ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Mingazova** Jméno: **Guzel** Osobní číslo: **474704**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra počítačů**

Studijní program: **Otevřená informatika**

Studijní obor: **Softwarové inženýrství**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Software complex for automated scheduling

Název diplomové práce anglicky:

Software complex for automated scheduling

Pokyny pro vypracování:

The diploma thesis examines timetable generation problem. The scheduling theory has been under studying for over 50 years and there is no unequivocal solution of scheduling that would satisfy all the conditions that have been established. The existing scheduling algorithms are examined and analyzed in the work. The developed software package will consists of 3 parts and have possibility of subsequent timetable correction. The main program based on the chosen algorithm prepares a timetable. The second part of the complex realizes checking of the created timetable of errors and concurrences. The third part represents a database in MS Excel files format that contains the necessary data about students, audiences, lecturers.

Seznam doporučené literatury:

1. Hahn-Goldberg S. Defining, Modeling, and Solving a Real University Course Timetabling. URL: http://tidel.mie.utoronto.ca/pubs/Theses/hahn-goldberg.masc.pdf.
2. Marco C., Krzyztof S., Mauro B., and Olivia Rossi D. ´An Effective Approach for the University Course Timetabling Problem´ Journal of Scheduling, vol. 9, no. 5, pp. 403-432, 2006.
3. Erben W., Keppler J. ´A genetic algorithm solving a weekly course timetabling problem´ Proc. of the 1st Int. Conf. on Practice and Theory of Automated Timetabling, LNCS 1153, pp. 198-211, 1995.
4. Colorni A., Dorigo M., Maniezzo V. ´A Genetic Algorithm to Solve the Time-table Problem´ Technical Report No. 90-060, Politecnico di Milano, Italy, 1990.
5. D. De Werra. ´An introduction to timetabling´, European Journal of Operational Research, vol. 19, issue 2, 151-162, 1985.
6. Kuldeep Singh Sandhu. ´Automating Class Schedule Generation in the Context of a University Timetabling Information System´, Phd dissertation, School of Management, Nathan Campus, Griffith University, 2001.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Miroslav Bureš, Ph.D., laboratoř inteligentního testování softwaru FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

Datum zadání diplomové práce: **16.04.2018** Termín odevzdání diplomové práce: **25.05.2018**

Platnost zadání diplomové práce: **30.09.2019**

Ing. Miroslav Bureš, Ph.D. podpis vedoucí(ho) práce	podpis vedoucí(ho) ústavu/katedry	prof. Ing. Pavel Ripka, CSc. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomantka bere na vědomí, že je povinna vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

.

Datum převzetí zadání

Podpis studentky

Czech Technical University in Prague

Faculty of Electrical Engineering

Department of Computer Science and Engineering

Master's Thesis

Software complex for automated scheduling

Guzel Mingazova

Supervisor: Ing. Miroslav Bures, PhD

Study Program: Open Informatics

Field of Study: Software Engineering

May 24, 2018

Acknowledgements

I would like to express my sincere acknowledgments to my supervisors, Ing. Miroslav Bures (CVUT) and Ing. Dmitrii Tumakov (KFU) for guidance and immense to help on my thesis. I want to especially thank my family, my dear brother and colleagues for supporting and encouraging me throughout my thesis work. Also, I want to appreciate Turilova Ekaterina, Enikeev Arslan all Czech team for the opportunity to participate in the Double Degree program (CVUT, KFU)

Declaration

I hereby declare that I have completed this thesis independently and that I have listed all the literature and publications used.
I have no objection to usage of this work in compliance with the act §60 Zákon č. 121/2000Sb. (copyright law), and with the rights connected with the copyright act including the changes in the act.

Prague, May, 2018 . . . .

Abstract

The diploma thesis examines timetable generation problem. The scheduling theory has been under studying for over 50 years and there is no unequivocal solution of scheduling that would satisfy all the conditions that have been established. In thesis we solve the problem of automated scheduling in the institute. Mathematical model and formation of timetable in particular cases are presented in the work. In the second part realized validation of the generated timetable on the presence of errors and concurrences. The third part represents the formalization of data store organization in MS Excel files format that contains the necessary data about groups, classes, lecturers in the Institute of Computational Mathematics and Information Technologies.

Contents

Introduction 3

1.1 Thesis content 3

Background from Theory of Scheduling 5

1.1 General information about Theory of Scheduling 5

1.2 Formulation of the scheduling task 6

1.3. Main issues and approaches of automated scheduling in a higher educational institution 7

1.4 Review and analysis of existing automated scheduling software products 9

1.5 The review of existing algorithms for solving tasks of automatic scheduling 20

Timetable scheduling 25

2.1 The procedure for scheduling in manual mode 25

2.2 Requirements and system constraints 26

2.3 Mathematical formulation of the problem 27

2.3.1 Notations 27

2.3.2 Constraints of the system 28

2.3.3 The objective function 30

2.4 Models of the scheduling with various problem formulation variations 31

2.4.1 Timetable 1 31

2.4.2 Timetable 2 34

2.5 Validation of the timetable 36

2.5.1 Description of the program 36

2.6 Collecting data organisation part 42

Conclusion 45

3.1 Summary 45

Bibliography 46

Chapter 1

Introduction

The problem of scheduling is the subject of scientific research since the middle of the last century. The area of their use includes various spheres of human activity: transportation, mass service, industry, education, etc.

The practical work propose a variety of problems that can not be effectively solved by methods of full enumeration. For the most models of scheduling theory finding the optimal timetable is formidable problem, and the solution for problems close to real conditions is even more difficult since these solutions must satisfy set often conflicting production, organizational and psychophysiological constraints.

The rapid growth and development of higher education institutions requires a new approach to solve the tasks of managing the educational activity in universities.

This approach has been realized in the practice of modern means of computer technologies and mathematical methods in the management of higher education institutions during the recent years. Nowadays, various types of automation systems for technical processes which have always been done manually are becoming increasingly common.

1.1 Thesis content

Manual scheduling by the scheduler (who responsible for timetable) for a sufficiently large number of academic groups of the institute takes a large amount of time. In addition, a large number of constraints and requirements that need to be considered within the formulation of the timetable lead to numerous mistakes that affect both the organization of the educational process and the effectiveness of teaching.

In this thesis, we will solve the problem of automatic scheduling at the institute.

Before starting the work a number of tasks were assign.

It is necessary to investigate the subject area and theoretical basis of this type of problem at the first stage. That is:

- To study the basics of the Theory of scheduling and formulate the problem of scheduling for educational institutions considering the features of the subject area.
- To examinate the existing literature on the topics of automated scheduling. To consider what questions are establishes and what approaches are considered in the articles.
- To investigate and analyse the software market of automated scheduling.

- To exanimate and analyse the algorithms for solving this problem.

At the second stage:

- To study the process of scheduling at the institute, to identify the main aspects.
- To develop a mathematical model for the solution of this problem considering studied material and the peculiarities of the educational process organization at the institute.
- To create a schedule via the formulated mathematical model. Use the library and find the solution.

Since the process of automation scheduling involves the possibility of manual editing by the scheduler, the task to develop a program for validation the generated timetable is assigned.

In addition, it is necessary to determine the form of data store organization necessary for scheduling.

Chapter 2

Background from Theory of Scheduling

1.1 General information about Theory of Scheduling

The need for planning arises throughout the entire life of people. Each person is engaged in planning their own time, without using complex algorithms and calculations. As soon as work and plans increase in quantity and number, there is a need for additional conditions, while the scheduling is carried out not for one person, but for a whole team - the complexity of planning appears. In the process of solving a number of such problems, principles, methodologies and general recommendations for scheduling were developed, which were subsequently studied within the framework of the science section – Scheduling Theory.

It is possible to single out areas where scheduling problems arise:

- scheduling of vehicles, public transport;
- production planning, in particular, operations ordering by executors and by time;
- planning classes in various educational institutions;
- scheduling of work schedules;
- when planning sports events;
- and etc.

Basically, the problems of scheduling theory are reduced to choosing among the set of admissible schedules of those solutions on which the optimal value of the objective function is achieved. Tasks are optimization, where the optimal is maximum or minimum of some objective function. The optimality of the schedule for its expediency, and the admissibility is understood in the sense of its feasibility.

The origin of the theory of schedules can be attributed to the beginning of the 20th century. In 1910, the American scientist Henry Lawrence Gantt developed the first format for **presenting the timetables, known as the «Gantt Chart» [1].** From 1903 to 1919 he publishes a number of scientific papers, exploring this area of planning.

Another illustrative example is the invention of the conveyor by Henry Ford. In April 1913, he launched the first experiment on the use of an assembly line [2]. In this interpretation, the conveyor is a way of organizing the production of products, in which the production process is divided into separate stages and simultaneously there are several products located at different stages of production. In this case, there is a sequence of operations (stages), which cannot be violated. Another important feature of this production is the parallel execution of independent operations. Conveyor production of Ford allowed reducing the time of cars production by 1.5 times.

All tasks of scheduling theory can be classified:

- On the division of scheduling theory (network planning, Project scheduling, compilation of temporary tables, Time tabling, vehicles scheduling, Transport scheduling, devices scheduling, Machine scheduling).
- According to the type of the objective function (tasks for constructing an acceptable schedule, multicriteria optimization tasks, tasks with minmax optimization criteria, tasks with summary optimization criteria).
- By the type of the solution sought (the tasks on matching, distribution, ordering).

1.2 Formulation of the scheduling task

Achieving the greatest progress on the way to solving the problem of scheduling theory is associated with great difficulties. Despite this, the tasks of scheduling theory have been and are still being practiced by a large number of specialists.

The most general formulation of the tasks of scheduling can be determined. Using a set of resources or devices, a certain task system must be executed. The goal is to find an efficient algorithm for ordering these tasks, which optimize the required measure of efficiency, given the properties, resources, and imposed constraints. As measures of efficiency the length of the schedule, the average time of the assignment and other factors can be taken. The information, on the basis of which the ordering decisions are made, is given in advance.

The formulation of the task of scheduling educational classes has its own peculiarities. If in the general theory of schedules the attendants at a given time cannot perform more than one task, then for the schedule of training sessions in the same classroom (process), activities (tasks) for two or more groups can be conducted simultaneously. When transferring the general theory to the schedule of educational classes, some assumptions should be taken into account:

- The set of tasks is the training classes of the teacher and training groups.
- The time model is discrete. The distribution is periodically repeated over a time interval. The lesson length is the same (the unit of discretization of the interval is the same).
- Classes have attachment to objects (teachers, training groups, classes).
- **Processes (classrooms, classes) have a capacity, some number $C \geq 1$, which determines** the number of tasks that can be processed at a given time.

Taking into account the peculiarities of this type, it is possible to formulate the scheduling task of educational classes. Using a variety of classrooms (computer, lecture, laboratory, etc.)

and a given set of time intervals (academic classes, lessons), it is necessary to make a distribution of educational sessions for objects (lecturers, teachers, training groups) with the best criterion of optimality.

1.3. Main issues and approaches of automated scheduling in a higher educational institution

The process of automating certain processes raises a number of problems, regardless of the field of implementation: acquisition and financing issues, implementation issues, software product maintenance.

The very concept of "automation" determines the use of technical and software tools that free a person partly or completely from direct participation in the processes. This can be the acquisition, transformation, transfer, use of materials and resources or information. Before the automation process, its formalization is realized and obtaining instructions for achieving the results of the process.

The automation process applied to the activities of the institution includes the following stages.

The first stage is the identification of the problem, the assessment of the possibilities and the need for automation.

The second stage includes the choice, implementation of the software, the formation of requirements for the software and hardware complex.

The third stage is the introduction of a software package (product).

The fourth stage implies maintenance and support of the hardware and software complex.

Taking into account the fact that the introduction of an automated system reduces the degree of human participation in the performance of certain operations, the automation process requires a clear formulation of the requirements for it, the allocation of the basic functions that need to be automated.

Appointment of an employee, responsible for carrying out this process and making decisions on automation matters, is also not an unimportant point. Qualified specialists within their duties must promptly eliminate the failures arising in the system, eliminate violations, and solve certain user problems.

The choice of the software product or its development environment should not be separated from the choice of technical support, which will be executed further. The existing information system of the university should also be taken into account.

Described process of analyzing the activities of a university is usually called a "pre-project survey of an institution." In the course of the survey, a complete model of organization is constructed, describing not only the interaction of structural units, but also the operations and information flows that they implement.

There is an opinion that an experienced dispatcher will be able to make a schedule so that it will meet the interests of the educational process and the public life of the educational institution. However, this does not reflect the reality. The manual solution of the task of scheduling is time-consuming, need qualified specialists; at the same time, the result of such decision often turns out to be far from optimal. After input of the initial information, its coordination is required, while the impossibility of obtaining the required schedule can be determined even at the analysis stage.

At the time of scheduling, deadlock situations may arise. All this requires changing the initial data and easing the constraints, and here undoubtedly a human is needed. Without introducing these changes, the timetable will not be of practical value. It should also be taken into account that the schedule can change during its use, i.e. after compilation, and here the human factor is very important. In this regard, it is important to support this process with automated methods and procedures.

The main advantage is that automated compilation eliminates a lot of routine work, such as: searching for possible options for adding more items to the schedule, checking the requirements, searching for random errors in the finished schedule, scheduling on paper in the form of various tables (for teachers, groups, audiences). The computer in this case is also a tool that greatly enhances person's abilities, because people are not able to sort through and analyze the same number of scheduling options in comparison with the computer.

In recent years, numerous attempts have been made to improve the planning of the educational process by constructing algorithms for optimizing the tasks of planning the academic work of the university using computer technology and software. Practical implementation of educational planning with the use of web technologies takes place only in a few universities. Analysis of the state of these developments allows us to draw the following conclusions:

- development and introduction of the tasks of automatic control systems by higher educational institutions is carried out in an initiative manner and these works, as a rule, are aimed at solving certain problems. The disunity of the groups of researchers and developers led to the creation of many systems aimed at developing algorithms and programs designed to serve only a particular institution.

- many systems impose all responsibility for accounting real requirements of the schedule on the developer. In particular, the requirements of teachers, constraints on the number of classes per day, per week - all these and many other routine tasks in such systems have to be solved most often by a person using methods of search.

- existing programs do not assume a multi-user mode of operation and do not support all necessary electronic document management.

- development of standard unified elements for the creation of a unified automated management system for higher education is not being introduced.

- the existing programs have a very inconvenient interface for entering the initial data and editing the completed schedule.

When developing algorithms for automated scheduling, the problem of creating universal algorithms that take into account the specific conditions of each specific problem is acute. Such algorithms should be sufficiently "flexible", i.e. without substantially changing them, it would be possible to include and exclude requirements from the system of requirements for the schedule.

For the systems of scheduling, there is a strong dependence on the specifics of particular educational institutions at the level of mathematical models and data representation, which makes the use of standard systems more difficult. The system, created in one university, usually cannot be used effectively in the other one without modification and refinement. In addition, many of these systems were created quite a long time ago and with their help it is impossible to solve the task effectively.

To solve existing problems, it is required to build a flexible and easily adaptable system based on new principles, using modern web technologies. A system that makes up the schedule in accordance with selected criteria and specified requirements is needed. To cover the most typical cases, it is necessary to create several typical algorithms that implement scheduling. This system should be able to supplement and modify the existing database and user interface. All this would make it possible to specify in each institution the requirements that meet its conditions, and by selecting and setting the appropriate algorithm - to obtain the required schedule.

1.4 Review and analysis of existing automated scheduling software products

Many researchers have been engaged in solving the problem of compiling the curriculum both in secondary educational institutions and in higher education institutions for a long time [3,4,5]. The problem still remains relevant and unsolved.

Given the fact that the products are oriented at schools, lyceums, colleges, universities, scheduling is complicated by the need to take into account a large number of conflicting factors: the capacity of auditoriums, sanitary and epidemiological rules and norms (SANPIN), the study load and preferences of teachers, the classroom fund of the educational institution, the study load of pupils (students) and others. Automating curriculum creating process is a way out of this dilemma.

Regardless of automation objects, the domestic market of information systems offers a considerable number of products for implementation, the ultimate goal of which is to improve the quality of education. However, each of the products has both advantages and disadvantages, their functionality is very different. Hardware requirements and other features can cause difficulties in adapting to a particular educational institution. Despite all the conditions, these software products solve one problem.

In this paper, the most popular and widespread products on the Russian software market with demo versions were examined, which allow analyzing the interface and embody the process of scheduling in practice.

The following criteria became important in the analysis of software tools and allowed us to compare them with the rest:

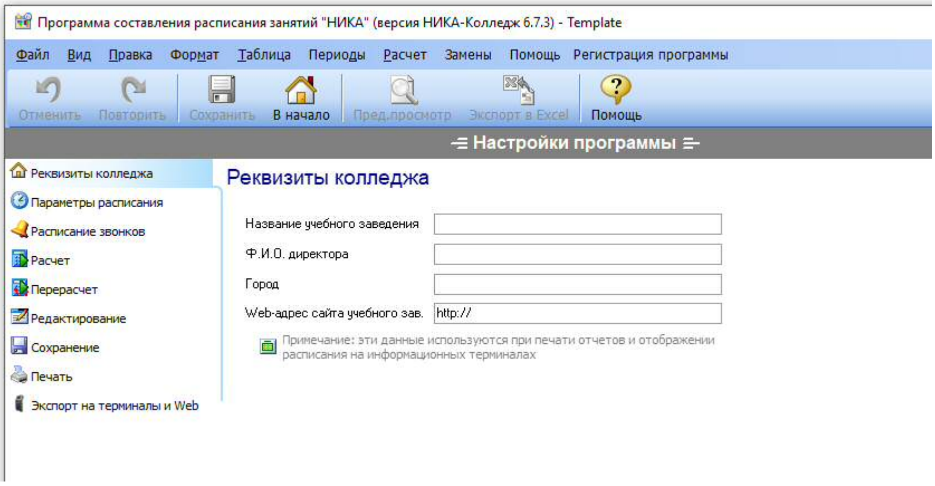
- Flexible setting. How many constraints are there to set to optimize the schedule?
- Convenience of the interface and ease of mastering. Whether the interface is intuitively understood, how comfortable it is for the user who does not have deep computer knowledge to apply.
- The efficiency of the algorithm. How well the timetable reflected the criteria and requirements.
- Price.
- System requirements of the programs.
- Adjusting and optimizing the completed schedule. Is it possible to make adjustments to an already created schedule.

We provide a brief description of each of the studied programs.

The program "Nika-College"

The "Nika-College" program is a product of Nika-Soft, which deals with programs of automation the activities of educational institutions. The program under consideration automates the scheduling of classes for colleges, vocational schools and technical schools.

The program interface contains 9 blocks. Before starting to work with this program, it is suggested to establish the requisites of the educational institution in the window "Requisites of the college" (Fig. 1.1) for further display on the schedule.


Программа составления расписания занятий "НИКА" (версия НИКА-Колледж 6.7.3) - Template
Файл Вид Правка Формат Таблица Периоды Расчет Замены Помощь Регистрация программы
Отменить Повторить Сохранить В начало Пред.просмотр Экспорт в Excel Помощь
Настройки программы
Реквизиты колледжа
Параметры расписания
Расписание звонков
Расчет
Перерасчет
Редактирование
Сохранение
Печать
Экспорт на терминалы и Web

Реквизиты колледжа

Название учебного заведения

Ф.И.О. директора

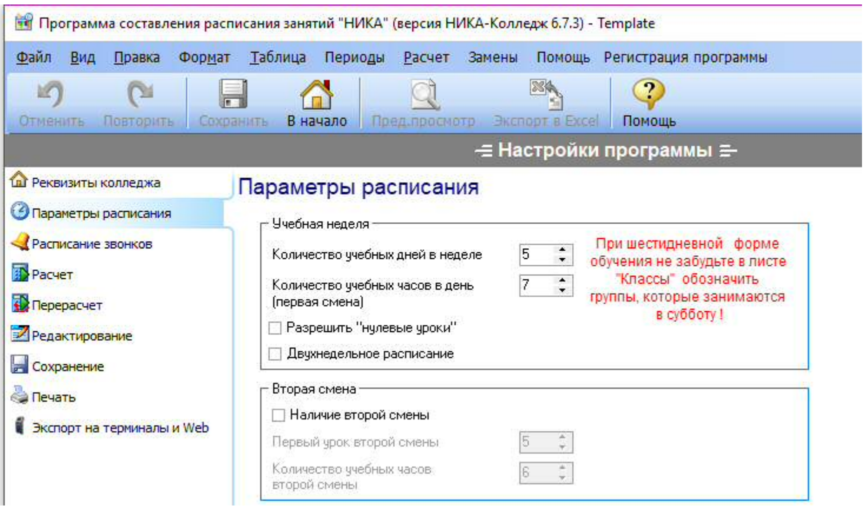
Город

Web-адрес сайта учебного зав. http://

Примечание: эти данные используются при печати отчетов и отображении расписания на информационных терминалах

Figure 1.1: Requisites of the college

In the "Schedule options" window (Fig. 1.2), it is suggested to make constraints on the studying weeks, fill in the lists of groups, cabinets, subjects and group loads.


Программа составления расписания занятий "НИКА" (версия НИКА-Колледж 6.7.3) - Template
Файл Вид Правка Формат Таблица Периоды Расчет Замены Помощь Регистрация программы
Отменить Повторить Сохранить В начало Пред.просмотр Экспорт в Excel Помощь
Настройки программы
Реквизиты колледжа
Параметры расписания
Расписание звонков
Расчет
Перерасчет
Редактирование
Сохранение
Печать
Экспорт на терминалы и Web

Параметры расписания

Учебная неделя

Количество учебных дней в неделе 5

Количество учебных часов в день (первая смена) 7

□ Разрешить "нулевые уроки"

□ Двухнедельное расписание

При шестидневной форме обучения не забудьте в листе "Классы" обозначить группы, которые занимаются в субботу !

Вторая смена

□ Наличие второй смены

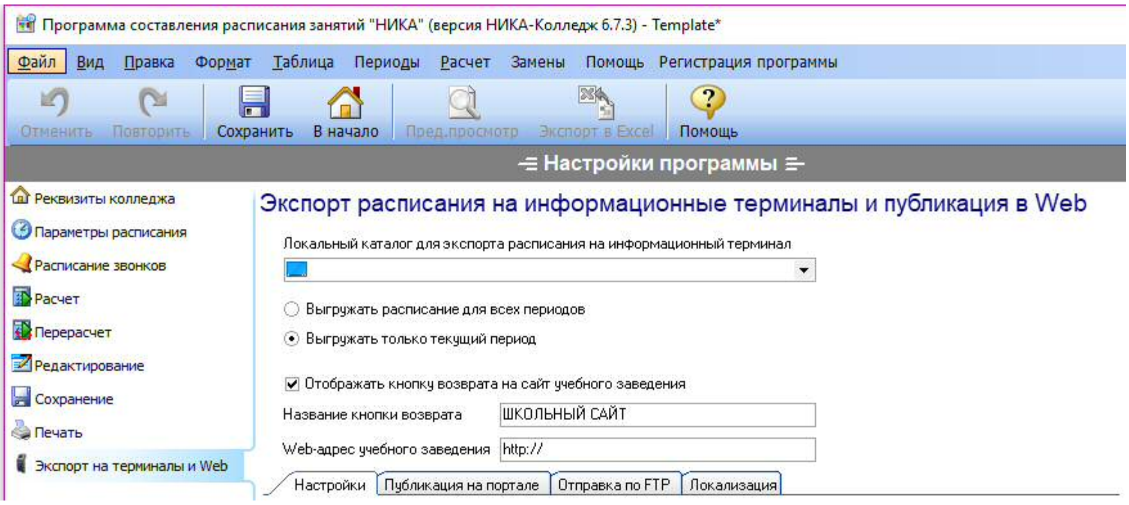
Первый урок второй смены 5

Количество учебных часов второй смены 6

Figure 1.2: Schedule options

Then the preparation for the calculation and the automatic calculation of the schedule. The calculated schedule appears on the "Schedule" sheet, where by manual editing it is possible to modify it and print it.

Completed schedule can be published on the website and exported to the information terminal (Fig.1.3).



Экспорт расписания на информационные терминалы и публикация в Web

Локальный каталог для экспорта расписания на информационный терминал

○ Выгружать расписание для всех периодов

● Выгружать только текущий период

☑ Отображать кнопку возврата на сайт учебного заведения

Название кнопки возврата: ШКОЛЬНЫЙ САЙТ

Web-адрес учебного заведения: http://

Figure 1.3: Schedule export

The program provides the following functions:

- division of groups into students on a five- or six-day period;
- flexible starting hours of classes and the second shift;
- splitting groups into subgroups (up to three), including splitting on different subjects;
- doubling, three times increase and so on for a given number of hours;
- scheduling considering the features of the existing classroom fund;
- association of groups into streams (including further division into groups);
- teaching two or more subjects by one teacher;
- displaying and printing the schedule of classes in five forms, including individual teacher schedules and timetables for classes;
- export of the schedules for display on the college website and at information stands;
- export of the schedule to Excel;
- **convenient tool for hourly monitoring individual teachers’ schedule, as well as "friendly** semi-automatic" for manual editing of the timetable.

The program takes into account:

- the presence of a second shift, a five- or six-day form of studying;
- characteristics of the office fund;
- methodical days and unwanted hours of work for teachers;

When creating the schedule of classes, an effective mechanism for monitoring the correctness of input of the initial data is provided, signaling the presence and nature of user errors [6].

This approach gives the advantage of the ability to evaluate the solutions obtained, the disadvantage is that a large number of constraints make it difficult to make a decision when one or constraints for optimizing the schedule are weakened.

This program has an automatic mode of operation, not providing optimal criteria usage, so the possibility of such a curriculum is limited to obtaining only an acceptable schedule, but not optimal. This program can be used for preliminary analysis of constraints in the task.

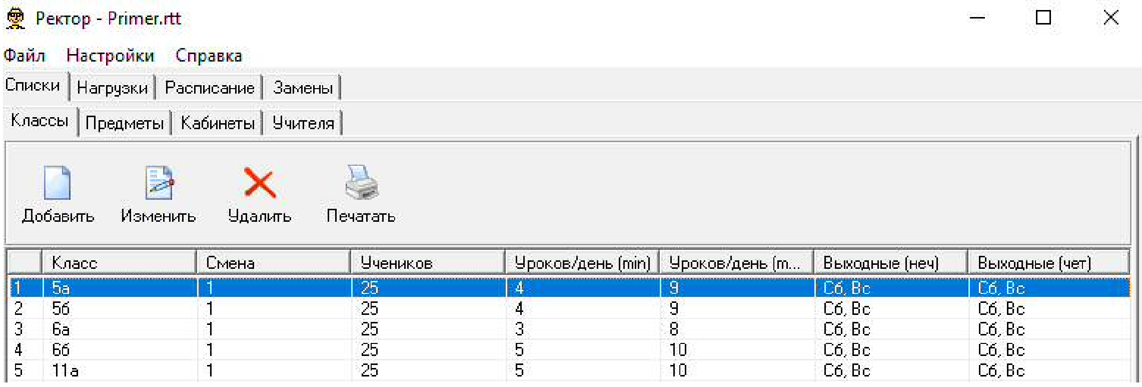
The system requirements for the program are as follows. The executable module of the program (Shedule5.exe) takes about 3.5 Mb. The file sizes for storing the schedule, depending on the size of the school, range from 20 to 300 Kb. "NIKA" can be installed on a computer with the Windows operating system, starting with the version of Windows 2000. In operating systems of the Linux family, the application of the program requires the use of appropriate emulators of the Windows environment. For working with the schedule it is recommended to use the screen resolution modes from 1024 * 768 and more, with a 32-bit color palette. [7]

The program "Rector-University"

The Rector-University program is a tool for scheduling classes in the system of higher professional education, in the Windows environment. According to the computer magazine PC Magazine for 2008 it was the best program in the Russian software market.

The program interface consists of four sections: "Lists", "Loads", "Schedule" and "Substitutions".

The "Lists" section (Fig. 1.4) serves for entering, editing and printing lists:
Departments, Specialties, Groups, Disciplines, Auditoriums, Lecturers, Types of activities.



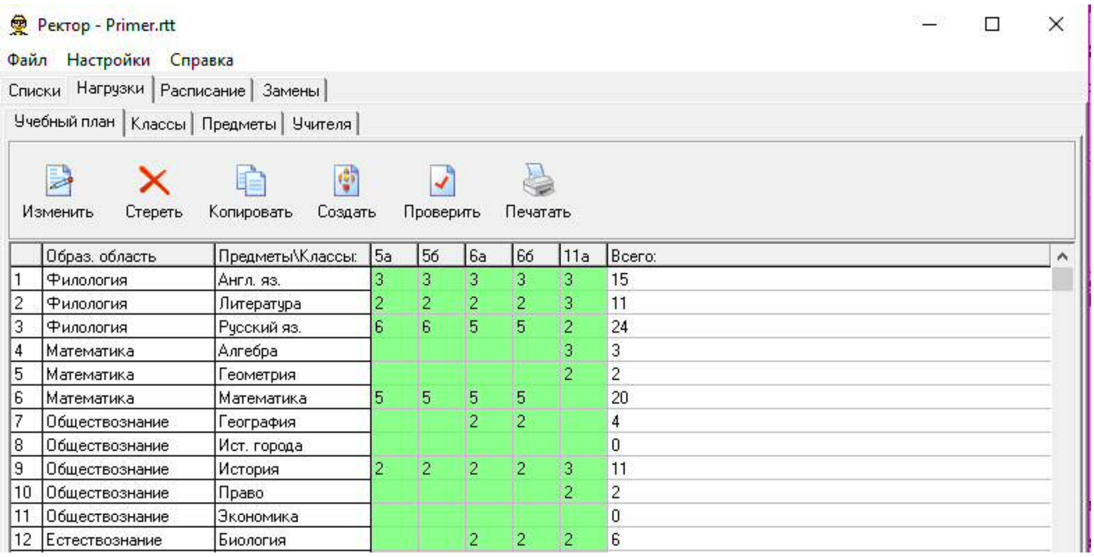
	Класс	Смена	Учеников	Уроков/день (min)	Уроков/день (m...	Выходные (неч)	Выходные (чет)
1	5а	1	25	4	9	Сб, Вс	Сб, Вс
2	5б	1	25	4	9	Сб, Вс	Сб, Вс
3	6а	1	25	3	8	Сб, Вс	Сб, Вс
4	6б	1	25	5	10	Сб, Вс	Сб, Вс
5	11а	1	25	5	10	Сб, Вс	Сб, Вс

Figure 1.4: Lists

The "Loads" section (Fig. 1.5) provides input, editing and printing of the following sections:

- study plans for specialties,
- load of teachers,

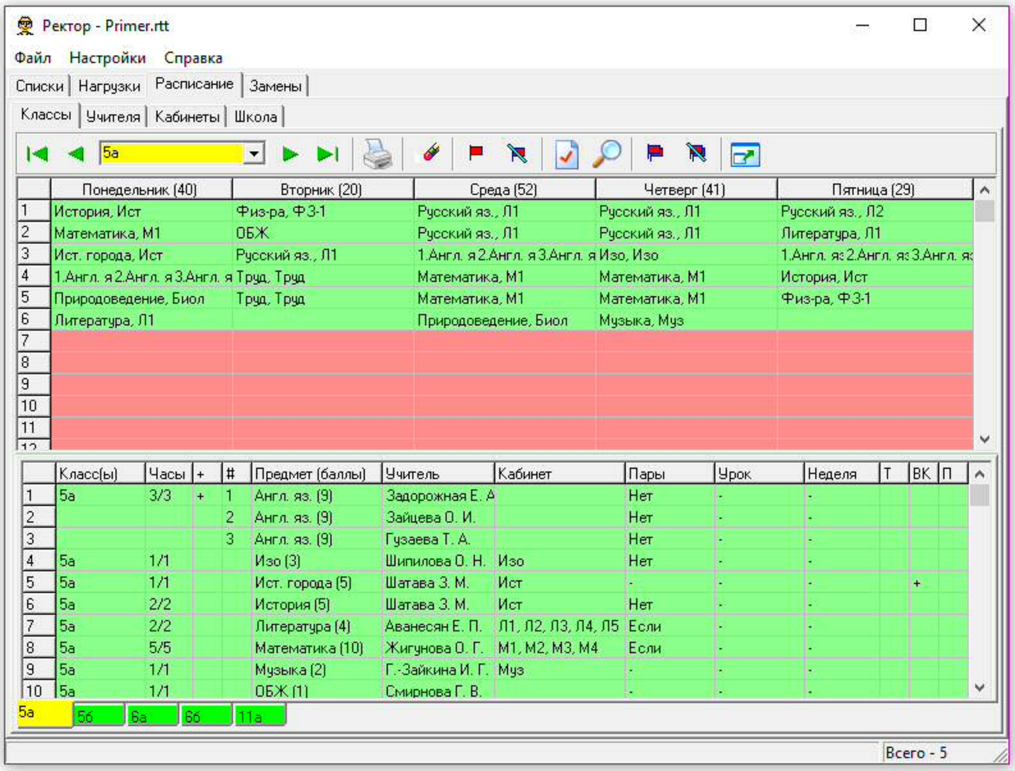
- schedules for the distribution of hours by week for a semester,
- reports on the load of teachers, departments and the university as a whole.



	Образ. область	Предметы\Классы:	5а	5б	6а	6б	11а	Всего:
1	Филология	Англ. яз.	3	3	3	3	3	15
2	Филология	Литература	2	2	2	2	3	11
3	Филология	Русский яз.	6	6	5	5	2	24
4	Математика	Алгебра					3	3
5	Математика	Геометрия					2	2
6	Математика	Математика	5	5	5	5		20
7	Обществознание	География			2	2		4
8	Обществознание	Ист. города						0
9	Обществознание	История	2	2	2	2	3	11
10	Обществознание	Право					2	2
11	Обществознание	Экономика						0
12	Естествознание	Биология			2	2	2	6

Figure 1.5: Loads

Section "Schedule" (Fig. 1.6) is intended for scheduling. The schedule can be created for specific group, certain teacher, separate auditorium, university as a whole.



	Понедельник (40)	Вторник (20)	Среда (52)	Четверг (41)	Пятница (29)
1	История, Ист	Физ-ра, Ф3-1	Русский яз., Л1	Русский яз., Л1	Русский яз., Л2
2	Математика, М1	ОБЖ	Русский яз., Л1	Русский яз., Л1	Литература, Л1
3	Ист. города, Ист	Русский яз., Л1	1.Англ. я 2.Англ. я 3.Англ. я	Изо, Изо	1.Англ. яз 2.Англ. яз 3.Англ. яз
4	1.Англ. я 2.Англ. я 3.Англ. я	Труд, Труд	Математика, М1	Математика, М1	История, Ист
5	Природоведение, Биол	Труд, Труд	Математика, М1	Математика, М1	Физ-ра, Ф3-1
6	Литература, Л1		Природоведение, Биол	Музыка, Муз	
7					
8					
9					
10					
11					
12					

	Класс(ы)	Часы	+	#	Предмет (баллы)	Учитель	Кабинет	Пары	Урок	Неделя	Т	ВК	П
1	5а	3/3	+	1	Англ. яз. (9)	Задорожная Е. А		Нет	-	-			
2				2	Англ. яз. (9)	Зайцева О. И.		Нет	-	-			
3				3	Англ. яз. (9)	Гузаева Т. А.		Нет	-	-			
4	5а	1/1			Изо (3)	Шипилова О. Н.	Изо	Нет	-	-			
5	5а	1/1			Ист. города (5)	Шатава З. М.	Ист	-	-	-		+	
6	5а	2/2			История (5)	Шатава З. М.	Ист	Нет	-	-			
7	5а	2/2			Литература (4)	Аванесян Е. П.	Л1, Л2, Л3, Л4, Л5	Если	-	-			
8	5а	5/5			Математика (10)	Жигунова О. Г.	М1, М2, М3, М4	Если	-	-			
9	5а	1/1			Музыка (2)	Г.-Зайкина И. Г.	Муз	-	-	-			
10	5а	1/1			ОБЖ (1)	Смирнова Г. В.		-	-	-			

Figure 1.6: Schedule

The "Substitutions" section allows to operate with teacher replacements.

The program provides the opportunity to schedule the lessons in three modes (automatic, manual, combined) and move from one to another.

When compiling a schedule, the program takes into account the following features:

- the distribution of lecturers in the departments and groups in the specialties;
- scheduling of load distribution by weeks;
- reports on the distribution of loads of departments and universities by occupation;
- automatic and manual scheduling modes.

When scheduling in manual mode, the program suggests possible options for arranging the lessons of the chosen teacher, possible options for filling empty cells in the group schedule, and monitoring the number of seats in the classrooms.

It is possible to save the finished schedule of one, all or some groups and teachers in Microsoft Word, Excel or HTML formats. [8]

The program "Rector-University" can run on any computer with the operating system Windows XP, Windows Vista or Windows 7.

Among the shortcomings are the lack of client-server architecture, the ability to work through the web interface, as well as the lack of the ability to refine and adjust the constraints of the task. The program provides search only for the allowed schedules (there is no optimization).

“AVTORaspisanie” system

The system is one of the programs of the MMIS Laboratory designed to automate the management of the educational process in higher and secondary special educational institutions.

The system has eight basic modifications of the program for various educational institutions (AVTOR School, AVTOR College , AVTOR Art College , AVTOR High School , AVTOR High School Semestric, AVTOR M High School Semestric, AVTOR Educational Centres, AVTOR High School Pro).

Possibilities of scheduling module for universities:

- Accounting for the appointment and capacity of auditoriums.
- Formation of the document "Passport of the auditorium".
- Scheduling of studying sessions in conditions of deficiency of the auditorium fund.
- Making the schedule, taking into account the strict sequence of studying the discipline by the type or theme of the lesson.
- Formation of "complex" flows.
- Change settings for each group.

- Taking into consideration the preferences of teachers.

- Accounting the time of transportation between the remote bodies of the educational organization.

- Planning of non-academic classes for the group.

- Optimizing the "windows" of the employment of teachers, students or auditoriums.

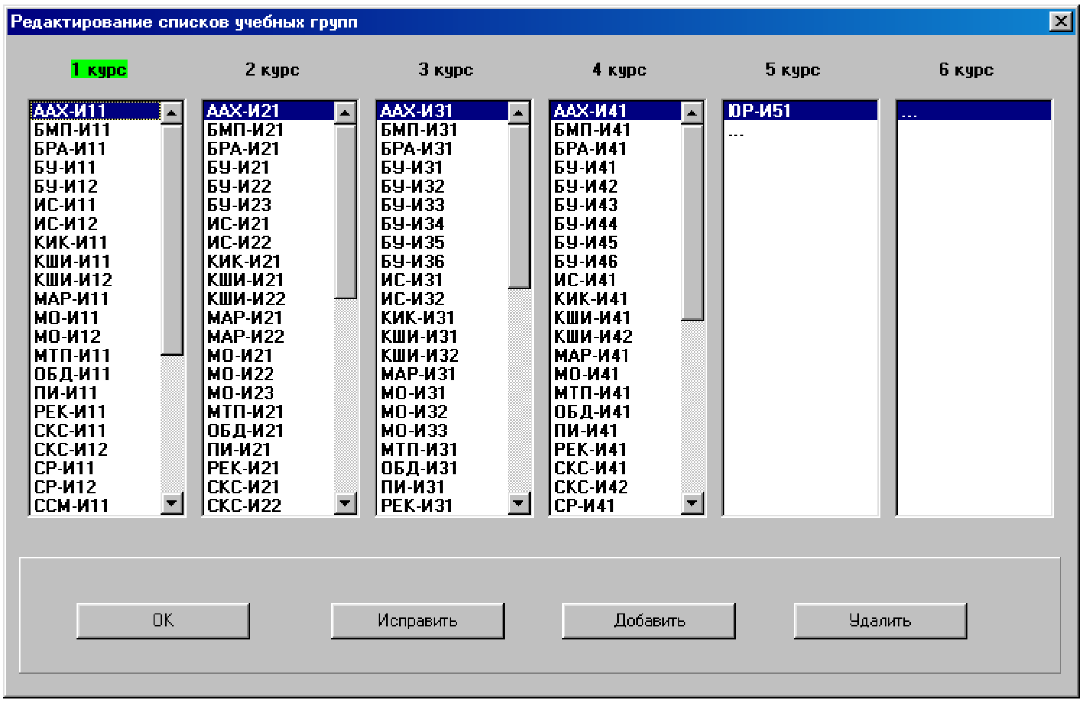
- Integration of the schedule with personal calendars Bitrix24, Microsoft Outlook, Google on smart phones of students and pedagogical workers.

- Generating analytical reports on the employment of teachers, auditoriums, and the quality of the timetable.

The scheduling is carried out in the following order: an acceptable schedule is automatically compiled, if the schedule is not compiled, then in the interactive mode it is possible to eliminate errors detected during the operation of the analysis of initial data block, and you can also eliminate errors in manual mode.

The sequence of work on scheduling is as follows:

1. Input (and adjustment) of lists (Fig. 1.8): studying groups, classrooms, teachers. academic subjects, departments of the university.



1 курс	2 курс	3 курс	4 курс	5 курс	6 курс
ААХ-И11	ААХ-И21	ААХ-И31	ААХ-И41	ЮР-И51	...
БМП-И11	БМП-И21	БМП-И31	БМП-И41	...	
БРА-И11	БРА-И21	БРА-И31	БРА-И41		
БУ-И11	БУ-И21	БУ-И31	БУ-И41		
БУ-И12	БУ-И22	БУ-И32	БУ-И42		
ИС-И11	БУ-И23	БУ-И33	БУ-И43		
ИС-И12	ИС-И21	БУ-И34	БУ-И44		
КИК-И11	ИС-И22	БУ-И35	БУ-И45		
КШИ-И11	КИК-И21	БУ-И36	БУ-И46		
КШИ-И12	КШИ-И21	ИС-И31	ИС-И41		
МАР-И11	КШИ-И22	ИС-И32	КИК-И41		
МО-И11	МАР-И21	КИК-И31	КШИ-И41		
МО-И12	МАР-И22	КШИ-И31	КШИ-И42		
МТП-И11	МО-И21	КШИ-И32	МАР-И41		
ОБД-И11	МО-И22	МАР-И31	МО-И41		
ПИ-И11	МО-И23	МО-И31	МТП-И41		
РЕК-И11	МТП-И21	МО-И32	ОБД-И41		
СКС-И11	ОБД-И21	МО-И33	ПИ-И41		
СКС-И12	ПИ-И21	МТП-И31	РЕК-И41		
СР-И11	РЕК-И21	ОБД-И31	СКС-И41		
СР-И12	СКС-И21	ПИ-И31	СКС-И42		
ССМ-И11	СКС-И22	РЕК-И31	СР-И41		

Figure 1.8: The lists of studying groups

2. Setting the schedule parameters:

- maximum length of the working day (in class hours);

- number of working days per week;

- call schedule and etc.

3. Setting up requirements for audiences

4. Configure of characteristics of classes types

5. Input and control of source data:

5.1. By classes (characteristics and features):

5.2. By lecturers (priorities and preferences);

5.3. By academic groups (curriculum and their features).

6. Arrangement of hard-planned classes (using the "Schedule Editor").

7. Checking the correctness of the source data.

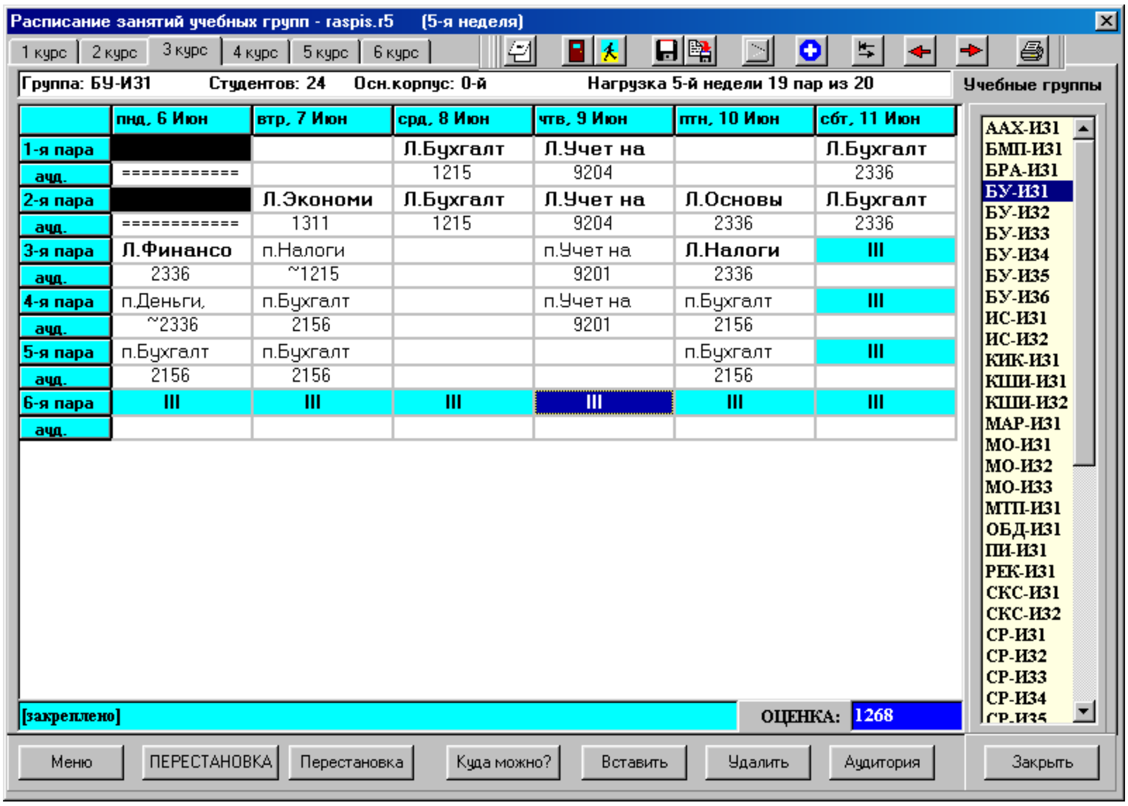
To do this, the Menu option "Schedule / Analysis of source data" for studying groups, for teachers and for auditoriums is used.

8. Building a schedule in automatic mode (Fig.1.10):

At this stage, the completed version of the schedule will be received. Now one can see it and (if necessary) modify it in the Schedule Editor mode.

9. Correcting the schedule in the mode of automated editing.

10. Displaying the schedule in the document files for printing [9]:



Расписание занятий учебных групп - raspis.r5 (5-я неделя)

Группа: БУ-И31 Студентов: 24 Осн.корпус: 0-й Нагрузка 5-й недели 19 пар из 20

	пнд, 6 Июн	втр, 7 Июн	срд, 8 Июн	чтв, 9 Июн	птн, 10 Июн	сбт, 11 Июн
1-я пара			Л.Бухгалт	Л.Учет на		Л.Бухгалт
ауд.	============		1215	9204		2336
2-я пара		Л.Экономи	Л.Бухгалт	Л.Учет на	Л.Основы	Л.Бухгалт
ауд.	============	1311	1215	9204	2336	2336
3-я пара	Л.Финансо	п.Налоги		п.Учет на	Л.Налоги	III
ауд.	2336	~1215		9201	2336	
4-я пара	п.Деньги,	п.Бухгалт		п.Учет на	п.Бухгалт	III
ауд.	~2336	2156		9201	2156	
5-я пара	п.Бухгалт	п.Бухгалт			п.Бухгалт	III
ауд.	2156	2156			2156	
6-я пара	III	III	III	III	III	III
ауд.						

[закреплено] ОЦЕНКА: 1268

Fig.1.10: Academic groups schedule

The main advantages of the program are:

- compactness of system files and the ability to work in large educational institutions with complex schedules;

- the system allows one to create a new schedule, quickly adjust, save and print various scheduling options, modify them during the academic year;

- availability of detailed statistics and objective evaluation of the quality of the schedule option;

- the ability to support any national language (at the request of the client).

For the successful operation of the system, the following minimum computer configuration is required: a processor of at least Pentium, RAM not less than 16Mb, WINDOWS 98 operating system, screen mode: 800x600 points (or more). To work with and to store the database, as well as to fully output one version of the schedule requires about 5Mb of free space on the hard disk (hard drive) [9].

“1C: Automated scheduling. University”

The product is designed to solve the tasks of automated scheduling and operational management of premises in universities. Provides the ability to schedule in automatic, manual and mixed modes, taking into account many constraints and conditions. In this case, you can build both a valid schedule and an optimized one, in which the number of spare hours or the number of occupied classrooms is reduced [10].

The program implements the following functions:

- scheduling in modes (manual, automatic, mixed);

- integrated means of integration with "1C: University", "1C: University PROF";

- a convenient form for quick manual modification of the schedule by dragging and dropping "drap & drop";

- making several schedules and choosing the best;

-scheduling in terms of semesters, scenarios (pessimistic, optimistic), departments, type of timetable, the period on which the schedule is composed; taking into account the preferences and opportunities of teachers, student groups, premises; consolidation of schedules and much more [10].

The process of creating a schedule in the system is implemented as follows:

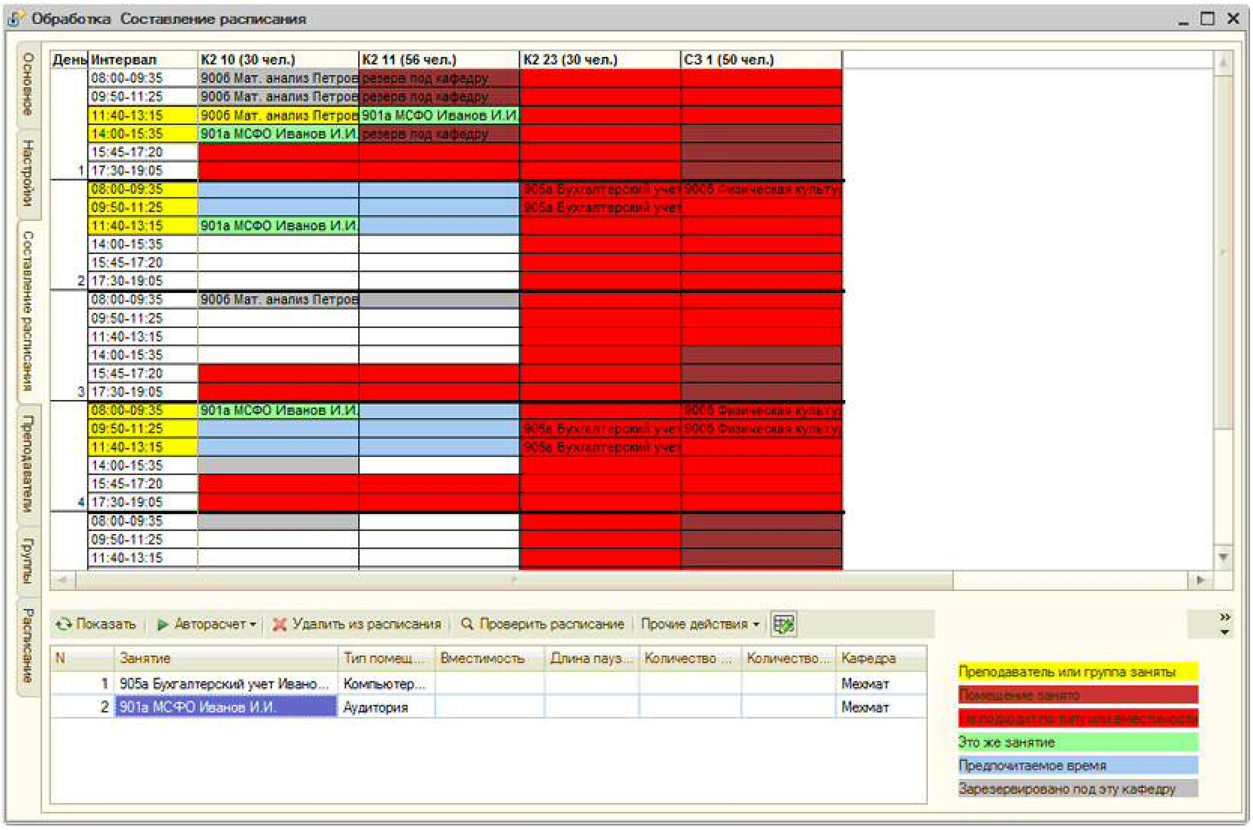
Stage 1 - Input of primary information: groups, disciplines, teachers, auditoriums;

Stage 2 - Entering data on the curriculum for a semester (what teacher, for which group, for what class and for how long the lesson is conducted).

Stage 3 - Entering constraints and preferences for lecturers, students, classrooms;

Stage 4 – Formation of the curriculum.

Figure 1.12 shows the main form for scheduling ("chess", the rows - the days of the week, lessons, columns - rooms).



День	Интервал	К2 10 (30 чел.)	К2 11 (56 чел.)	К2 23 (30 чел.)	С3 1 (50 чел.)
1	08:00-09:35	9006 Мат. анализ Петров	резерв под кафедру		
	09:50-11:25	9006 Мат. анализ Петров	резерв под кафедру		
	11:40-13:15	9006 Мат. анализ Петров	901а МСФО Иванов И.И.		
	14:00-15:35	901а МСФО Иванов И.И.	резерв под кафедру		
	15:45-17:20				
	17:30-19:05				
2	08:00-09:35			905а Бухгалтерский учет	9006 Физическая культу
	09:50-11:25			905а Бухгалтерский учет	
	11:40-13:15	901а МСФО Иванов И.И.			
	14:00-15:35				
	15:45-17:20				
	17:30-19:05				
3	08:00-09:35	9006 Мат. анализ Петров			
	09:50-11:25				
	11:40-13:15				
	14:00-15:35				
	15:45-17:20				
	17:30-19:05				
4	08:00-09:35	901а МСФО Иванов И.И.			9006 Физическая культу
	09:50-11:25			905а Бухгалтерский учет	9006 Физическая культу
	11:40-13:15			905а Бухгалтерский учет	
	14:00-15:35				
	15:45-17:20				
	17:30-19:05				
	08:00-09:35				
	09:50-11:25				
	11:40-13:15				

N	Занятие	Тип помещ...	Вместимость	Длина пауз...	Количество ...	Количество...	Кафедра
1	905а Бухгалтерский учет Ивано...	Компьютер...					Мехмат
2	901а МСФО Иванов И.И.	Аудитория					Мехмат

Преподаватель или группа заняты

Помещение занято

Не подходит по типу или вместимости

Это же занятие

Предпочитаемое время

Зарезервировано под эту кафедру

Figure 1.12: Scheduling. Chess.

The screen displays the classes that have been put into the schedule, the cells are highlighted with a special color, where the lesson can or cannot be set from the lower list of "unsettled" classes.

Yellow color - the lecturer or group is busy at this time;

Fire-brick color - the room is occupied (reserved by another department);

Red color - the room does not fit in type or capacity;

Green - the same activity;

Blue - preferred time for studying (from the point of view of the teacher or students);

Gray - the room is reserved for this department [10].

The user can manually drag classes from the lower list to a chess table or move classes on it. Thus, after calculating the chess automatically, it is possible to change it manually, or, conversely, put something manually, the balance can be calculated automatically.

The system automatically shows all collisions and non-observance of conditions and allows one to work them quickly through. If there are errors in the schedule, they will be displayed on the right side of the screen with a description.

The system requirements for the program are Database managing system - MS SQL, IBM DB2, Oracle Database. Operating systems - Windows, Linux, Mac OS, iOS [10].

All considered software products are not free. The prices for product licenses at the time of 2018 with installation for 1 computer are as follows:

The program "Nika-College" - 12000 rub.

The system “AVTORaspisanie” - 80000 rub.

The program "Rector-University"– 8000 rub.

The program “1C: Automated scheduling. University” - 70000 rub.

The program "Nika-College" and the program "Rector-University" are relatively simple and less capacious. They do not use the Database managing system, which, on the one hand, reduces the price of the product, but on the other hand, it affects the maintenance of reference information. Programs have a small number of settings and constraints, allow you to search for available schedules, without the possibility of optimization. Self-introduction of changes is impossible. However, comparing with other products, these programs are much cheaper and easier to use.

The system “AVTORaspisanie” and “1C: Automated scheduling. University” provide the ability to put a large number of constraints and scheduling requirements. Systems are well scalable and implement functionality in both automatic and manual modes. The systems are more voluminous; they consist of subsystems and are much more expensive.

1.5 The review of existing algorithms for solving tasks of automatic scheduling

The following algorithms were considered:

- Genetic algorithm
- Methods of integer programming
- Method of graph theory
- Simulated annealing algorithm

Genetic algorithm

This algorithm consists of the following stages:

1. Formation of the initial population

Randomly, the initial population consisting of N individuals (schedule options) is formed.

2. Selection of individuals

Selection of the most adapted individuals (schedule options) with more preferable values of suitability functions, in comparison with the rest.

3. Crossing-over

Crossing. On the basis of crossing chromosomes of parents, chromosomes of descendants are created.

4. Mutation

To some of the individuals after crossing, the mutation operator is applied. The mutation introduces additional diversity into the current population and thereby expands the search space for the optimal solution.

5. The selection operator

The selection operator performs the functions of a filtering tool, which selects individuals in the population that have a low suitability function value. The selected found weak individuals are removed from the population until the number of the population becomes the same as in the beginning.

6. Checking the termination conditions of the algorithm

A new generation, formed as a result of application of selection, crossing, mutation operators, called a population of descendants, replaces the parent population, after which the condition for terminating the algorithm is checked. It is based on using the increment of the suitability function if, for several generations, the increment in the value of the suitability function of the "best" individual turns out to be insignificant, then the algorithm's work ends. If the termination condition given in the algorithm is fulfilled, the next step is made, otherwise the selection process is completed and the process of finding the optimal solution continues.

7. Choosing the best solution

At this stage among the received individuals the best individual is selected, who will be the solution of the problem. The best individual is the one with the minimum suitability function value. [11].

Disadvantages:

- the quality of the solution depends significantly on the size and diversity of the initial population;

- solutions obtained as a result of several experiments for the same problem may vary slightly;

- a weak specifics account of the task of studying sessions schedule when organizing its solution.

Benefits:

- the algorithm works with codes that represent a set of parameters that directly depend on the arguments of the objective function;

- in the process of searching, the algorithm uses several points of the search space, and does not go from point to point;
- in the process of operating, the algorithm may not use any additional information about the problem, but if there is such, the algorithm convergence can be accelerated;
- the genetic algorithm uses both probability rules for generating new search points, and deterministic for the transition from one point to another;
- for high-dimensional tasks, the speed of the algorithm can be "regulated" by the size of the population [12].

Methods of integer programming

The problem of integer programming is reduced to identifying variables which meaning is to be found, compiling a mathematical model of the problem in the form of constraints that describe the problem and impose certain constraints on the desired variables, and the composing of the objective function.

Consider the most general statement of the integer programming problem for scheduling.

There are q objects, presented as

Every object contain lectures

There are *r* curricula . Each curriculum defines the list of subjects that group study. This means that subjects should take place in different time.

If the number of study periods $-p$, there is a maximum number of sessions that should be planned for the lecture period *k* (the number of classrooms available in the time period *k*). Then the formulation of the problem is as follows:

The application of the algorithm consists of the steps:

- selection of variables;
- compiling a mathematical model (defining constraints for variables);
- formulation of the objective function;
- finding the maximum (minimum) of the objective function using mathematical methods.

It is needed to find

$$i=1..q;\quad k=1..p$$

With constraints:

Where , if lecture of subject is appointed on time *k* and in the opposite case.

Constraint (1) determines that each subject has a correct number of lectures (classes).

Constraint (2) provides that at any given time there cannot be more classes than the classrooms.

Constraint (3) prevents the conflict of lectures (classes) at the same time.

Constraint (4) contains a binary requirement that turns a linear programming problem into a linear / integer programming problem.

Disadvantages of the algorithm:

- exponential increase in time spent on finding a better (acceptable) solution with increasing dimensionality of the problem being solved;
- lack of guarantee of an acceptable solution;
- because of the large dimension of the mathematical model, it is difficult to evaluate the influence of various factors on the process of solving the problem and its result;
- complexity of preferences accounting.

Method of graph theory

In this case, an undirected graph is constructed, in which each vertex is a lesson planned in the curriculum. If there are conflicts between any two vertices, then they are connected by an edge. This is equivalent to the prohibition of simultaneous conducting classes. Then the problem is reduced to coloring the graph in a given number of colors.

Stages of the algorithm:

1. Defining a variety of activities in the curriculum.
2. Presenting each lesson in the form of a graph vertex.
3. Joining the vertices of the graph with edges when it is not possible to conduct classes at the same time.
4. Solving the task of coloring the graph into a given number of colors.

Disadvantages:

- small efficiency when applying exact methods for coloring graphs of large dimension;

- lack of ability to take preferences into consideration.

Benefits:

- a simple mathematical model;
- application of the graph together with heuristic methods can give good results.

Simulated annealing algorithm

Simulated annealing algorithm is based on imitation of the physical process, which happens while crystallization of a substance from a liquid state to a solid.

Stages of the algorithm:

1. compose the correct schedule and set a high temperature value T = T0;
2. change the schedule Z = Z';
3. calculate the objective function for the changed schedule Δ = f(Z') – f(Z);
4. replace the previous schedule with the received one if it is better than the previous one (Δ ≤ 0), if not, then the probability of replacing is p = e–Δf/T;
5. lower the temperature;
6. until the termination criteria is passed, move to step 2

The main disadvantages are:

- ineffective for compiling schedules in modern mass education systems because of the large dimension of the problem;
- in order to obtain an effective solution, it is necessary to apply the Boltzmann or Cauchy scheme, which leads to considerable expenditures of computing power.

Conclusion of the chapter

In this chapter, we turned to the origins and the background of the Theory of Scheduling. Defined the main concepts and tasks that are studied in this area. We examined the formulation of the task of scheduling in educational institutions. We investigated what questions are raises in the task of automating the formation of the timetable and what are the main approaches. The paper deals with the most frequently encountered in the literature algorithms for solving problems in this field. The review and analysis of several popular programs on the software market, their functionality, advantages and disadvantages are explored in this part.

Chapter 3

Timetable scheduling

2.1 The procedure for scheduling in manual mode

Before beginning to work, the Institute's data sheets are compiled and prepared, namely:

- list of groups
- group curricula
- list of classrooms
- list of lecturers
- workload of lecturers

The schedule is compiled for a period - semester.

At the first stage, the flow classes of Physical Education and Foreign Language are put in the schedule. They are fixed. This is due to the fact that classes for groups are distributed in advance in the Department of Physical Education and at the Department of Foreign Languages. Then this schedule of classes is sent to the university institutes.

The scheduling for groups begins with the first courses (1-3). Their classroom classes are set, mainly in the morning. Classes for senior courses, magistracies, depending on the free hours, are put in the second shift.

When appointing teachers for practical or lecture classes one is guided by the document "lecturer load", which contains a list of subjects taught by one or another lecturer, the number of working hours of the lecturer. The qualification of the lecturer is taken into account regarding the possibility of conducting streaming lecture classes.

When classrooms are distributed between studying groups (the list of the university's classrooms and their capacity is organized in advance), the number of people in the group and the capacity of the classroom into which the group is put are taken into account. If the number of students in the group is more than 24 people, lesson in the computer class is carried out by dividing the group into subgroups.

For each group, the number of classes per week of a subject is calculated as follows. The number of classroom hours of the studying group (according to the curriculum) is divided by the number of weeks in the semester. The received number is divided by 2 (as the length of the lesson is taken into account).

After classes for certain groups are assigned to specific auditoriums and teachers are assigned, a checking is made for the presence of overlays in the timetable (one teacher is assigned to two different, non-stream groups at the same time, or the same audience is occupied by two or more groups at the same time).

If the auditorium fund of the university is not enough to distribute the classes for all study groups, a request is made to the academic department of the university on the topic of the presence of free classrooms.

2.2 Requirements and system constraints

In the course of scheduling, it is necessary to take into account a number of hard and soft (additional) constraints.

Hard constraints:

- Accordance of the number of seats in the auditorium to the number of people in the group. The capacity of audiences should be sufficient for the groups that are engaged in it. It is possible that in one classroom there is a lesson for several groups simultaneously (flow).
- Conformity of the auditorium to the type of class. Laboratory classes should be conducted in laboratory classrooms, lectures in lecture rooms.
- Failure of overlay. One teacher should not be assigned to different groups at the same time (if not a stream lecture). The same audience should not be occupied by two or more groups at the same time.
- Carrying out the full amount of lessons. The required number of hours of certain subjects for study groups should be taken into account (according to the curriculum).
- Taking into account the working hours of teachers according to their workload.
- Constraint on the amount of daily lessons.

Soft constraints:

- No repetition of lecture or practical lessons of the same subject within one day.
- Consideration of preferences for part-time teachers.
- Even distribution of lecture classes or practical ones on the same subject in time with the same periodicity.
- Compactness of teachers' workload.
- Absence of "windows" of studying groups.

2.3 Mathematical formulation of the problem

The construction of a mathematical model is based on the terms of linear programming.

2.3.1 Notations

The following notations are introduced:

N – The number of studying groups

k_r – The number of the group in the flow r ($k_r = 1, \ldots, G_r$)

R – The number of studying flow

r – Flow number (r = 1, ..., R). Groups of the same flow use the same auditorium fund for lecture classes.

T – Multiple studying days for groups and flows

t – the number of the school day of the week ($t \in T$)

J – Total number of classes

j – the **number of classes ($j = 1, \ldots, J$)**

Each group has a curriculum, which indicates the number of lecture, practical classes for a particular subject during the semester. To formulate the problem, we used a variable defining

W_{kr} – the number of auditorium classes, held with studying group k_r flow r during a week

These classes include lecture classes and seminars (practices):

S_r - the number of lecture classes of flow *r*

S_{sr} – the number of lecture classes of subject *s* in a week

s_r – the number of subject in the list for the flow r ($s_r = 1, \ldots, S_{sr}$)

Q_{kr} – the number of practical classes of the group k_r

Q_{skr} – the number of practical classes of subject *s* group k_r during a week

q_{kr} – the number of subject in the list for the group k_r ($q_{kr} = 1, \ldots, Q_{skr}$)

There is a list of lecturers:

p – **Number (name) of the lecturer (p = 1,…, P).**

Boolean variables $\delta^p_{rs_r}$ and $\Delta^p_{rk_rq_{kr}}$ define the following:

$$\delta^p_{rs_r} = \begin{cases} 1, & \text{if in the flow r lecture class } s_r \text{ is held by lecturer p;} \\ 0 & \text{– in the opposite case;} \end{cases}$$

$$\Delta^p_{rk_rq_{kr}} = \begin{cases} 1, & \text{if the group } k_r \text{ seminar } q_{kr} \text{ is held by lecturer p;} \\ 0 & \text{– in the opposite case;} \end{cases}$$

As the load of lecturers is formed before scheduling, $\delta^{p}_{rs_r}$ and $\Delta^{p}_{rk_r q_{kr}}$ are assumed to be given.

The lecturer's load also determines the number of hours worked by the lecturer during the week:

N_p - **auditorium load of lecturer p (p = 1 ,…,P).**

The auditorium fund is formed by the time of scheduling and is determined by the following aggregates:

$\{A_l\}$ – Aggregate of auditoriums for lecture classes.

$\{A_p\}$ – Aggregate of auditoriums for seminars.

A_l – The number of elements of the aggregate $\{A_l\}$.

A_p – The number of elements of the aggregate $\{A_p\}$.

$A_l + A_p$ – the number of auditoriums of combined aggregates $\{A_l\} \cap \{A_p\}$.

The task of scheduling is the following: "Define the day of the week and the number of the class on that day for each lecture (in the flow) and practical lesson (in the group), taking into account the fulfillment of the constraints defined below and maximization of the objective function".

The following Boolean variables are introduced:

$$y^{s_r}_{rt_j} = \begin{cases} 1, & \text{if in the flow r in a day t class j there is a lecture } s_r; \\ 0 & \text{– in the opposite case;} \end{cases}$$

$$x^{q_{kr}}_{rk_r t_j} = \begin{cases} 1, & \text{if in the flow r in a day t class j group } k_r \text{ has seminar } q_{kr}; \\ 0 & \text{– in the opposite case;} \end{cases}$$

2.3.2 Constraints of the system

Constraint 1. For each group k_r all types of classroom work (according to the curriculum) must be performed during the week:

For each group, the auditorium load is carried out according to lectures and seminars of each subject:

$$\sum_{t \in T} \sum_{j=1}^{J} x^{q_{kr}}_{rkrtj} = Q_{skr} \qquad \forall r = 1,\dots, R; \quad \forall k_r = 1,\dots, G_r; \tag{1}$$

$$\sum_{t \in T} \sum_{j=1}^{J} y^{s_r}_{rtj} = S_{sr} \qquad \forall q_r = 1,\dots, Q_{kr}; \quad \forall s_r = 1,\dots, S_r.$$

Constraint 2. In any day t on every class j for each group k_r it is impossible to have more than one lesson:

$$\sum_{q_{kr}=1}^{Q_{kr}} x_{rk_r t_j}^{q_{kr}} + \sum_{s_r=1}^{S_r} y_{rt_j}^{s_r} \le 1 \quad \forall r = 1,\dots, R; \ \forall k_r = 1,\dots, G_r;$$

$$\forall t \in T; \quad \forall j = 1,\dots, J. \tag{2}$$

Constraint 3. Each lecture s_r and seminar q_{kr} of the definite subject respectively for every flow r and all groups k_r can be held not more than one time in any day *t*:

$$\sum_{t \in T} \sum_{j=1}^{J} \left(x_{rk_r t_j}^{q_{kr}} + y_{rt_j}^{s_r} \right) \le 1 \qquad \forall r = 1,\dots, R; \quad \forall k_r = 1,\dots, G_r;$$

$$\forall q_r = 1,\dots, Q_{kr}; \tag{3}$$

$$\forall s_r = 1,\dots, S_r.$$

Variables $x_{rk_r t_j}^{q_{kr}}$ и $y_{rt_j}^{s_r}$ connect the type of class and the time when it is held. Product of numbers $\Delta_{rk_r q_{kr}}^{p} \cdot x_{rk_r t_j}^{q_{kr}}$ and $\delta_{rs_r}^{p} \cdot y_{rt_j}^{s_r}$ connect the time of the lesson with the name of the lecturer who conducts it.

Constraint 4. In every day *t* and in each class *j* lecturer *p* can have no more than one lesson of one discipline on one flow or in one group:

$$\sum_{r=1}^{R} \left(\sum_{s_r=1}^{S_r} \delta_{rs_r}^{p} y_{rt_j}^{s_r} + \sum_{k_r=1}^{G_r} \sum_{q_{kr}=1}^{Q_{kr}} \Delta_{rk_r q_{kr}}^{p} x_{rk_r t_j}^{q_{kr}} \right) \le 1 \quad \forall t \in T;$$

$$\forall j = 1,.., J; \quad \forall p = 1,\dots, P. \tag{4}$$

Constraint 5. Every lecturer *p* in a week must have a definite number of auditorium classes (lectures, seminars) in accordance with lecturer workload:

$$\sum_{t \in T} \sum_{j=1}^{J} \sum_{r=1}^{R} \left(\sum_{s_r=1}^{S_r} \delta_{rs_r}^{p} \cdot y_{rt_j}^{s_r} \right) = N_{pl}$$

$$\sum_{t \in T} \sum_{j=1}^{J} \sum_{r=1}^{R} \left(\sum_{k_r=1}^{G_r} \sum_{q_{kr}}^{Q_{kr}} \Delta_{rk_r q_{kr}}^{p} \cdot x_{rk_r tj}^{q_{kr}} \right) = N_{pp} \qquad \forall p = 1,\dots, P; \tag{5}$$

Constraint 6. Each day, each class the number of lectures and seminar classes should not exceed the auditorium fund available in the university:

$$\sum_{r=1}^{R} \sum_{s_r=1}^{S_r} y_{rt_j}^{s_r} \le A_l.$$

$$\forall t \in T; \quad \forall j = 1,\dots, J. \tag{6}$$

$$\sum_{r=1}^{R}\sum_{k_r=1}^{G_r}\sum_{q_{kr}=1}^{Q_{kr}} x_{rk_rt_j}^{q_{kr}} \leq A_p. \qquad (7)$$

In addition, for all sets of intersecting aggregates {Al} and {Ap}, the following conditions must be fulfilled:

$$\sum_{s_{r_{i1}}=1}^{S_{r_{i1}}} y_{r_{i1}t_j}^{s_{r_{i1}}} + \ldots + \sum_{s_{r_{in}}=1}^{S_{r_{in}}} y_{r_{in}t_j}^{s_{r_{in}}} + \sum_{k_{r_{l1}}=1}^{G_{r_{l1}}} \sum_{q_{kr_{l1}}}^{Q_{kr_{l1}}} x_{r_{l1}k_{r_{l1}}t_j}^{q_{kr_{l1}}} + \ldots +$$

$$+ \sum_{k_{r_{lm}}=1}^{G_{r_{lm}}} \sum_{q_{kr_{lm}}}^{Q_{kr_{lm}}} x_{r_{lm}k_{r_{lm}}t_j}^{q_{kr_{lm}}} \leq A_{l_{i1}} + \ldots + A_{l_{in}} + A_{p_{l1}} + \ldots + A_{p_{lm}}. \qquad (8)$$

Constraint 9. The number of classes for students of each group should not exceed the standardized number of hours (the number of classes) during the study day.

$$\sum_{j=1}\Big(\sum_{q_{kr}=1}^{Q_{kr}} x_{rk_rt_j}^{q_{kr}} + \sum_{s_r=1}^{S_r} y_{rt_j}^{s_r} \Big) \leq C. \qquad \forall r = 1,\ldots, R; \quad \forall k_r = 1,\ldots, G_r. \qquad (9)$$

$$\forall t \in T;$$

2.3.3 The objective function

The objective function can maximize or minimize different values. Consider the following statement of the objective function.

The objective function is to maximize the lecturer's auditorium load in the days when they have it. Weight factors are introduced, through which the relevant lecturer status should be taken into account. In some cases, based on expert estimates, individual weights can be used taking into consideration other factors.

The choice of the criterion of the quality of scheduling is reduced to maximizing the weighted number of days off from classroom work for all lecturers that under the condition of a fixed length of the working week, is equivalent to the maximum aggregate compaction of the auditorium load.

Formula for the auditorium load per day t for the lecturer p:

$$Q_t^p = \sum_{r=1}^{R}\left(\sum_{s_r=1}^{S_r} \delta_{rs_r}^{p}\, y_{rt_j}^{s_r} + \sum_{k_r=1}^{G_r}\sum_{q_{kr}=1}^{Q_{kr}} \Delta_{rk_rq_{kr}}^{p}\, x_{rk_rt_j}^{q_{kr}} \right). \qquad (10)$$

The following constraints are introduced:

$$1 \leq Q_t^p + Mz_t^p \leq M \quad \forall t \in T_{kr}; \quad \forall p = 1,\ldots, P, \qquad (11)$$

where M –random positive sufficiently large number; z_t^p - the desired Boolean variable. From (10) follows that if $Q_t^p = 0$, then z_t^p = 1, and if $Q_t^p > 0$, then z_t^p = 0.

Taking into account the optimization criterion in additional constraints (10), and also introducing weight factors of the lecturer's status Ω_p, we obtain the required optimal criterion:

$$\sum_{t\in T_{kr}}\sum_{p=1}^{P}\Omega_p z_t^p \to \max. \tag{12}$$

Approach the formulation of the objective function from the other side. It is necessary to distribute group lectures and practices for a week in an equable way. That is, try to ensure the number of classes every day for each group was the same.

Minimize the following function:

$$\sum_{t\in T}\left(\sum_{j=1}^{J}\left(\sum_{q_{kr}}^{Q_{kr}} x_{rk_r tj}^{q_{kr}} + \sum_{s_{r=1}}^{S_r} y_{rtj}^{s_r}\right) - \sum_{j=1}^{J}\left(\sum_{q_{kr}}^{Q_{kr}} x_{rk_r(t-1)j}^{q_{kr}} + \sum_{s_r=1}^{S_r} y_{r(t-1)j}^{s_r}\right)\right) \to \min \tag{13}$$

$$\forall r = 1,\dots,R; \quad \forall k_r = 1,\dots,G_r.$$

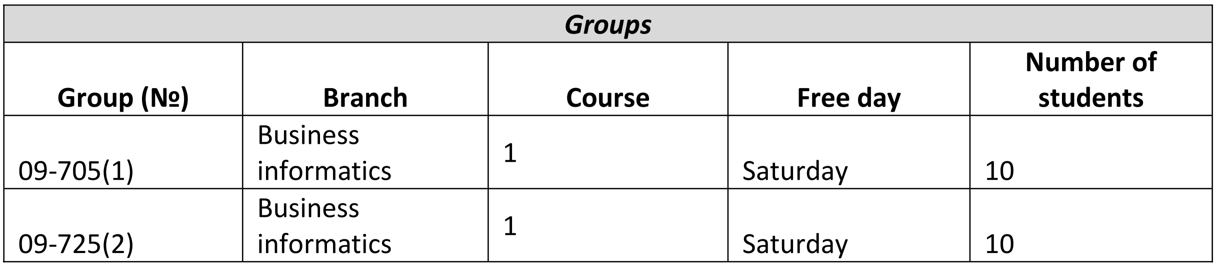
In the process of studying the procedure of manual scheduling, it was revealed that there are streaming exercises that are distributed first. In this regard, we can specify the objective function that maximizes the value of the variables responsible for the predefined time of classes of the streams.

2.4 Models of the scheduling with various problem formulation variations

2.4.1 Timetable 1

The first model illustrates scheduling for the next set of groups, teachers and audiences (Fig. 2.1, 2.2, 2.3).

Groups that are part of the same flow:



Groups				
Group (№)	**Branch**	**Course**	**Free day**	**Number of students**
09-705(1)	Business informatics	1	Saturday	10
09-725(2)	Business informatics	1	Saturday	10

Figure 2.1: Groups

Teachers			
№	**Full name**	**Taught disciplines**	**Department**
1	Tumakov D.N.	Combinatorial optimisation	Department of Applied Mathematics
2	Бухтоярова Л.В.	Physical Education	Department of Physical Education
3	Ivanova E.E.	Foreign Language	Department of Foreign Languages

Figure 2.2: Teachers

Classes			
Class (№)	**Type**	**Capacity**	**Building**
216	lecture	80	1
200	lecture	80	1
1008	practice	20	2

Figure 2.3: Classes

The scheduling formulation used the curriculum of the groups and the teaching load of these teachers (Fig.2.4):

Lecturer load			
Full name	**Taught disciplines**	**Group (№)**	**Type**
Ivanova E.E.	Foreign Language	09-705 (1) 09-725(2)	l
Bukhtoyarova L.V.	Physical Education	09-705 (1) 09-725(2)	l
Tumakov D.N.	Combinatorial optimisation	09-705 (1)	p
Tumakov D.N.	Computer architecture and programming in C++	09-725(2)	p

Figure 2.4: Lecturer load

It should be noted that the subjects were distributed within 3 working days, each day has 2 identical in duration studying period.

Given the necessary set of parameters and the workload of groups and teachers, it was necessary to distribute 2 lecture classes at the flow, one practical class for the group (09-705) and 3 practical classes for the group (09-725).

Using the mathematical model described previously, the following constraints were set for this case:

1) The number of lecture classes of each subject at the flow was limited with the number according to the curriculum. (1)

The number of practical classes for each group of a particular subject was limited the number according to the curriculum.

2) For each group at the same time no more than 1 classes is conducted. (2)

3) Each teacher can not be assigned more than 1 classes, in more than 1 flow and 1 group, at a specific time. (3)

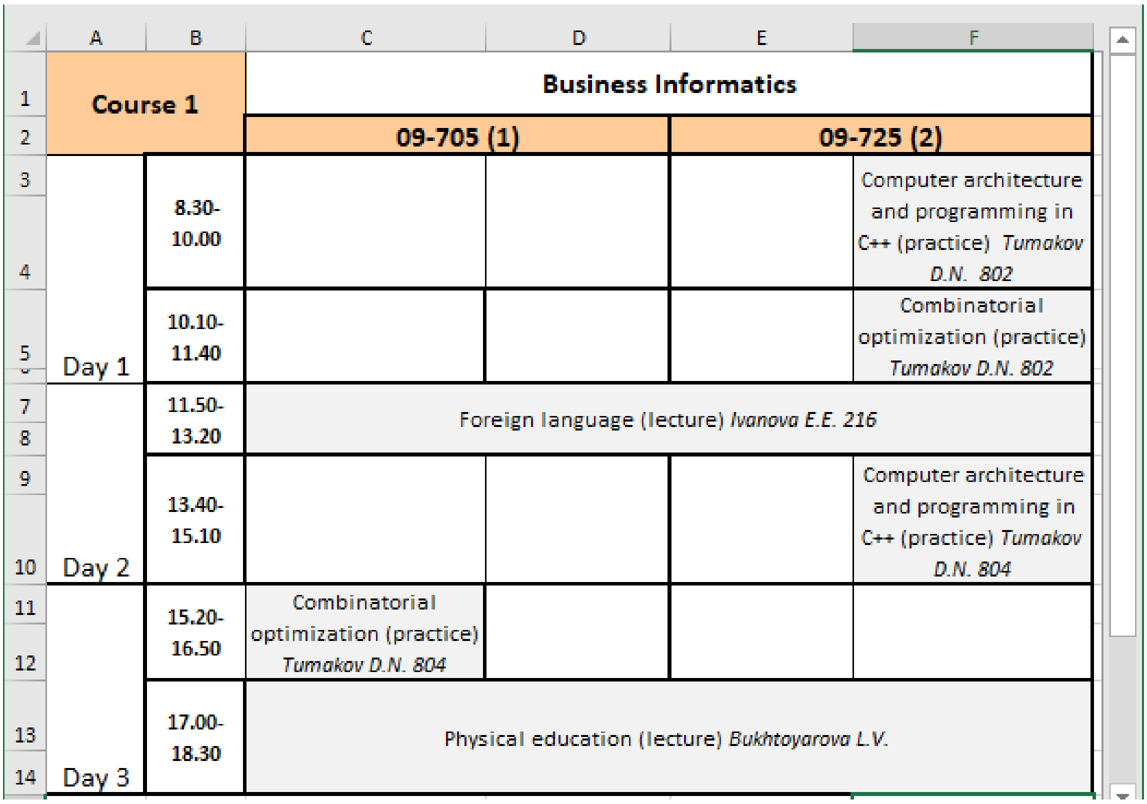
4) A certain class of a group can not be assigned no more than 1 time per day. (4)

5) Each teacher works the prescribed (for the lecturer load) number of classes. (5)

6) For the experiment, the number of lecture audiences ($Al = 1$) and practical audiences ($Ap = 2$) was limited. (6,7)

The objective function in this case was determined as follows. Since the time and place of the flow lectures (Foreign Language and Physical Education) at the institute are fixed before the formation of a full timetable, the variables responsible for carrying out these classes on day 2 period 1, and on day 3 last period were maximized.

As a result, the following distribution of classes was obtained. There were no overlaps for teachers and groups in the timetable for the time. The finished schedule can be seen in Fig. 2.5 in Excel format.



	A	B	C	D	E	F
1	Course 1		Business Informatics			
2			09-705 (1)		09-725 (2)	
3 4	Day 1	8.30-10.00				Computer architecture and programming in C++ (practice) *Tumakov D.N. 802*
5 6		10.10-11.40				Combinatorial optimization (practice) *Tumakov D.N. 802*
7 8	Day 2	11.50-13.20	Foreign language (lecture) *Ivanova E.E. 216*			
9 10		13.40-15.10				Computer architecture and programming in C++ (practice) *Tumakov D.N. 804*
11 12	Day 3	15.20-16.50	Combinatorial optimization (practice) *Tumakov D.N. 804*			
13 14		17.00-18.30	Physical education (lecture) *Bukhtoyarova L.V.*			

Figure 2.5: Timetable 1

2.4.2 Timetable 2

In the second case, the scheduling was complicated by the introduction of more flows number and a large number of classes for the groups (Fig. 2.6, 2.7, 2.8).

Constraints was formed on the basis of the mathematical formulation of the problem. All the strong conditions for the formation of the timetable were used, which assume an elimination the overlapping of the time for classes for the groups, the time of the teachers' work, the number of the classes and provide the necessary lecturer load and curriculum for the groups.

Groups				
Group (№)	**Branch**	**Course**	**Free day**	**Flow (№)**
09-705	Business informatics	1	Saturday	1
09-725	Business informatics	1	Saturday	2
09-735	Applied Mathematics	1	Friday	2

Figure 2.6: Groups

Teachers			
№	**Full name**	**Taught disciplines**	**Department**
1	Tumakov D.N.	Combinatorial optimisation	Department of Applied Mathematics

2	Бухтоярова Л.В.	Physical Education	Department of Physical Education
3	Ivanova E.E.	Foreign Language	Department of Foreign Languages

Figure 2.7: Teachers

Classes			
Class (№)	**Type**	**Capacity**	**Building**
216	lecture	80	2
1010	lecture	80	1
1006	practice	20	1

Figure 2.8: Classes

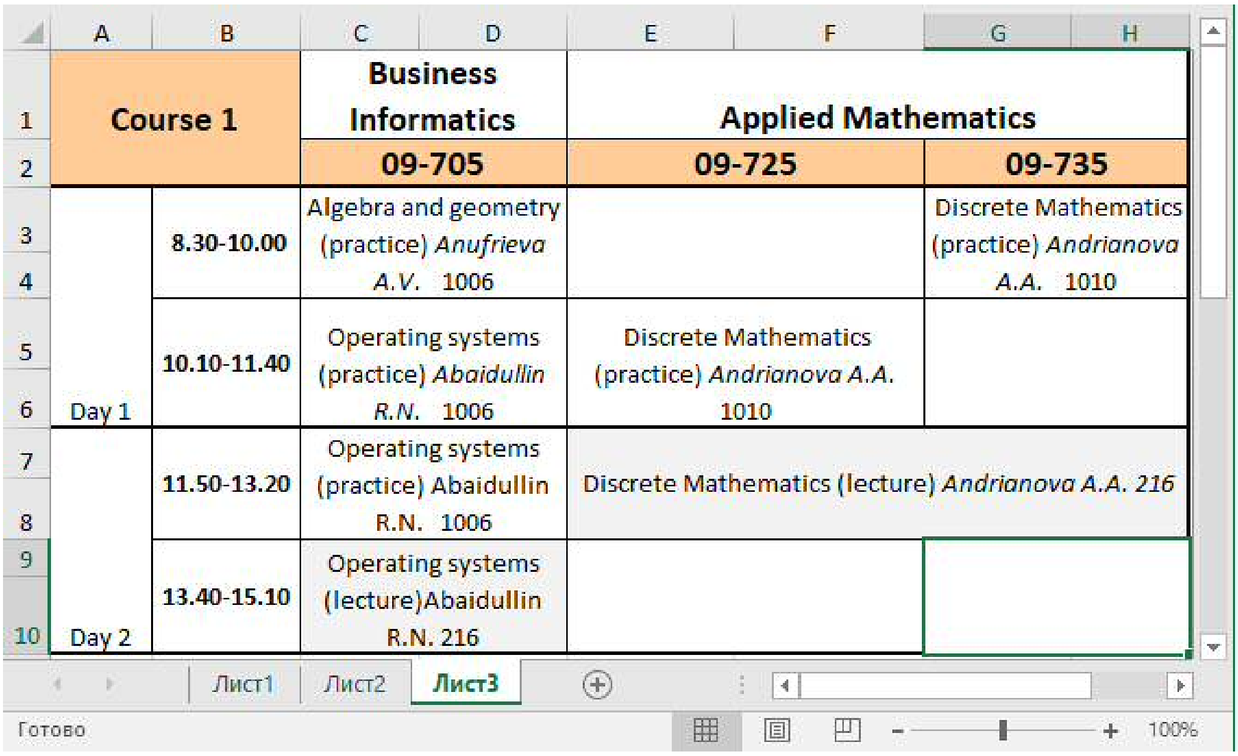
The scheduling formulation used the curriculum of the groups and the teaching load of these teachers (Fig.2.9):

Lecturer load			
Full name	**Taught disciplines**	**Group (№)**	**Type**
Anufrieva A.V.	Algebra and geometry	09-705	p
Abaidullin R.N.	Operating systems	09-705	l, p
Andrianova	Discrete Mathematics	09-725 09-735	l, p

Figure 2.9: Lecturer load

The objective function was determined in the way to make a bigger load on the beginning of the day (first period of the day). For this purpose we introduced weight coefficients for each period, in each day.

The result of scheduling for this case is illustrated in the Figure 2.10.



Course 1		Business Informatics	Applied Mathematics	
		09-705	09-725	09-735
Day 1	8.30-10.00	Algebra and geometry (practice) *Anufrieva A.V.* 1006		Discrete Mathematics (practice) *Andrianova A.A.* 1010
	10.10-11.40	Operating systems (practice) *Abaidullin R.N.* 1006	Discrete Mathematics (practice) *Andrianova A.A.* 1010	
Day 2	11.50-13.20	Operating systems (practice) Abaidullin R.N. 1006	Discrete Mathematics (lecture) *Andrianova A.A. 216*	
	13.40-15.10	Operating systems (lecture)Abaidullin R.N. 216		

Figure 2.10: Timetable 2

2.5 Validation of the timetable

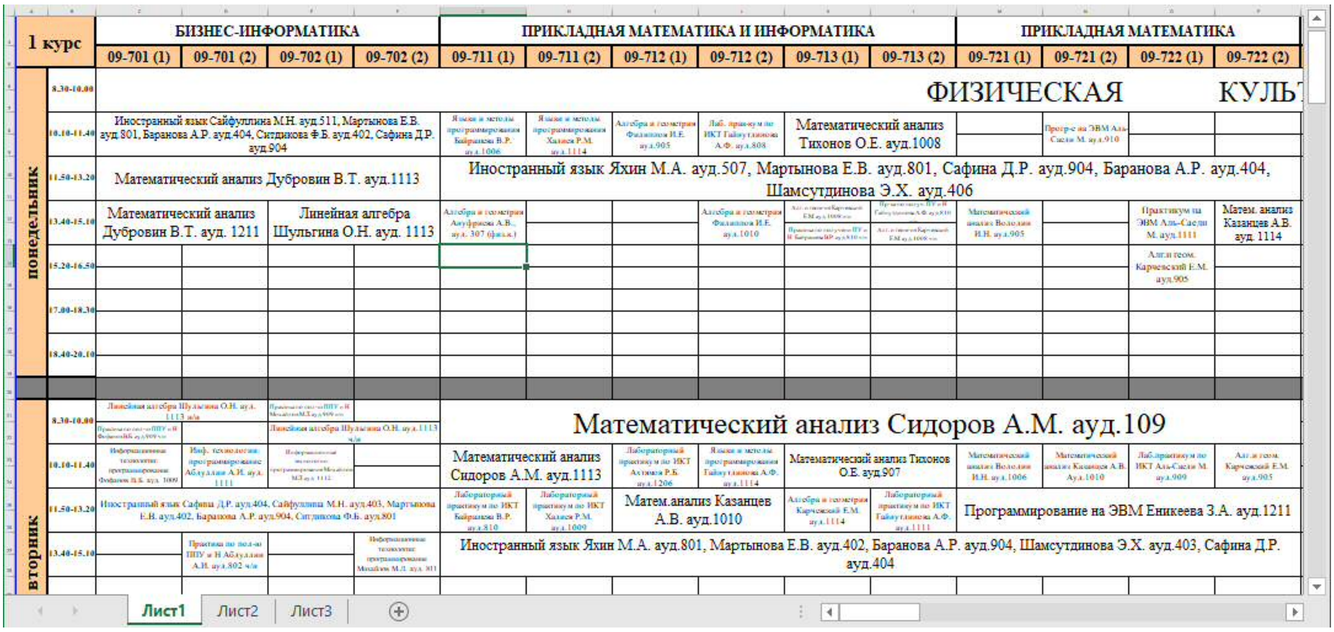
2.5.1 Description of the program

This part of the work describes the process of checking the schedule for errors.

The format of the schedule is defined as follows. The rows are divided according to the academic days of the week and the periods of the classes in each of the days. Columns define academic groups, which in turn are grouped into academic flows. The branches of studying and the number of the course of academic groups are also indicated.

Each cell contains the name of the class, the teacher's full name, the number of the room and the type of class (lecture or practical).

The timetable is formed in MS Excel format. Part of the institute's timetable can be seen in Fig.2.11.



1 курс		БИЗНЕС-ИНФОРМАТИКА				ПРИКЛАДНАЯ МАТЕМАТИКА И ИНФОРМАТИКА						ПРИКЛАДНАЯ МАТЕМАТИКА			
		09-701 (1)	09-701 (2)	09-702 (1)	09-702 (2)	09-711 (1)	09-711 (2)	09-712 (1)	09-712 (2)	09-713 (1)	09-713 (2)	09-721 (1)	09-721 (2)	09-722 (1)	09-722 (2)
понедельник	8.30-10.00	ФИЗИЧЕСКАЯ КУЛЬ													
	10.10-11.40	Иностранный язык Сайфуллина М.Н. ауд.511, Мартынова Е.В. ауд.801, Баранова А.Р. ауд.404, Ситдикова Ф.Б. ауд.402, Сафина Д.Р. ауд.904				Языки и методы программирования Байрашева В.Р. ауд.1006	Языки и методы программирования Хадиев Р.М. ауд.1114	Алгебра и геометрия Филиппов И.Е. ауд.905	Лаб. практикум по ИКТ Гайнутдинова А.Ф. ауд.808	Математический анализ Тихонов О.Е. ауд.1008			Прогр-е на ЭВМ Аль-Саели М. ауд.910		
	11.50-13.20	Математический анализ Дубровин В.Т. ауд.1113				Иностранный язык Яхин М.А. ауд.507, Мартынова Е.В. ауд.801, Сафина Д.Р. ауд.904, Баранова А.Р. ауд.404, Шамсутдинова Э.Х. ауд.406									
	13.40-15.10	Математический анализ Дубровин В.Т. ауд. 1211		Линейная алгебра Шульгина О.Н. ауд. 1113		Алгебра и геометрия Ануфриева А.В., ауд. 307 (физ.к.)			Алгебра и геометрия Филиппов И.Е. ауд.1010	Алг. и геом. Карчевский Е.М. ауд.1008 ч/н Практикум по изучению ПУ и Н Байрашева В.Р. ауд.810 н/н	Практикум по изучению ПУ и Н Гайнутдинова А.Ф. ауд.810 Алг. и геом. Карчевский Е.М. ауд.1008 ч/н	Математический анализ Володин И.Н. ауд.905		Практикум на ЭВМ Аль-Саели М. ауд.1111	Матем. анализ Казанцев А.В. ауд. 1114
	15.20-16.50													Алг.и геом. Карчевский Е.М. ауд.905	
	17.00-18.30														
	18.40-20.10														
вторник	8.30-10.00	Линейная алгебра Шульгина О.Н. ауд. 1113 н/н Практикум по изучению ПУ и Н Фофанов В.Б. ауд.909 ч/н		Практикум по изучению ПУ и Н Михайлов М.Л. ауд.909 н/н Линейная алгебра Шульгина О.Н. ауд.1113 ч/н		Математический анализ Сидоров А.М. ауд.109									
	10.10-11.40	Информационные технологии: программирование Фофанов В.Б. ауд. 1009	Инф. технологии: программирование Абдуллин А.И. ауд. 1111	Информационные технологии: программирование Михайлов М.Л. ауд. 1112		Математический анализ Сидоров А.М. ауд.1113		Лабораторный практикум по ИКТ Ахтямов Р.Б. ауд.1206	Языки и методы программирования Гайнутдинова А.Ф. ауд.1114	Математический анализ Тихонов О.Е. ауд.907		Математический анализ Володин И.Н. ауд.1006	Математический анализ Казанцев А.В. Ауд.1010	Лаб.практикум по ИКТ Аль-Саели М. ауд.909	Алг.и геом. Карчевский Е.М. ауд.905
	11.50-13.20	Иностранный язык Сафина Д.Р. ауд.404, Сайфуллина М.Н. ауд.403, Мартынова Е.В. ауд.402, Баранова А.Р. ауд.904, Ситдикова Ф.Б. ауд.801				Лабораторный практикум по ИКТ Байрашева В.Р. ауд.810	Лабораторный практикум по ИКТ Хадиев Р.М. ауд.1009	Матем.анализ Казанцев А.В. ауд.1010		Алгебра и геометрия Карчевский Е.М. ауд.1114	Лабораторный практикум по ИКТ Гайнутдинова А.Ф. ауд.1111	Программирование на ЭВМ Еникеева З.А. ауд.1211			
	13.40-15.10		Практика по пол-ю ППУ и Н Абдуллин А.И. ауд.802 ч/н		Информационные технологии: программирование Михайлов М.Л. ауд. 811	Иностранный язык Яхин М.А. ауд.801, Мартынова Е.В. ауд.402, Баранова А.Р. ауд.904, Шамсутдинова Э.Х. ауд.403, Сафина Д.Р. ауд.404									

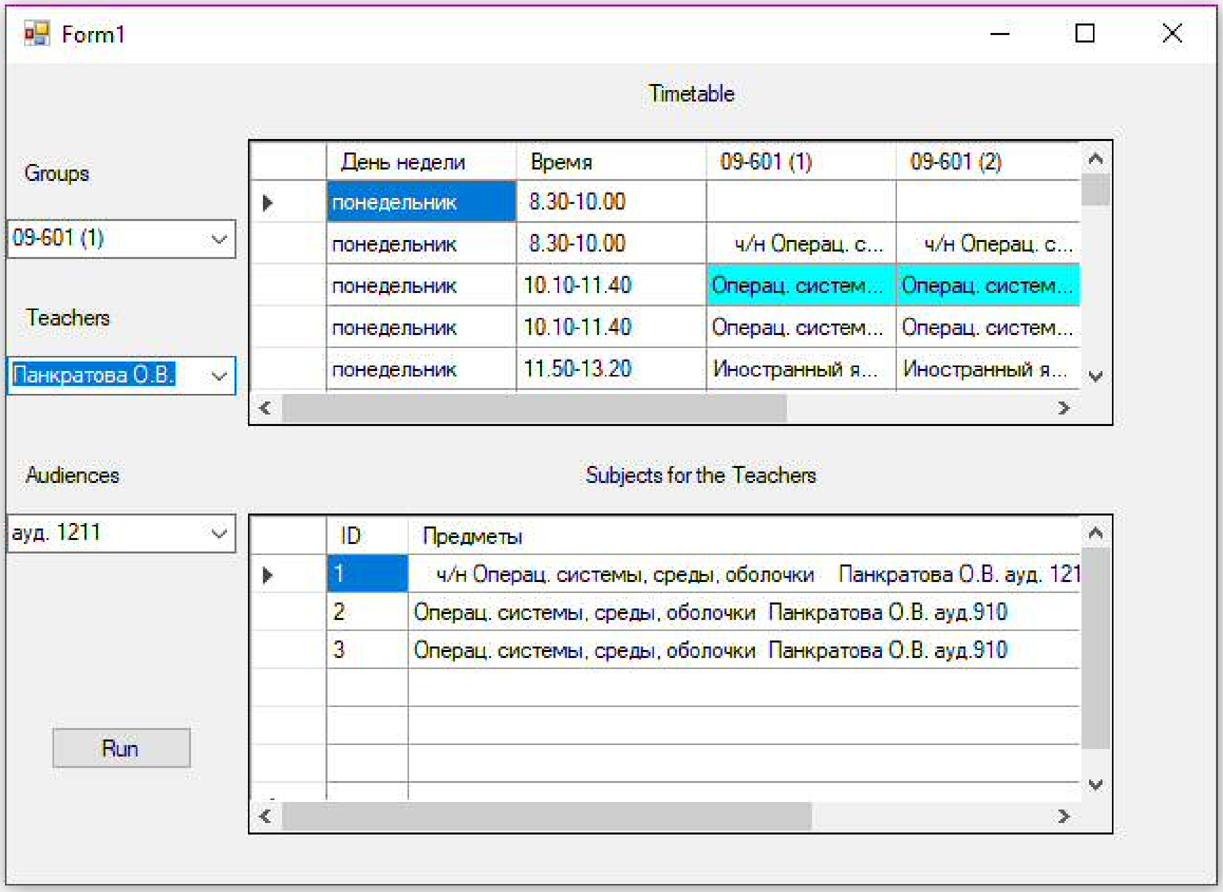
Figure 2.11: Institute timetable

Using the object-oriented programming language C # there was developed a program that allows user to check this timetable for the following points:

–Assigning a teacher in particular day, at particular time into two or more different groups (if this is not lecture on academic flow).

–Assigning a certain class in particular day, at the same time for two or more lecture or practice classes.

The program reads data from an Excel file, displays it in a simplified form using the Windows Form interface (Fig. 2.12). Also there are formed drop-down lists of groups, teachers and classes on the Windows form. By selecting a particular teacher from the drop-down list, you can see a list of items that he / she teaches according to the schedule.



Timetable

Groups: 09-601 (1)

Teachers: Панкратова О.В.

Audiences: ауд. 1211

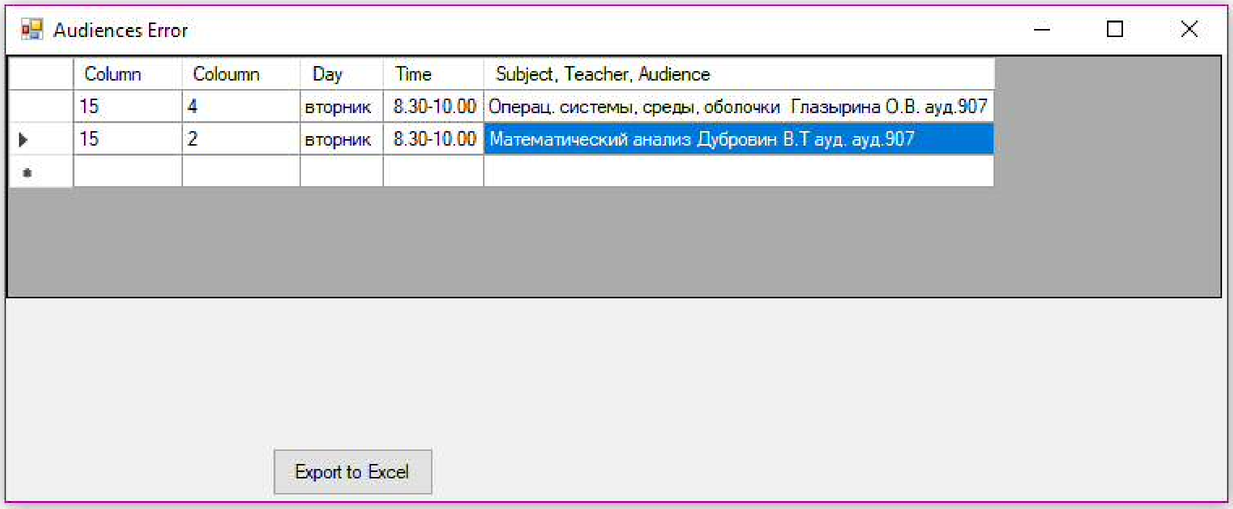
	День недели	Время	09-601 (1)	09-601 (2)
▶	понедельник	8.30-10.00		
	понедельник	8.30-10.00	ч/н Операц. с...	ч/н Операц. с...
	понедельник	10.10-11.40	Операц. систем...	Операц. систем...
	понедельник	10.10-11.40	Операц. систем...	Операц. систем...
	понедельник	11.50-13.20	Иностранный я...	Иностранный я...

Subjects for the Teachers

	ID	Предметы
▶	1	ч/н Операц. системы, среды, оболочки Панкратова О.В. ауд. 121
	2	Операц. системы, среды, оболочки Панкратова О.В. ауд.910
	3	Операц. системы, среды, оболочки Панкратова О.В. ауд.910

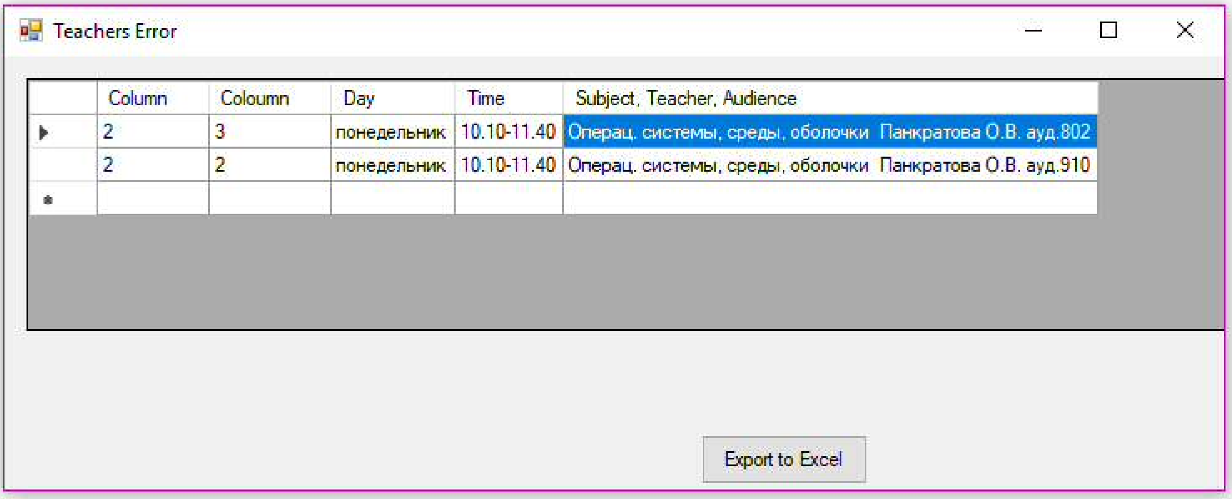
Figure 2.12: Interface of the program

After selecting the "Check" option a window that displays the time overlays for the practical and lecture clacces (Fig.2.13) and the window that displays the time overlays for the teachers (Fig.2.14) is created. That means one class or one lecturer is set in the timetable at the same time, which is unacceptable.



	Column	Coloumn	Day	Time	Subject, Teacher, Audience
	15	4	вторник	8.30-10.00	Операц. системы, среды, оболочки Глазырина О.В. ауд.907
▶	15	2	вторник	8.30-10.00	Математический анализ Дубровин В.Т ауд. ауд.907
*					

Figure 2.13: Audiences Error



	Column	Coloumn	Day	Time	Subject, Teacher, Audience
▶	2	3	понедельник	10.10-11.40	Операц. системы, среды, оболочки Панкратова О.В. ауд.802
	2	2	понедельник	10.10-11.40	Операц. системы, среды, оболочки Панкратова О.В. ауд.910
*					

Figure 2.14: Teachers Error

This program allows to export files in an MS Excel format.

This checking program allows to identify overlays in the timetable which generally occurs after its manual editing, making changes by the user (the dispatcher who is engaged in scheduling).

It should be noted that when creating the program, specific data structures were used. For example dictionaries are created inside the dictionary (to store data on teachers). Dictionary teachersWithStud consists of a key field - the name of the teacher and the dictionary enclosed in it, containing the subject field, which is maintained by the selected teacher. Filling is as follows:

teachersWithStud [surn] = new Dictionary <int, List <string >> ();

teachersWithStud [surn] [i] = new List <string> (); teachersWithStud [surn] [i] .Add (s);

During the program developing the integration technologies of C # with MS Excel which are described below were used.

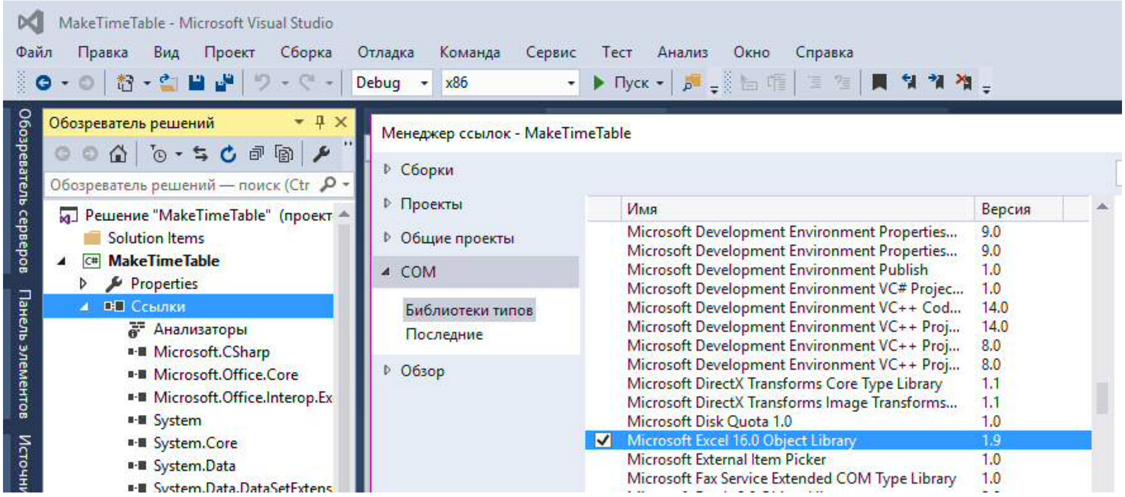
There are several mechanisms for integrating C # with Microsoft Excel.

To implement data retrieval from a Microsoft Excel file, can be used additional object libraries (Aspose Excel), use the OLE DB / ODBC interface set which allows applications to access data stored in different data sources or data stores, work through XML, through Open XML and etc.

For the Microsoft Excel integration the method of adding assembly reference from the COM category and connect the Microsoft Excel 16.0 Object Library (Fig.2.15) have been used in the work. After adding the library three new libraries appear in the Solution Explorer:

- Microsoft.Office.Core;
- Microsoft.Office.Interop.Excel;

- VBIDE.



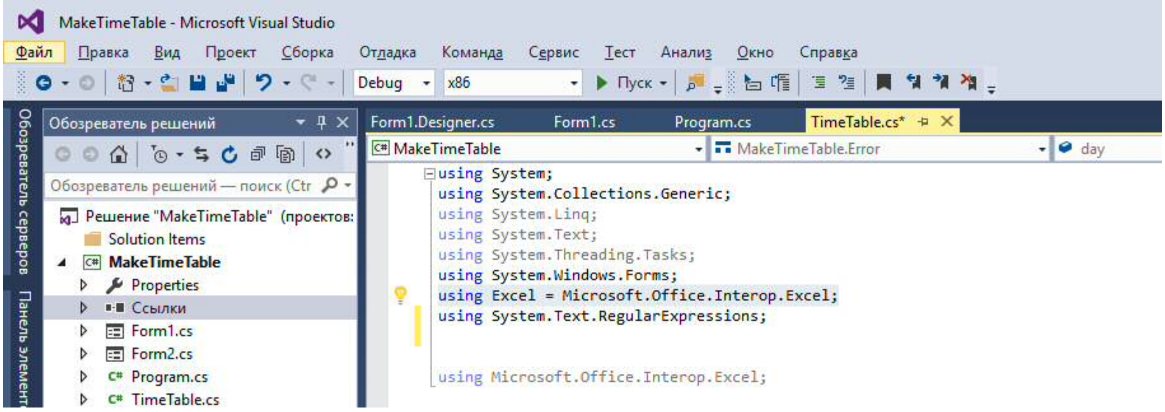
MakeTimeTable - Microsoft Visual Studio
Файл Правка Вид Проект Сборка Отладка Команда Сервис Тест Анализ Окно Справка
Debug
x86
Пуск
Обозреватель решений
Обозреватель решений — поиск (Ctr
Решение "MakeTimeTable" (проект
Solution Items
MakeTimeTable
Properties
Ссылки
Анализаторы
Microsoft.CSharp
Microsoft.Office.Core
Microsoft.Office.Interop.Ex
System
System.Core
System.Data
System.Data.DataSetExtens
Обозреватель серверов
Панель элементов
Источни
Менеджер ссылок - MakeTimeTable
Сборки
Проекты
Общие проекты
COM
Библиотеки типов
Последние
Обзор

	Имя	Версия
	Microsoft Development Environment Properties...	9.0
	Microsoft Development Environment Properties...	9.0
	Microsoft Development Environment Publish	1.0
	Microsoft Development Environment VC# Projec...	1.0
	Microsoft Development Environment VC++ Cod...	14.0
	Microsoft Development Environment VC++ Proj...	14.0
	Microsoft Development Environment VC++ Proj...	8.0
	Microsoft Development Environment VC++ Proj...	8.0
	Microsoft DirectX Transforms Core Type Library	1.1
	Microsoft DirectX Transforms Image Transforms...	1.1
	Microsoft Disk Quota 1.0	1.0
☑	Microsoft Excel 16.0 Object Library	1.9
	Microsoft External Item Picker	1.0
	Microsoft Fax Service Extended COM Type Library	1.0

Figure 2.15: C# and MS Excel integration

The next step was to connect the namespace:

using Excel = Microsoft.Office.Interop.Excel;



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using Excel = Microsoft.Office.Interop.Excel;
using System.Text.RegularExpressions;


using Microsoft.Office.Interop.Excel;
```

Figure 2.15: C# and MS Excel integration

The objects that Excel operates on have a hierarchical structure and are required for information exchange between the application and the server. The "Application" object is an Excel application that can contain one or more (Excel) books. Links to books contains the "Workbooks" property.

The "Workbook" object contains books that contain one or more pages. Links to these pages or (and) the diagrams contain the properties of "Worksheets" and "Charts".

A “Worksheet” object can contain cell objects or groups of cells. Links to them are implemented us**ing the “Range”object.**

In order to create an Excel file, a new instance of the application is loaded into memory.

Excel.Application excel = new Excel.Application ();

excelApp.Visible = true;

A workbook is added:

```
Excel.Workbook workbook = excel.Workbooks.Add (TypeMissing)
```

To get the necessary sheet of the document, the sheet number is added (the report starts with 1):

```
Excel.Worksheet sheet = excel.Worksheets.get_Item (1);
```

A specific range of cells can be selected. The font, size, color and background can be defined for it :

```
Excel.Range range = sheet.get_Range (sheet.Cells [1, 1], sheet.Cells [10, 10]);
range.Cells.Font.Name = "Times New Roman";
range.Cells.Font.Size = 12;
```

To set the color, the System.Drawing namespace is connected

```
range.Cells.Font.Color = ColorTranslator.ToOle (Color.Red);
range.Interior.Color = ColorTranslator.ToOle (Color.FromArgb (0xFF, 0xFF, 0xCC));
```

To save the active workbook or the certain workbook in a new folder with a new name (with the necessary file format, access mode, and so on), Save and SaveAs methods are used, respectively:

```
excel.Application.ActiveWorkbook.Save ();
excel.Application.ActiveWorkbook.SaveAs (@ "C: \ Document.xlsx", Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Excel.XlSaveAsAccessMode.xlNoChange, Type.Missing, Type. Missing, Type.Missing, Type.Missing, Type.Missing);
```

To open an existing document "Open" method is used, the main parameter specifies the path to the file, the remaining parameters remain empty.

```
excel.Workbooks.Open (@ "C: \ Users \ User \ Documents \ Document.xlsx", Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing, Type.Missing);
```

The search and reading of information from the document is realized in the work. At the stage of determining the size of the working part of the page, the NamedRange.Find method is used. It allows to return a Microsoft.Office.Interop.Excel.Range object that represents the first cell.

The NamedRange.FindNext (Object) method continues the search that was started from the Find method. It has the "After" parameter - the cell after which you want to perform a search. If this argument is not specified, the search starts after the cell in the upper-left corner.

2.6 Collecting data organisation part

Data store is organized using the MS Excel package. Scheduling is formed on the institute which includes about 100 groups, 90 lecturers, 12 classrooms.

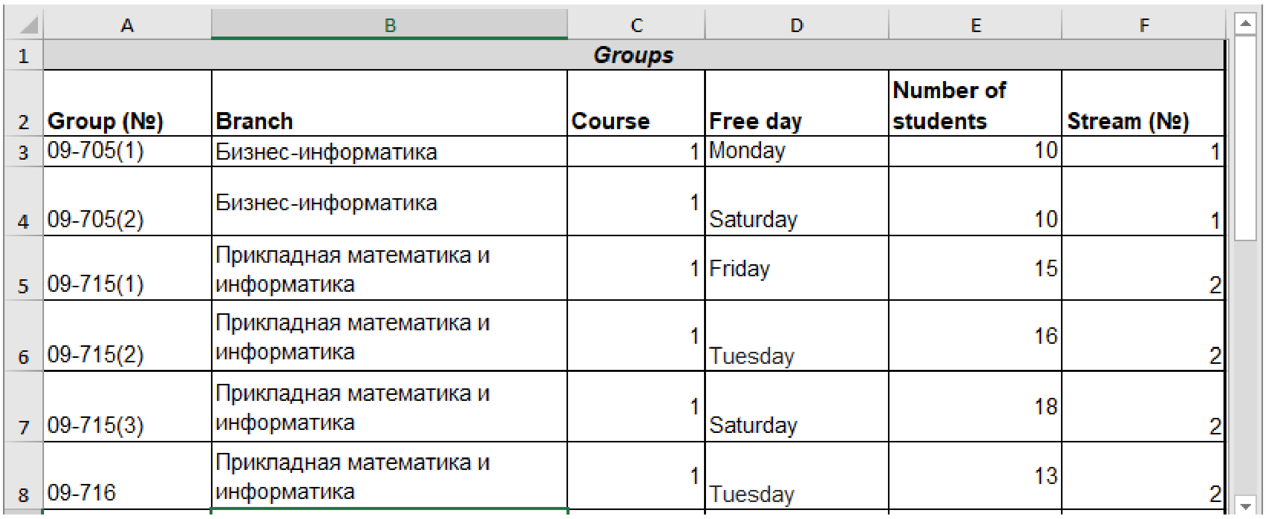
Lists contain all the necessary information for scheduling the timetable and are filled in by the dispatcher who is responsible for scheduling.

Data store is organized using the MS Excel package. Scheduling is formed on the institute which includes about 100 groups, 90 lecturers, 12 classrooms.

Lists contain all the necessary information for scheduling the timetable and are filled in by the dispatcher who is responsible for scheduling.

List of Groups.

List of Groups includes fields of the group numbers, branch of study, course, day off (in addition to the weekend the group has an additional day off, " library day"), the number of students in groups and academic flow number (Fig.2.16).

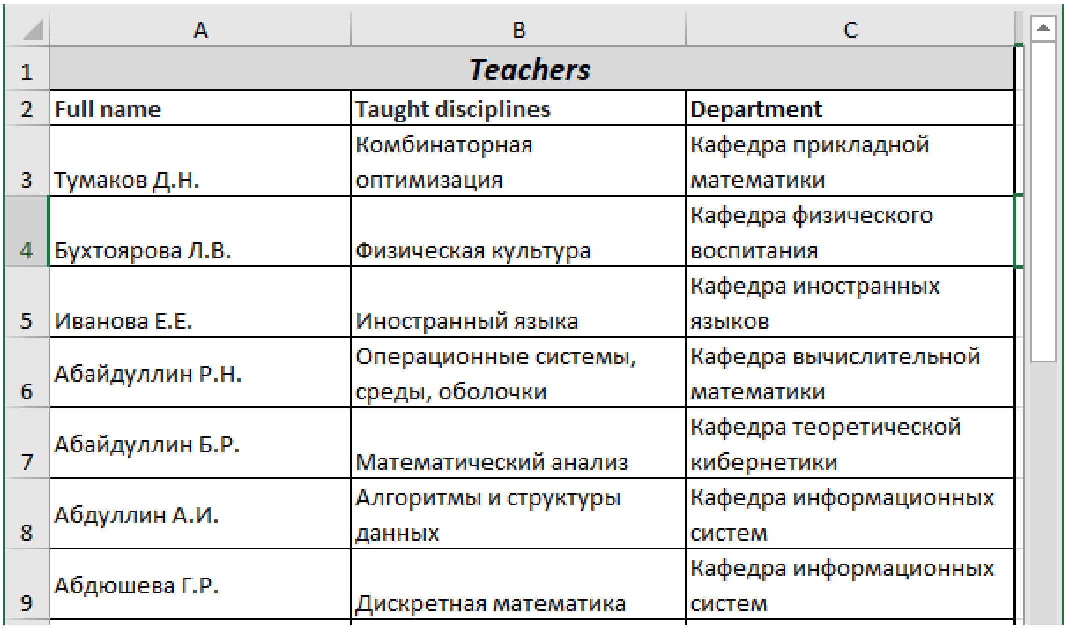


Groups

Group (№)	Branch	Course	Free day	Number of students	Stream (№)
09-705(1)	Бизнес-информатика	1	Monday	10	1
09-705(2)	Бизнес-информатика	1	Saturday	10	1
09-715(1)	Прикладная математика и информатика	1	Friday	15	2
09-715(2)	Прикладная математика и информатика	1	Tuesday	16	2
09-715(3)	Прикладная математика и информатика	1	Saturday	18	2
09-716	Прикладная математика и информатика	1	Tuesday	13	2

Figure 2.16: List of Groups

The list of teachers keeps the name of the teachers, the list of disciplines they conduct, the department to which they are attached (Fig. 2.17).

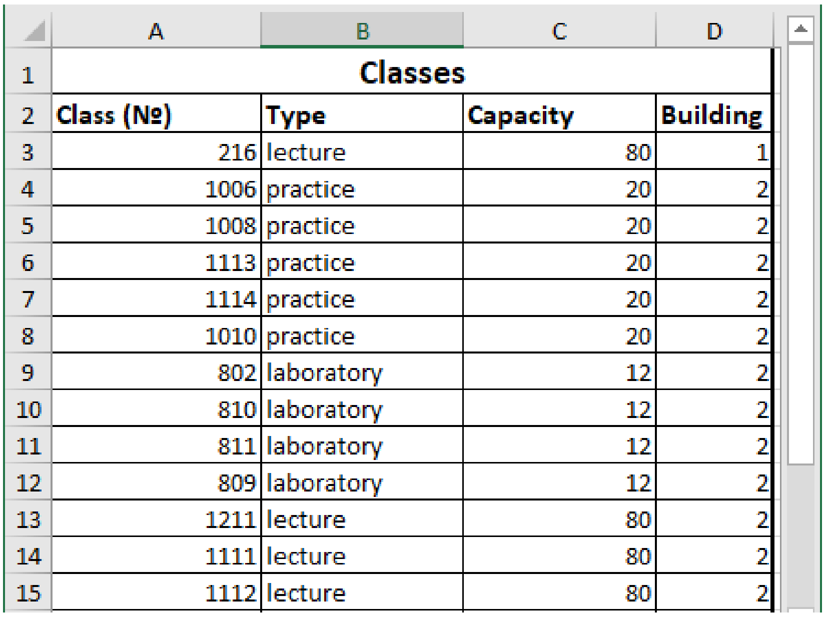


Teachers

Full name	Taught disciplines	Department
Тумаков Д.Н.	Комбинаторная оптимизация	Кафедра прикладной математики
Бухтоярова Л.В.	Физическая культура	Кафедра физического воспитания
Иванова Е.Е.	Иностранный языка	Кафедра иностранных языков
Абайдуллин Р.Н.	Операционные системы, среды, оболочки	Кафедра вычислительной математики
Абайдуллин Б.Р.	Математический анализ	Кафедра теоретической кибернетики
Абдуллин А.И.	Алгоритмы и структуры данных	Кафедра информационных систем
Абдюшева Г.Р.	Дискретная математика	Кафедра информационных систем

Figure 2.17: List of Teachers

Classes in the Institute are divided into the following types: lecture, practical and laboratory. The lists contain the number of the class, its capacity, type and building in which it is located (Fig. 2.18).

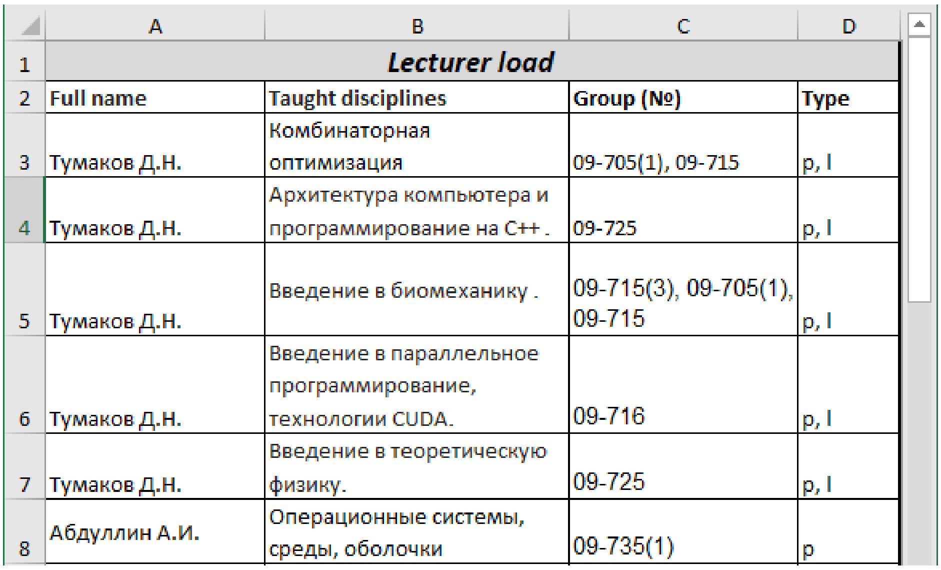


Classes

Class (№)	Type	Capacity	Building
216	lecture	80	1
1006	practice	20	2
1008	practice	20	2
1113	practice	20	2
1114	practice	20	2
1010	practice	20	2
802	laboratory	12	2
810	laboratory	12	2
811	laboratory	12	2
809	laboratory	12	2
1211	lecture	80	2
1111	lecture	80	2
1112	lecture	80	2

Figure 2.18: List of Classes

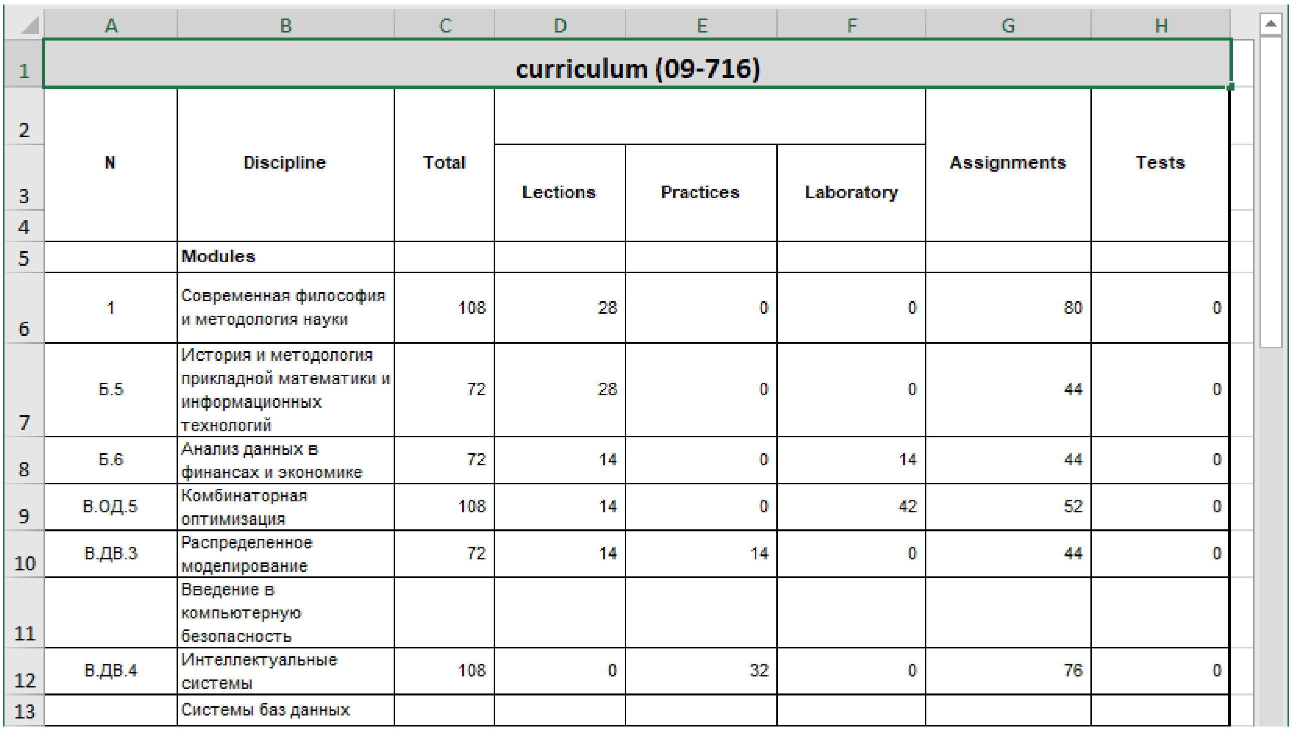
Each department forms a document "Load of teachers" before the beginning of the academic year which contains data about teachers, qualifications, the subjects they teach, number of hours during the semester and links to the groups (Fig. 2.19).



Lecturer load			
Full name	Taught disciplines	Group (№)	Type
Тумаков Д.Н.	Комбинаторная оптимизация	09-705(1), 09-715	p, l
Тумаков Д.Н.	Архитектура компьютера и программирование на C++ .	09-725	p, l
Тумаков Д.Н.	Введение в биомеханику .	09-715(3), 09-705(1), 09-715	p, l
Тумаков Д.Н.	Введение в параллельное программирование, технологии CUDA.	09-716	p, l
Тумаков Д.Н.	Введение в теоретическую физику.	09-725	p, l
Абдуллин А.И.	Операционные системы, среды, оболочки	09-735(1)	p

Figure 2.19: Lecturer load

The most important list the timetable is built on and what should be considered first of all when forming the timetable is "Curriculum of the group". It contains a list of disciplines that must be assigned to a certain group during the semester, the number of hours allocated for practical, lecture, laboratory classes (Fig. 2.20).



curriculum (09-716)							
N	Discipline	Total	Lections	Practices	Laboratory	Assignments	Tests
	Modules						
1	Современная философия и методология науки	108	28	0	0	80	0
Б.5	История и методология прикладной математики и информационных технологий	72	28	0	0	44	0
Б.6	Анализ данных в финансах и экономике	72	14	0	14	44	0
В.ОД.5	Комбинаторная оптимизация	108	14	0	42	52	0
В.ДВ.3	Распределенное моделирование	72	14	14	0	44	0
	Введение в компьютерную безопасность						
В.ДВ.4	Интеллектуальные системы	108	0	32	0	76	0
	Системы баз данных						

Figure 2.20: Curriculum

Conclusion

3.1 Summary

The primary results of this thesis are the following:

1. In this thesis, we have studied the problem of timetable scheduling for educational institutions.
2. Firstly we examinated the subject of the Theory of scheduling. Formulated the problem of scheduling for educational institutions area.
3. Main questions, features and approaches of scheduling automatization were investigated within the work.
4. The software market of automated scheduling were analyzed and more popular programs and systems have been experienced. We studied interfaces and found advantages and disadvantages each of them.
5. We reviewed the known algorithms for solving raised problem.

The main goal of the work was oriented on a specific institution. In this way, the main aspects and manual process of scheduling were investigated and analyzed.

1. On the basis of studied materials and features of the educational process organization the mathematical model were developed.
2. Using the library and the formulated mathematical model with different statements of the objective function, two variants of the timetable were created on different sets of input data. This made it possible to convinced in a correctness of mathematical formulation, since it is paramount and quite difficult concerning this sphere.
3. During the work the generated timetable validation program were developed and tested on the existing timetable.
4. In addition, the form of data store organization necessary for scheduling were determined and described in the thesis.
5. 4. In addition, the form of the data store organization necessary for scheduling were determined and described in thesis.

Bibliography

1. The Gantt chart. Available at: [https://en.wikipedia.org/wiki/Gantt_chart].

2. Henry Ford. Available at: [https://en.wikipedia.org/wiki/Henry_Ford].

3. Maslov M.G. Heuristic algorithm for solving the problem of scheduling in the university. // Mathematical methods in engineering and technology. [Proc. of the XV International Scientific Conference], Tambov, 2002, pp. 86-88 (in Russian).

4. Klevansky N.N. Analysis of the results of the automatic formation of the schedule of the university. [Proc. of the XII International Conference «Information Technologies in Education». Part IV], Moscow, 2002, P. 193 (in Russian).

5. The program «NIKA-Lux» Available at: http://www.nikasoft.ru/Products/ProductInfo.aspx?p=2.

6. The program for scheduling classes "NIKA". User guide. - Moscow: SOFT, 2011.

7. The program «Rector - VUZ». Available at: [http://rector.spb.ru/raspisanie-vuz-4u.php.].

8. Gubenko I.O. Secondary school AVTOR. User Manual, 2005.

9. The program “1C: Automated scheduling. University”. Available at: [https://solutions.1c.ru/catalog/asp_univer/features].

10. Alberto Colorni, Marco Dorigo “A Genetic Algorithm to solve the time table problem” Available at :[http://citeseerx.ist.psu.edu].

11. Erben W., Keppler J. “A genetic algorithm solving a weekly course timetabling problem” [Proc. of the 1st Int. Conf. on Practice and Theory of Automated Timetabling, LNCS 1153], 1995 pp. 198-211.

12. Hahn-Goldberg S. Defining, Modeling, and Solving a Real University Course Timetabling. Available at: [http://tidel.mie.utoronto.ca/pubs/Theses/hahn-goldberg.masc.pdf].

13. Marco C., Krzyztof S., Mauro B., and Olivia Rossi D. “An Effective Approach for the University Course Timetabling Problem“ [Proc. of the Journal of Scheduling, vol. 9, no. 5], 2006, pp. 403-432.

14. Erben W., Keppler J. “A genetic algorithm solving a weekly course timetabling problem” [Proc. of the 1st Int. Conf. on Practice and Theory of Automated Timetabling, LNCS 1153], 1995, pp. 198-211.