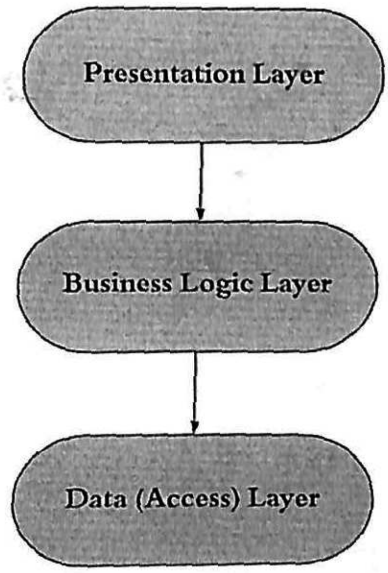

Figure 5.3: Logical Architecture

5.5 Development of a Scheduling Algorithm

In this subsection, the first are the classes of the subject area that underlie the module for scheduling classes and then the main steps that the system must perform in order to achieve its objectives goals.

As main algorithm used Metaheuristic algorithm. Metaheuristic deliver good quality in time.

Using the Metaheuristic algorithm, we save the partition. Thus, first we simplify the structure of the solution to the likely good, then we build the objective function based on the criteria for the quality of the schedule, we select good heuristics and apply metaheuristics [31].

Subject Area Classification The system was designed and implemented as part of an object-oriented approach. The basis of this approach is an object-oriented technology.

Object-oriented technology is based on the so-called object model. Its main principles are: abstraction, encapsulation, modularity, hierarchy, typification, concurrency and persistence. Each of these principles in itself is not new, but in the object model they were first applied together.

Thus, after a thorough analysis of the subject area, the following classes were identified that are included in the object model.

Class Day. Contains data and methods that manage data. Designed for storage and processing of data relating to a specific day of the week.

Class Group. Stores and processes information about a specific group.

Classroom Room. Stores and processes information about a specific room.

Class Teacher. Stores and processes teacher data.

Class Lesson. Stores and processes data on a specific academic lesson.

Class Start End times. It contains and processes data about a specific period of time and the state of the system in this period of time (busy teachers, classrooms, etc.).

Class Schedule. It contains all the necessary data and methods for scheduling. Includes instances of many other classes. In the class schedule, there are actually instances of all classes in the system in general. It is the main class, represents the top of the class hierarchy. Contains methods that directly control the scheduling process.

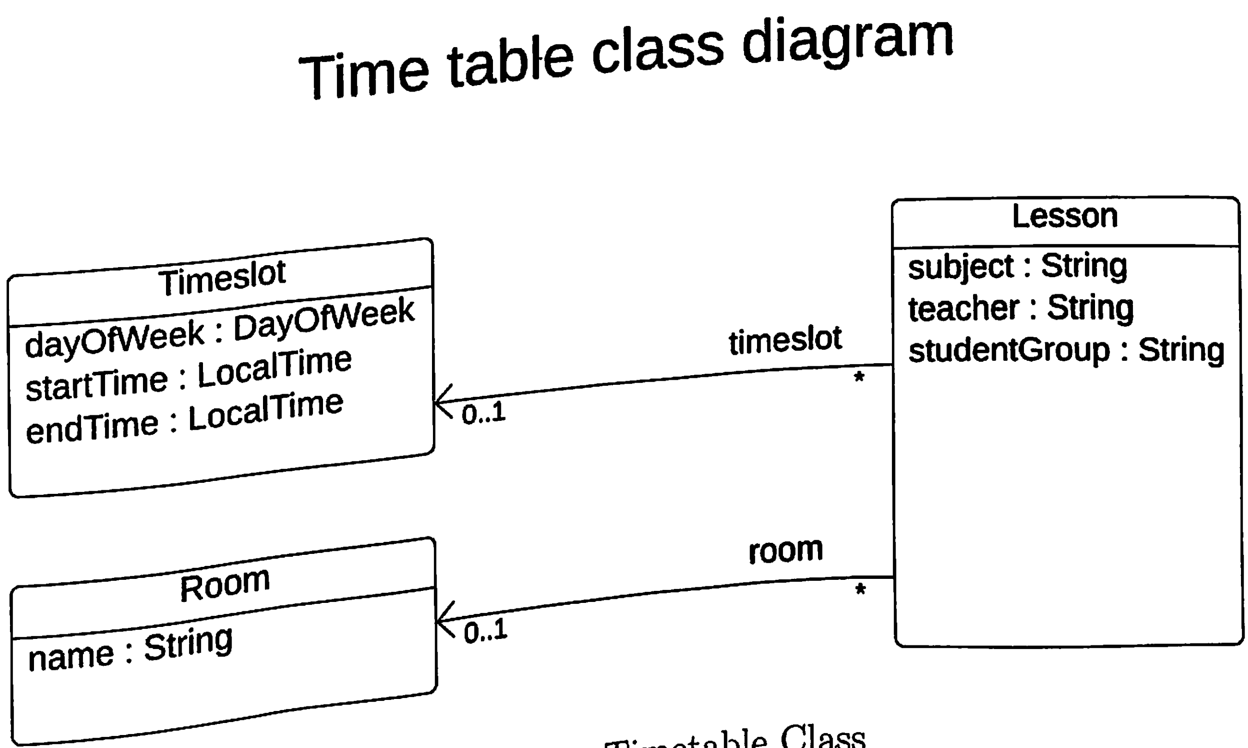

Figure 5.4: Timetable Class

Main Steps

Consider the main stages of the interaction of objects, leading to the desired result - a finished schedule. The main input for the algorithm is data from the curriculum.

Upload of Data

Before starting the scheduling algorithm itself, the system needs to do the work of forming structures with which the algorithm can work. To do this, you need to interrogate the database, get from it all the data necessary for work, and perform all the necessary checks for their correctness. All this It is executed when the "schedule" object is created.The base is interrogated, all necessary objects and auxiliary structures are created, classes are divided into blocks with considering all requirements. This process takes a lot of time, but it is very difficult for the system to operate with data, accessing the database directly if necessary. Access to the same data will occur repeatedly. In this case, the system will function not just slowly, but very slowly, since each call is accompanied by sending a request, processing the request, and issuing the result. To fulfill the request, the server needs to use resources (CPU and RAM) to build data structures, to produce the required table joins.

Scheduling

To schedule, you need to call the schedule method of the object "schedule". This method, in turn, sequentially calls the methods necessary for scheduling. We describe the operation of this method. First, the discipline search method for setting is launched. The method selects the lesson with the highest priority that has not yet been assigned. Then the method of searching for free cells for the selected lesson is launched. A block of free cells with the required characteristics (width and depth) and the highest total priority satisfying the main restrictions is searched. Then the method works, which checks the selected block to satisfy all other restrictions and requirements. If the selected block meets all the necessary requirements, then the method of assigning the selected lesson to the selected block of cells works. If not, then the method works, marking the selected cells as unsuitable for setting the selected discipline and a repeated search for the block of cells. And so on until cells are found that satisfy all the requirements. In the main algorithm, all checks related to the search for cells are deliberately divided into two methods (searching for free cells and checking all restrictions that are not met when selecting cells), which can significantly reduce the total number of checks, since the cell search method checks all cells, and verification method only selected. On real data, this gives a gain in time up to an order. After the appointment of a lesson, the method of revising the priorities of the cells, taking into account the new appointment, works.This is how the main cycle of the scheduling method works. Illustrating the operation of the method is presented in Figure 5.5.

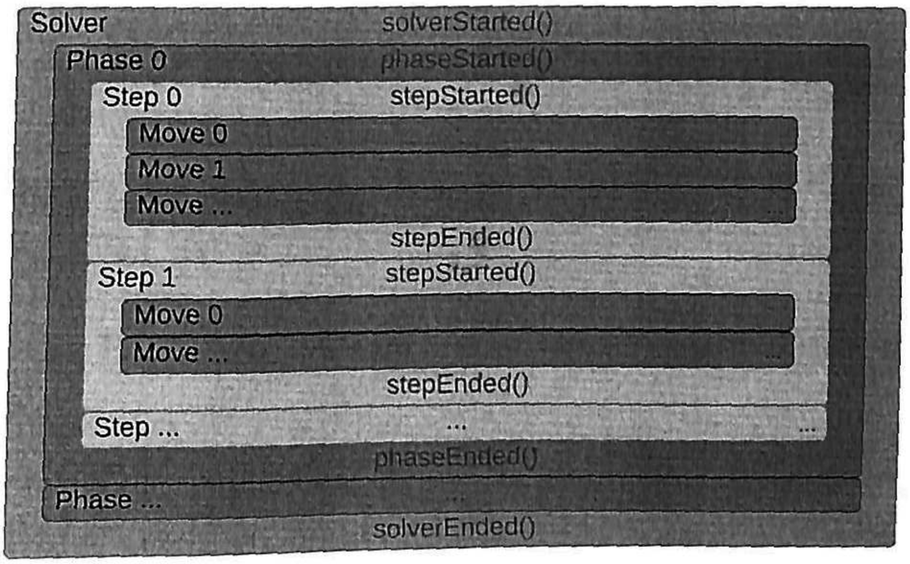

Figure 5.5: Solver Process

In particular, metaheuristic phases need to be taught to stop solving. For example, this may be due to a variety of factors, whether the time is up, or the perfect score was obtained only before the solution is used. You can't rely on finding the optimal solution (unless you know the optimal score), as a metaheuristic algorithm generally doesn't know the optimal solution.

This isn't a real-life problem, as finding the optimal solution can take longer than is available. The most important outcome is finding the best solution within the time available. So for that in problem of Scheduling used Termination. Terminates when an amount of time has been used. As example of Termination in Hours, Minutes, Seconds and etc. Figure 5.6

```
<termination>
  <hoursSpentLimit>2</hoursSpentLimit>
  <minutesSpentLimit>30</minutesSpentLimit>
</termination>
```

Figure 5.6: Termination Time

Description of software used

As the programming language, the base for the development environment, the JAVA language was chosen. This language combines the power, speed and efficiency of a low-level language and has all the advantages of a programming language. high level. JAVA is an object oriented language. Inner beauty, incredible power and huge capabilities make this language very attractive for development [32].

As the development environment was selected IntelliJ IDEA. JetBrains IntelliJ IDEA Software is a leading Java rapid development environment. IntelliJ IDEA is a high-tech complex of tightly integrated programming tools, including an intelligent source editor with advanced automation tools, powerful code refactoring tools, built-in support for J2EE technologies, integration mechanisms with the Ant / JUnit testing environment and version control systems, a unique code optimization and verification tool Code Inspection, as well as an innovative visual designer of graphical interfaces [33].

IntelliJ IDEA supports the basic principles of object-oriented programming - encapsulation, polymorphism and multiple inheritance, as well as newly introduced specifications and keywords in the Java language standard. Tools for working with databases and SQL files, including a convenient client and editor for the database schema. Also has Smart completion, tools for analyzing code quality, easy navigation, advanced refactoring and formatting for Java, Groovy, Scala, HTML, CSS, JavaScript, CoffeeScript, ActionScript, LESS, XML and many other languages.

As the main data format used XML. The XML language is used to create a data structure, which can then be transferred or stored. It is popular in all areas of programming, because it is distinguished by its ease of perception and universal reading by different applications.

The algorithm uses information from the SDU database. The data on groups, teachers, classrooms, temporary and other wishes, workplans, flows, work hours of the university are downloaded.

5.6 Development Web Interface

The system interface is divided into sections for various user roles. Authentication is required to enter some sections. Without authentication, the user can view the schedule for teachers, student groups, and classrooms. During authentication, the user role is checked and it falls into the corresponding section.

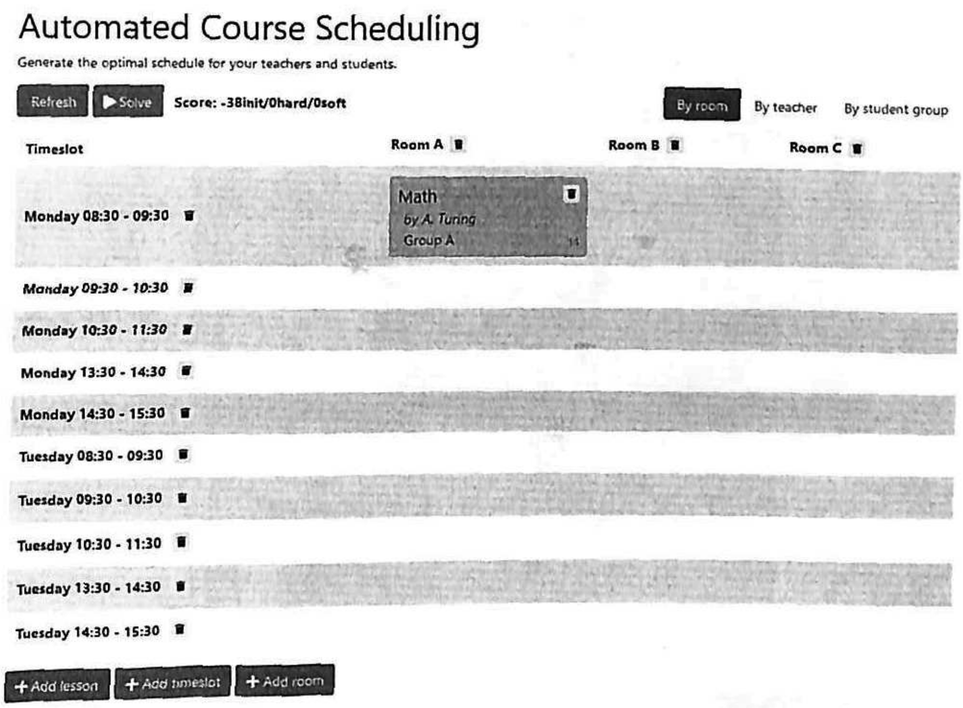

Figure 5.7: Homepage

At the moment, only the admin section has been worked out. The admin can perform the following actions on the system.

- Schedule by group, Figure 5.6;
- Schedule teachers, Figure 5.6
- Export special reports in XML;
- View special schedule forms, for example, a schedule for all classrooms or a schedule for a specific teacher or rooms, Figure 5.6;
- Accept applications to engage in an audience at a certain point in time;
- Modify records in some database tables via the web interface;
- Update data in local database from Portal database

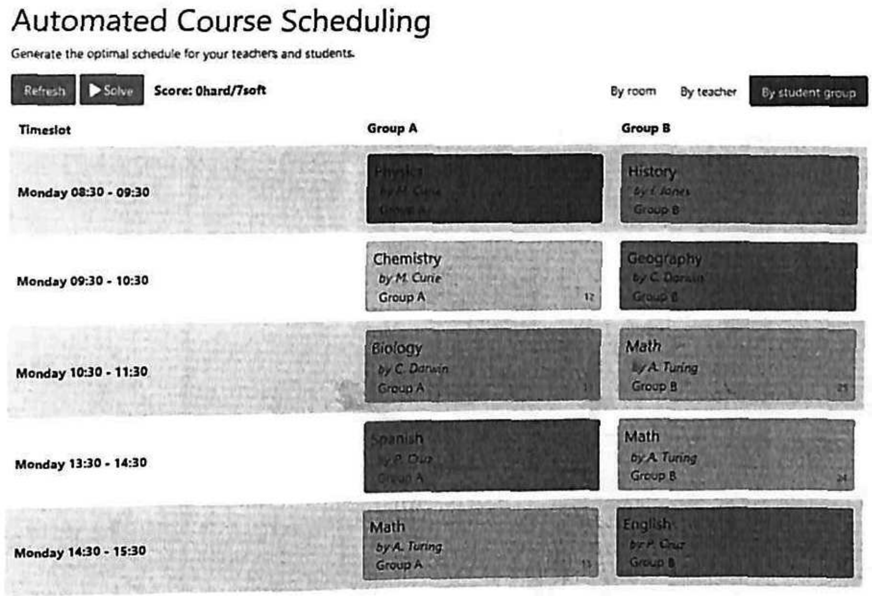

Figure 5.8: Scheduling By Groups

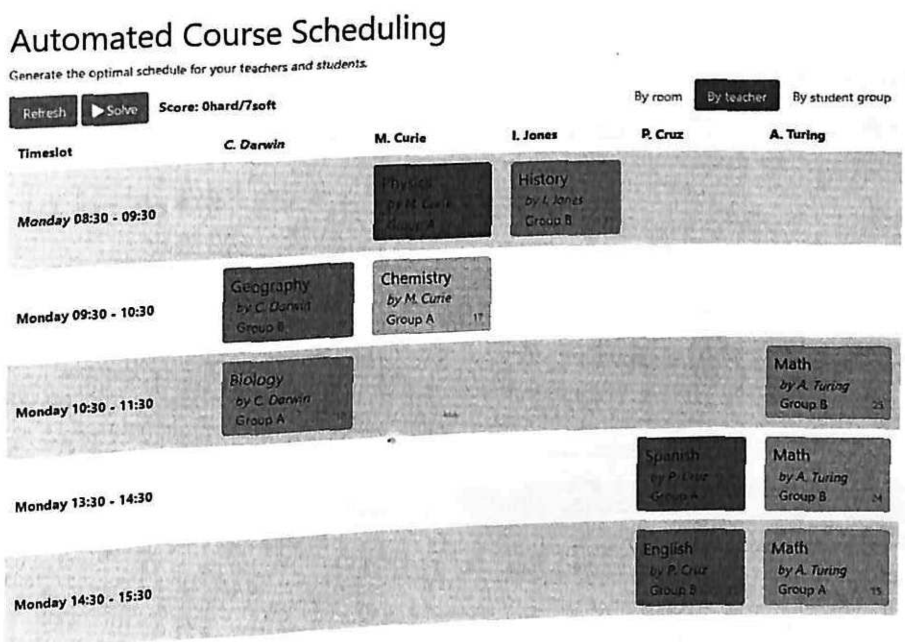

Figure 5.9: Scheduling By Teachers

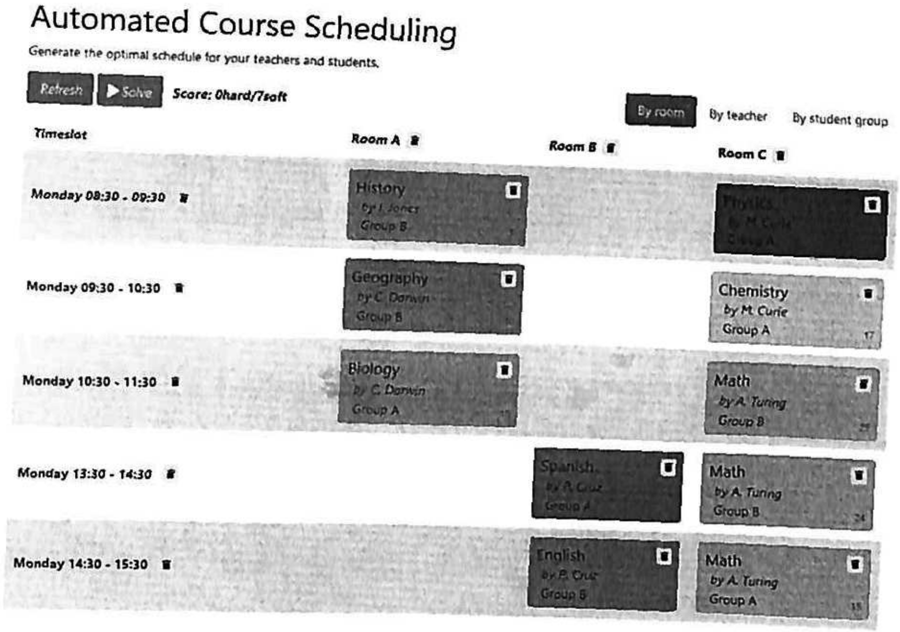

Figure 5.10: Scheduling By Rooms

The menu item for updating data from Portal is implemented, but is not tied to an interface, so the database is updated using a separate SQL Developer tool that runs SQL on the server and creates XML(Figure 5.6) formed file for importing to Local Database.

```
<CourseSchedule id="11">
  <id>0</id>
  <name>SDU</name>
<teacherList id="12">
<curriculumList id="13">
<courseList id="15">
<dayList id="16">
<timeslotList id="18">
<periodList id="19">
<roomList id="20">
<unavailablePeriodPenaltyList id="21">
<lectureList id="22">
<Lecture id="17632">
<id>3743</id>
<course reference="HIS 116.01"/>
<lectureIndexInCourse>0</lectureIndexInCourse>
<pinned>false</pinned>
</Lecture>
<Lecture id="17633">
<id>863</id>
<course reference="HIS 116.02"/>
<lectureIndexInCourse>0</lectureIndexInCourse>
<pinned>false</pinned>
</Lecture>
<Lecture id="17633">
<id>862</id>
<course reference="HIS 116.02"/>
<lectureIndexInCourse>1</lectureIndexInCourse>
<pinned>false</pinned>
</Lecture>
```

Figure 5.11: XML formatted information for Scheduling

As conclusion for this section, designed method, design schemes, and algorithms for solving the scheduling problem developed as part of the dissertation research are the basis for a hardware-software automated system.

Based on the "client-server" technology, a database has been developed that allows you to enter, store and edit the source data and set all the basic requirements and restrictions. The logical structure of the database is developed taking into account all the necessary requirements for the system, and allows you to relatively easily increase the functionality. The logical structure provides for a possible change in the source data, the introduction of additional restrictions, generalization to the most general case.

6. Analysis of results

The automated system developed as part of this dissertation research has been put into operation and is started using to schedule classes at SDU as MVP product. From a practical point of view, the most important is the question of assessing the quality of the results of its work - the class schedule. Therefore, the first subsection is devoted to the development of criteria for assessing the quality of class schedules. The second subsection is devoted to describing the results of the implementation of an automated system, starting with a description of the task and ending with obtaining estimates of quality indicators. The third subsection is devoted to the study of ways to improve the quality of problem solving.

6.1 Results of Testing System

This section describes the task, the process of preparing the initial data and evaluates the results of the solution.

Task description In the educational process of SDU in the fall semester of the 2018/2019 academic year, 226 teachers, 200 study groups, 4 faculties, 85 departments and 93 classrooms were involved.

```
<teacherList id="12">
<Teacher id="10009">
 <id>10009</id>
 <code>meirambek.zhaparov</code>
</Teacher>
<Teacher id="10012">
 <id>10012</id>
 <code>nazgul.abdinurova</code>
</Teacher>
<Teacher id="10013">
 <id>10013</id>
 <code>viktor.verbovskiy</code>
</Teacher>
<Teacher id="10014">
 <id>10014</id>
 <code>yedilkhan.amirgaliyev</code>
</Teacher>
<Teacher id="10019">
 <id>10019</id>
 <code>dinara.sarsenova</code>
</Teacher>
<Teacher id="10020">
 <id>10020</id>
 <code>aidos.shalbayev</code>
</Teacher>
```

Figure 6.1: XML formed teacher list for Scheduling

There were no restrictions on the duration of classes for normal groups. On average, the academic workload of a normal group is 40 academic hours per week.

An important part of the algorithm is taking into account the wishes of teachers, regarding the time of classes, expressed in numerical coefficients in the mathematical model. Example of Unavailable Period Time showm in Figure 6.1

```
<UnavailablePeriodPenalty id="1">
<id>1</id>
<course reference="CSS 514.01"/>
<period reference="10900"/>
</UnavailablePeriodPenalty>
```

Figure 6.2: Wishes of teachers

226 teachers submitted requests for the duration of the classes. In total, 600 wishes were submitted for an undesirable time for classes. Considering that the training cycle consists of two weeks of training, we get 14016 undesirable time

intervals.

The training load amounted to more than 2550 lessons with a total number of academic hours per week close to 3900.

The schedule of the semester under consideration did not fulfill 11 wishes for an undesirable time for classes.

Wish List		
Wish Number	Not Completed Wishes	Percent of Not Completed Wishes
600	11	1.8 %

Table 6.1: Table of Uncompleted Wishes in Schedule

Thus, the above indicators indicate a fairly high level of quality of the analyzed schedule. Only 1.8% of teachers' wishes for an undesirable time for classes were not satisfied,

Also analyzed methods, programs, algorithms gives as view of scheduling systems don't have effective way for scheduling based on priority of wishes(constraints). So we created new method for scheduling based on instructor and student preferences on priority facts.

It should be noted that the criteria of compliance with the wishes of teachers and students and the criterion of minimizing the number of windows conflict with each other. Those. for each criterion it is necessary to determine the priority of its importance. The mathematical model, the priority of importance of each criteria is determined by a numerical weight coefficient W = 1, 0.9, 0.8, ... etc.. In the current testing, each criterion was given the same weight coefficient is 1.

It is important to note that the number of collisions by group, teacher and audience was automatically calculated using the algorithm according to predefined rules, therefore, a large number of collisions for a schedule drawn up manually is NOT due to admin errors, but to incomplete initial data (for example, a subgroup is not indicated in the lesson for the subgroup).

The developed system for scheduling classes is a subsystem of the information system of educational-methodological management (SDU Portal). The automated system for scheduling classes was developed independently and was the core for the development of SDU Portal system, which was developed on its basis.

The functions of automated information processing, implemented and planned for implementation, concentrated in a single database, unification and optimal planning, allow us to formulate the concept of a unified university curriculum as a combination of working curricula of directions, specialties and training forms, optimal by the criterion of minimum resources, taking into account the set of restrictions established Departments and individual teachers, and in the future, students as part of the credit-modular principle of the formation of individual curricula

The results obtained in the previous subsection allow us to conclude that the lack of classrooms is not the only reason for unallocated classes. No less compelling reasons are the large volume and rigidity of the requirements for the schedule, as well as, probably, the empirical nature of the design algorithms used.

Algorithms for obtaining estimates are developed. The received estimates on the example of the fall semester of the 2018/2019 academic year allow us to conclude that it is of high quality.

Developed web interface for more flexible creating course schedules. Based in interface in easy to manage schedule in that form.

Analyzed results of scheduling which shows that problems is not in rooms capacity or not enough rooms or time, it mostly comes from lots of restriction made by teachers or departments when creating schedules.

7. Conclusion

In scheduling theory, there is no universal method for solving this problem. The high dimensionality of real scheduling problems does not allow the practical application of existing methods. It is advisable to seek a new empirical method involving the partial use of a combinatorial approach with the rejection of a complete enumeration of options by streamlining the list of operations.

As a specific applied problem to illustrate the main provisions of the dissertation, the task of compiling a class schedule for a higher educational institution, for which a solution with volume of limitations typical for a modern university. The development of a methodology for scheduling classes and implementing its automated system is relevant.

A formal model has been built, which provides for the need to perform an operation by several machines of various classes (specialized and universal), select a universal machine according to a given set of features, and take into account individual operating modes of the machines. Due to the high dimensionality of the model, its practical use requires search for a heuristic method effective in terms of labor intensity.

A method and algorithm for scheduling based on the ordering of its elements using flexible priorities are proposed. A model, calculation scheme, and algorithm have been developed for prioritizing schedule elements.

A method and algorithm for setting priorities in the scheduling process based on the neural network approach are developed.

A system of criteria for assessing the quality of class schedules and algorithms for their evaluation have been developed.

Based on the results obtained, an automated system for scheduling classes at a higher educational institution was built and put into operation.

References

[1] T. E.S., “Automation of schedule drawing in the educational management system”, *Science Territory*, vol. 1, no. 1, pp. 41–48, 2014.

[2] J. K. Lenstra, “Job shop scheduling”, *Combinatorial Optimization, Springer Link*, vol. 82, no. 1, pp. 199–207, 1992.

[3] J. E. G., *The Theory of Timetables*, ser. Training and teaching aids. Moscow State University, 2000.

[4] E. H. Daskalaki T. Birbas, “An integer programming formulation for a case study in university timetabling”, *European Journal of Operational Research*, vol. 153, no. 1, pp. 117–135, 2004.

[5] S.-T. C. Shyh-In Hwang, “Combinatorial optimization in real-time scheduling: Theory and algorithms”, *Journal of Combinatorial Optimization*, vol. 5, no. 1, pp. 345–375, 2001.

[6] M. M. A. A. O. Adewumi B. A.Sawyerr, “A heuristic solution to theuniversity timetabling problem”, *International Journal for Computer-Aided Engineering and Software*, vol. 26, no. 8, pp. 972–984, 2009.

[7] E. G. C. Jr., *Computer and Job-shop Scheduling*, ser. Scheduling. Wiley, New York, 1976.

[8] J. D. ULLMAN, “Np-complete scheduling problem”, *Journal of computer and system sciences*, vol. 10, no. 1, pp. 384–393, 1975.

[9] N. D. Edward M Reingold Jürg Nievergelt, *Combinatorial algorithms : theory and practice*, ser. Combinatorial analysis. Englewood Cliffs, N.J. : Prentice-Hall, 1977.

[10] L. W. M. Richard Walter Conway William L. Maxwell, *Theory of Scheduling*, ser. Scheduling. EDover, 2003.

[11] T. Yamada and R. Nakano, *Job-shop scheduling*, ser. IEE control engineering series 55. The Institution of Electrical Engineers, 1997.

[12] X. Xia and J. Zhang, "Optimal scheduling of conveyor belt systems under critical peak pricing", *Conference: IPEC, 2012 Conference on Power Ener*, vol. 1, no. 1, pp. 315–320, 2012.

[13] D. Ltd. (). Timetable pro, [Online]. Available: `https://digsee.com/en/products/small_programs/timetable/`.

[14] M. Lab. (). Express timetable, [Online]. Available: `https://www.mmis.ru/programs/avtor`.

[15] N. Company. (). Nika software, [Online]. Available: `https://nikasoft.ru/`.

[16] C. C. Inc. (). Cybermatrix class scheduler - student class scheduling software, [Online]. Available: `https://www.cybermatrix.com/class-scheduler.html`.

[17] 1. Company. (). 1c university, [Online]. Available: `https://solutions.1c.ru/catalog/asp_univer`.

[18] aSc Applied Software Consultants. (). Asctimetables, [Online]. Available: `https://www.asctimetables.com`.

[19] L. Lalescu and V. Dirr. (). Fet - free timetabling software, [Online]. Available: `https://lalescu.ro/liviu/fet/`.

[20] A. Foundation. (). Unitime - comprehensive university timetabling system, [Online]. Available: `https://www.unitime.org/`.

[21] Z. A. S., "Automation as the basis for optimal planning of the university educational process, автоматизация как основа оптимального планирования учебного процесса университета", *Mechatronics, automation, control.*, vol. 5, no. 1, pp. 45–48, 2007.

[22] K. Bulbul and P. Kaminsky, "A linear programming-based method for job shop scheduling", *Shaban University Journal*, pp. 1–26, 2013.

[23] M. Minoux, *Mathematical Programming: Theory and Algorithms*, ser. Programming. Wiley, 1986.

[24] A. S. Even A. Itai, "On the complexity of timetable and multi-commodity flow problems", *SIAM J: Comput.*, vol. 5, pp. 691–703, 1976.

[25] D. J. M. Gary, *NP-Complectness*, ser. NP-problems. MIR, 1982.

[26] K. Steele and H. O. Stefánsson, *Decision Theory*, ser. NP-problems. Metaphysics Research Lab, Stanford University, 2016.

[27] K. Y. G., "Theory and technology of programming. теория и технология программирования", *BHV Piterburg*, pp. 1–272, 2005.

[28] A. Neu. (). Why is ram so important and what does it do?, [Online]. Available: `https://phoenixts.com/blog/why-is-ram-so-important-and-what-does-it-do/`.

[29] S. McConnell, *Code Complete 2nd Edition*, ser. Programming. Microsoft, 2004.

[30] T. Tutisani. (). Global three layer application architecture, [Online]. Available: `http://www.tutisani.com/software-architecture/global-three-layer-architecture.html`.

[31] M. Z. Mohammed Azmi Al-Betar Ahamad Tajudin Khader, "University course timetabling using a hybrid harmony search metaheuristic algorithm", *IEEE Transactions on Systems Man and Cybernetics*, vol. 42, no. 5, pp. 664–681, 2012.

[32] H. Schildt, *Java: A Beginner's Guide, Eighth Edition*, ser. Programming. Oracle, 2018.

[33] JetBrains. (). Intellij idea, [Online]. Available: `https://www.jetbrains.com/idea/`.