Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

DOI: 10.37943/22VHYR8574

Diana Chigambayeva
PhD, Associate Professor, Department of Computational and Data Science
d.chigambayeva@astanait.edu.kz; orcid.org/0009-0009-4577-5081
Astana IT University, Kazakhstan

Makar Goryakin
Bachelor's Degree student, Computer Engineering Department
221345@astanait.edu.kz, orcid.org/0009-0004-3153-7668
Astana IT University, Kazakhstan

Polina Batova
Bachelor's Degree student, Computer Engineering Department
222316@astanait.edu.kz, orcid.org/0009-0000-5423-4548
Astana IT University, Kazakhstan

EDUCATIONAL FLOWS DISTRIBUTION SYSTEM OF UNIVERSITY ACADEMIC GROUPS

Abstract: Planning of educational flows and academic groups by the Academic department is one of the responsible, complex and labor-intensive tasks solved at the stage of the educational process preparation in the university. When planning the work, the Academic department strives to improve the quality of work in order to achieve the best indicators for the types of educational work. One of the main tasks of the IT educational programs is the data analysis and construction of an automated system. Wide opportunities for the implementation of this goal are provided by the topic of scientific research, which is relevant both for the Academic department, which makes up the teaching load of the university, in which educational flows are involved, and for the university as a whole, for the teaching staff of all educational departments. The problem under consideration is really relevant in each higher educational institution and is of scientific interest due to the fact that insufficient attention is paid to mathematical modeling in software development. The aim of the paper is the development and software implementation of the automated system in Python to optimize the business process of distributing educational flows. For the efficiency of writing business processes for optimizing the formation of educational flows and academic groups, the architecture, algorithms and functional model of the software product are described. The functional model of the business process of forming educational flows and academic groups participating in the teaching load calculation at the stage of the educational process preparation at the university is considered. Moreover, the innovative product prototype has been created, which allows us to distribute educational flows evenly and quickly to a large extent, while fulfilling the accepted criteria and limitations of the model. The paper describes the architecture, algorithms and functional model of the software product, which corresponds to the IT field. The development of this innovative program will be useful both for beginners in Python programming and for developers creating their startups.

Keywords: educational flows, educational process; educational work, types of educational work, teaching load, functional model, automated system, program.

Received: 19.05.2025 Accepted: 27.06.2025 Published: 30.06.202

Introduction

Generation of educational flows and academic groups by the Academic department is one of the important and labor-intensive tasks solved at the stages of the educational process preparation in the university. For the efficiency of writing business processes for generating and optimizing educational flows, it is necessary to form an optimal model. It should be noted that when forming the teaching load, the formation of educational flows and groups plays a major role. The initial data in the distribution of educational flows and academic groups according to Fig. 1 are: the working study plan (WSP) for the planned academic year, compiled on the basis of individual study plan (ISP) of students with mandatory components and components for choosing disciplines, the students contingent and classroom fund, as well as the criteria and restrictions.

The work study plan is developed for one academic year, which specifies the volume of credits for each discipline in terms of semesters or trimesters and classroom hours for each discipline.

The students contingent contains the total number of students, master's students, doctoral students in the context of each educational program (EP) and each course of study in accordance with each academic group, as well as the number of students registered for the disciplines of the EP for the planned academic year.

The classroom fund is divided into types of classrooms (flow lecture room, computer room, seminar room, laboratory room) and the number of seats in each classroom available for conducting classes.

The criterion defines the decision-making rule for assessing the optimality of forming educational flows and groups in them. For example, the criterion can be the maximum available distribution of the students number in each educational flows, and others.

Restrictions form the rules according to which educational flows must be formed and groups in flows must be assigned. For example, there are restrictions on the maximization of academic groups number in educational flows depending on the name, component and format of the discipline, the maximization of students number in the classroom, and so on. Restrictions can come from the Academic department, educational departments according to the university development strategy and the department's strategic development plan. Restrictions and criteria can include regulatory documents, internal regulations of the educational institution, and other regulatory documents.

After analyzing the initial data, the educational flows for lectures and academic groups for practical/seminar or laboratory classes are formed, which is the basis for calculating the total volume of teaching load for the university as a whole and for each department of educational programs (Fig. 1).

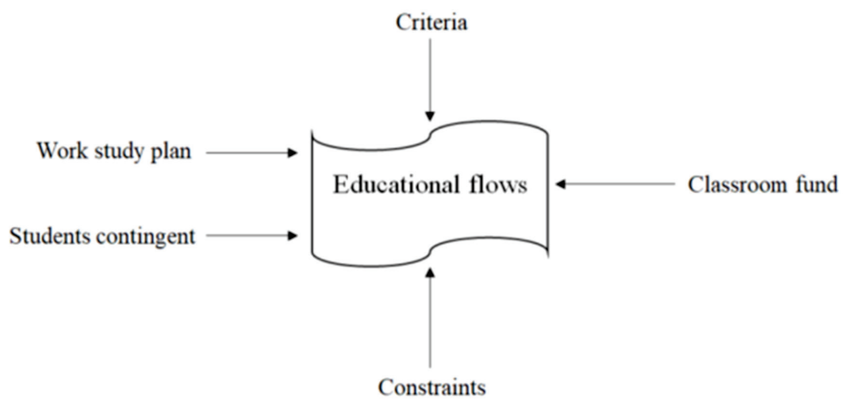


Figure 1. Initial data for the problem of distributing educational flows

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

The basic concepts concerning the formation of educational flows are presented. Educational flows at the university are formed according to the types of educational work:

- lecture session;
- practical lesson;
- laboratory lesson

and are distributed depending on the name of the educational program, disciplines, students contingent and type of classrooms.

For each educational program, the required set of course modules is approved. Then, the ISP is compiled for each student individually after completing the registration procedure for the disciplines. Then, the WSP is formed for one academic year, which reflects the volume of credits for each discipline in terms of semesters and classroom hours for each discipline. As the trimesters progress, the academic workload tends to increase gradually. Thus, in paper [1], the academic workload is shown to be evenly distributed between semesters and courses within the program. The algorithm for designing personality-oriented educational programs based on the analysis of educational process data is presented in the work [2]. In this paper an analysis of the structure and content of competence-oriented curricula is conducted, considering the definition of a compromise of interests of all social groups participating in the educational process. Namely, the compliance of all prerequisites and postrequisites of disciplines, the compliance of related educational programs in the construction of educational streams, and so on.

Each discipline of the WSP in turn has the following components:

- compulsory component (CC) – a list of academic disciplines and corresponding minimum volumes of academic credits, studied by students on a mandatory basis according to the university program;
- university component (UC) – a list of academic disciplines and corresponding minimum volumes of academic credits studied by students depending on the educational program of the university;
- elective component (EC) – a list of academic disciplines and corresponding minimum volumes of academic credits, independently chosen by students in any academic period, taking into account their prerequisites and postrequisites.

In addition, each discipline has the following training formats:

- offline format;
- online format;
- MOOC – Massive Open Online Course – a training course with mass interactive participation using e-learning technologies and open access via the Internet.

The main objective of the research work is the development and software implementation of an automated system in Python for optimizing the business process of distributing educational flows.

To achieve the main goal, it is necessary to complete the following tasks: development of a functional optimization model for the distribution of educational flows and groups, as well as the software product that solves the key problem. The model must take into account all existing restrictions and criteria to ensure uniform and effective distribution of educational flows and groups, which, in turn, will improve the quality of the educational process and job satisfaction of teachers.

The scientific novelty of this paper is expressed in the development of the optimization model for the educational flows and academic groups distribution at the university with given restrictions and criteria using the modern Python programming. Also, the development of the software product is applicable in any university, using acceptable restrictions and criteria, which will certainly speed up the educational process.

The scientific and practical significance of the research lies in the development of the innovative product prototype that allows solving the key task of business processes for distributing educational flows in accordance with the university development strategy. The development of this innovative program will be useful both for beginners in Python programming and for developers creating their startups.

Literature review

The research of the education quality problem, its optimization are described in the works [3], [4]. The functional model of optimization of distribution of the teaching staff workload is especially relevant [5], [6]. This paper examines the system of distributing educational flows and academic groups between educational departments. Note that when forming the teaching load of educational departments, the key factor is the educational flows and groups formation involved in calculating the workload. The working model for optimizing the distribution of academic groups across educational flows is particularly relevant under the following conditions:

- in case of a dynamically changing teaching load;
- in case of a shortage of classroom funds;
- depending on the disciplines components (CC/UC/EC);
- in case of a dynamically changing contingent of students;
- depending on the course format (online, offline, MOOC);
- depending on the specifics of educational programs [6].

Previously conducted studies on the teaching load distribution [7] have shown that all of them to one degree or another describe the mechanisms of teaching load distribution, but at the same time practically do not pay attention to the distribution methodology. Here, when calculating the teaching load in the educational flows distribution, several criteria can be considered by analogy:

a) homogeneity of academic groups, when groups from only one educational program being seated in one educational flow;

b) the average number of groups in educational flows distribution, with uneven distribution of groups across different educational flows;

c) maximizing the educational flows of each discipline of each type of study work, taking into account the quality of the learning process.

As can be seen, these decision-making criteria are quite ineffective, which in turn causes the urgency of the problem. Currently, one of the possible ways to improve the success of the educational flows distribution at the university is an algorithm based on deep learning [8], [9].

The implementation of the conceptual provisions of the management theory in the educational process is impossible without using the modern tools based on automation and using of management principles at the processes of the teaching load optimization, as well as research of operations and optimization of numerical methods [10], [11]. In this regard, the synthesis of the studied methods was carried out and the generalized method for generating educational flows was proposed, which is based on models for supporting decision-making.

Thus, the obtained model combine the advantages of existing models, and their application will allow finding "optimal" solutions to the problem of initializing educational flows in an acceptable time. In addition, the created model will be the basis for developing special software products for automating business processes for generating educational flows at the university. Thus, the problem being solved is considered as a problem of best approximation, which is reduced to the minimax problem. The results of the study can serve as a basis for further research in this area and be useful for the development of a modern education system.

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

Functional model

A functional model of a business process is a representation of a process or system that focuses on the functions or operations performed by the system under various conditions [12]. It describes what the system does or should do without specifying how exactly these functions are performed [12]. Functional models are used to analyze requirements, design systems, and document system functions [13]. The mathematical model for optimizing resource allocation was developed to reduce the time required to make management decisions on resource allocation problems in the work [14], [15].

The construction of functional and mathematical models for the distribution of the educational work volume at the university can be found in [10] (block A3, Fig. 2), as well as the teaching load distribution among university teachers can be found in [11] (block A4, Fig. 2).

The functional model described for the process of distributing educational flows contains input and output data, the main functions, their interrelations in business processes, as well as the criteria and constraints that control these processes. The model indicates the key tasks of the entire business process, who is responsible for them ("holders") and how they interact to achieve the final goal.

Fig. 2 shows the initial level of the functional model A0 "Form the university's teaching load for the departments" with flows of incoming and outgoing information, procedures for completing the specified tasks, and control and management elements on the basis of which it is carried out.

The model consists of four blocks: block A1 "Prepare/modify source data", block A2 "Form the number of group flows by discipline", block A3 "Calculate the teaching load of the university", block A4 "Approve the teaching load for each department". Each of the blocks A1-A4 represents separate sub-stages within the overall process of the teaching load distributing at the university (Fig. 2).

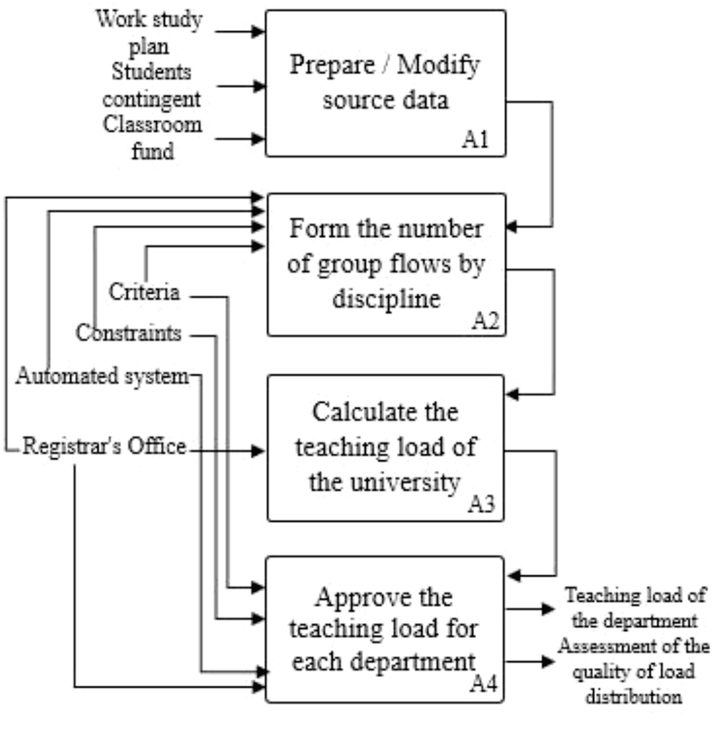


Figure 2. Functional model A0

Next in Fig. 3 the context diagram A2 "Form the number of group flows" is shown.

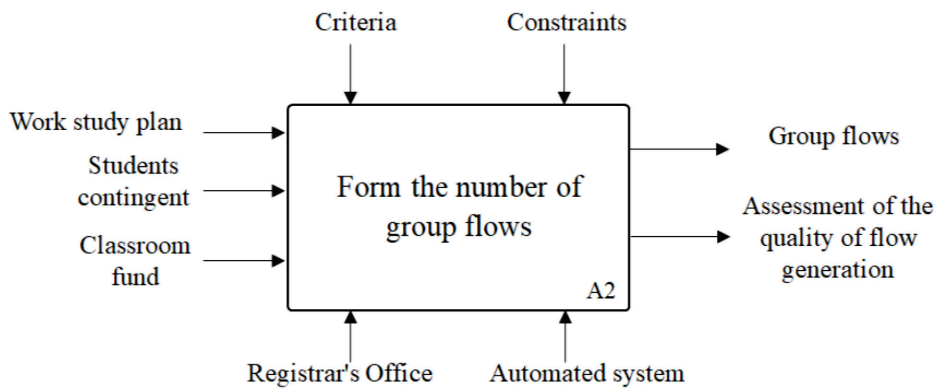


Figure 3. Context diagram A2

The functional diagram shown in Fig. 4 is the decomposition of the context diagram A2 and consists of the following blocks: block A21 "Enter/modify initial data", block A22 "Assign a number of group flows to disciplines" and block A23 "Control the quality of formation".

The process of entering and modifying the initial data (A21) is the initial stage in the system of educational flows and academic groups forming. It precedes the educational flows formation (A22) and is directly related to the quality control of the educational flows formation (A23). At this stage, the Academic department or the Registrar's office enters the initial data or adjusts the planned data that will be used to form the educational flows and groups.

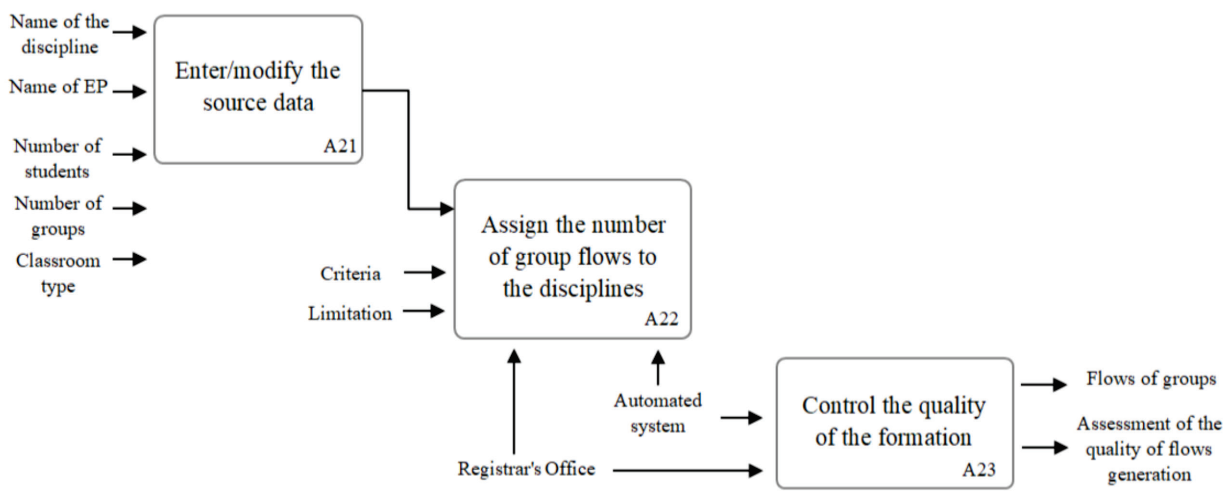


Figure 4. Decomposition of the Context diagram A2

The initial data may include the name, component and format of the discipline, the name of the educational program, the number of students, the number of academic groups, and the type of classrooms. The process requires attention to detail and accuracy, since the initial information serves as the basis for all subsequent operations of teaching load distributing. After entering the data, the procedure for their verification and approval follows. This determines that all critical parameters are taken into account and the data is ready for the automated system using in the next stage - the educational flows and academic groups formation.

The process of forming the number of educational flows and academic groups (A22) follows from the stage of input and modification the initial data (A21). This process is engaged in the structuring educational flows and academic groups of students based on the provided

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

data. The purpose of the process is to form the optimal number of educational flows for the efficient use of resources and classroom fund. The process takes into account certain criteria and restrictions, which include not only classroom capacity, but also the requirements of educational programs. It is important to achieve a balance between available resources and the educational needs of students so that each discipline is adequately represented in the course schedule in the future. This requires consistency of the data entered in the previous stage and a clear understanding of the current academic and strategic goals of the university.

Assigning a certain number of educational flows and academic groups to disciplines is critical to ensuring the quality of the educational process, as it directly affects the teaching load, the course schedule and the occupancy of classrooms. This forms the basis for a flexible and effective schedule that can be easily adapted to changes in the educational process.

Quality control of the educational flows formation (A23) is the final stage in the cycle of managing educational flows. This process is necessary to ensure the effectiveness of the educational flows formation and compliance with all established criteria and restrictions. At this stage, the results of educational flows formation are assessed, which allows identifying any errors or inconsistencies that have arisen in the process. The quality assessment of educational flows formation is a comprehensive check that can lead to return the previous stages to correct data or change the restrictions of the educational flow. This ensures continuous improvement of business processes and maintains a high standard of the educational process at the university.

Prototype of innovative product

The Python programming language was used to develop the program. Python was chosen due to its conciseness and the presence of convenient built-in libraries for working with .xlsx documents.

The program consists of 5 .py extension files, linked together and used for different parts of the application, and an image file for a shortcut with the .ico format. These files form the monolithic application, wrapped in the .exe format, which allows you to easily run it on any personal computer without additional installation steps.

Input data:

- Excel table "Formation of educational flows". From this table, for each discipline, the number of academic groups of which educational programs is taken, the total number of students, the number of educational flows for various types of educational work (lectures, practical classes, etc.) are then calculated.
- Excel table "Students contingent", from which the number of students in each academic group for each educational program is also then calculated using the program.
- Excel tables "Selection of elective disciplines for the 2nd year", "Selection of elective disciplines for the 3rd year", containing the information on the elective disciplines choicen by students. Elective disciplines are available only for students in the 2nd and 3rd years of the bachelor's degree programs.

The program initializes and seats educational flows for lectures and practical classes of undergraduate students of all courses in all disciplines, including disciplines of the elective component (EC). One educational flow can consist of one or several academic groups of one or different educational programs depending on the discipline (general block, mathematical block), discipline component (CC/UC/EC), course format (offline, online, MOOC), types of academic work and other criteria. Information about disciplines, academic groups and the contingent of students is taken from Excel tables (three input files). The results of the program are saved in a separate Excel file. For ease of use, a simple graphical interface has been developed that accepts input data.

DOI: 10.37943/22VHYR8574

Description of the program

The application window is launched. The user needs to load the required initial data via the graphical interface and click on the button "Create educational flows and groups" (Fig. 5).

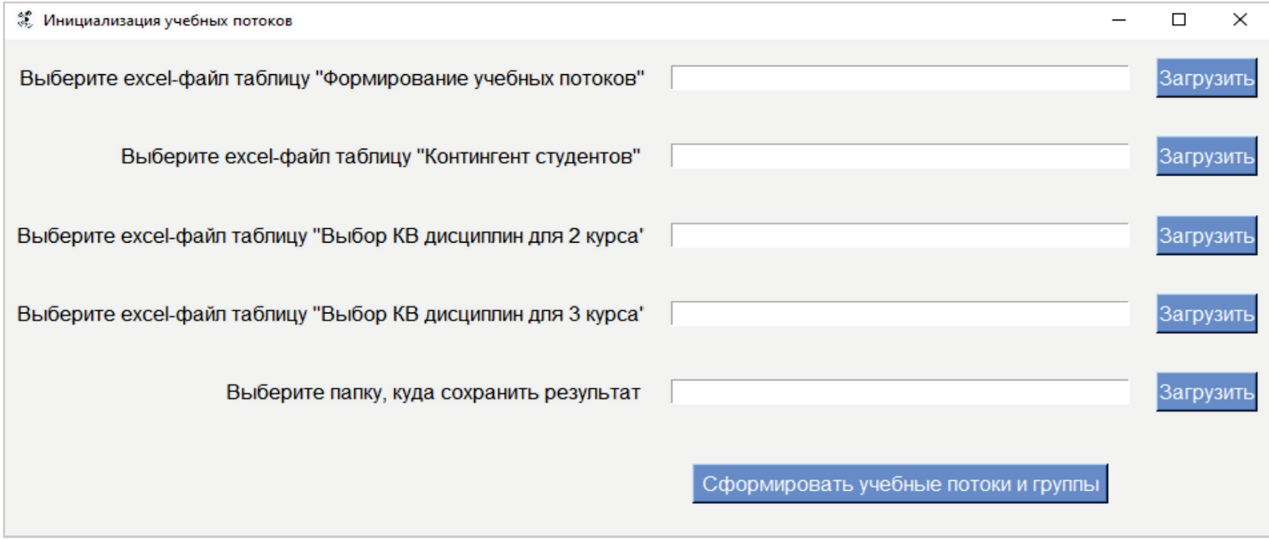


Figure 5. Appearance of the program

Then the program reads data from Excel files, calculates the number of students in each academic group and the number of students who have chosen certain elective courses, and processes each sheet for the 1st, 2nd and 3rd years in turn from the table of educational flows formation. Thus, for each row, the number of academic groups for each discipline and each educational program is determined. Based on this, using the *genenerate_groups* and *genenerate_groups_kv* functions, the names of the academic groups that take this discipline are created. Then it is checked whether the discipline belongs to the EC component and, depending on this, educational flows with groups are formed using the *generate_streams* functions for CC/UC disciplines or the *generate_lecture_streams_kv* and *generate_seminar_streams_kv* functions for EC disciplines, respectively.

Let's describe the *generate_streams* function.

The function accepts the following parameters:

- *lecture_streams_count* – the number of educational flows into which academic groups need to be distributed;
- *all_groups* – a list of all student groups by educational programs;
- *number_of_students* – a dictionary, where the keys are groups and the values are the number of students in each group.

The software process operates according to the following algorithm:

1. Sorting the EPs by the number of groups in descending order to distribute the most numerous EPs first;
2. Calculating the average number of groups per educational flow (*groups_per_stream*);
3. Dividing the groups of one EP into educational flow.

From the sorted list, equal flows from *groups_per_stream groups* are created from each EP in turn. The remaining groups of each EP are saved in a separate list. Thus, the "ideal" flows are obtained, consisting only of groups of one EP. For example, SE-2201, SE-2202, SE-2203, SE-2204, SE-2205 (Fig. 6):

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

```
for groups in all_groups:
    streams = split_list(groups, groups_per_stream)
    for stream in streams:
        if len(stream) == groups_per_stream:
            result_streams.append(stream)
        else:
            remaining_groups.append(stream)
```

Figure 6. Fragment of the program code

4. Calculating how many undistributed groups and unfilled educational flows remain;

5. Distributing the remaining EP groups among educational flows taking into account maximizing the number of students in one educational flow.

For unallocated groups, each EP checks whether they can be made into a separate educational flow, depending on the number of students in these groups (Fig. 7):

```
max = -1
max_stream = []
for remaining_group in remaining_groups:
    if remaining_streams != 0 and remaining_groups_count % remaining_streams == len(remaining_group):
        num = calculate_number_of_students(remaining_group, number_of_students)
        if num > max:
            max = num
            max_stream = remaining_group
if len(max_stream) > 0:
    result_streams.append(max_stream)
    remaining_groups.remove(max_stream)
```

Figure 7. Fragment of the program code

6. Dividing the remaining groups into educational flows: remaining groups are sorted by number of students in descending order and combined into one common flat list. This list is again split into subgroups of size *groups_per_stream*, which are added to *result_streams*.

7. Reducing and merging "extra" educational flows: if the number of resulting lecture flows exceeds the required number of lecture flows *lecture_streams_count*, the least populated lecture flow is removed and its groups are distributed across other lecture flows to balance the number of students (Fig. 8):

```
while len(result_streams) > lecture_streams_count:
    sorted_streams = sorted(result_streams, key=lambda stream: calculate_number_of_students(stream, number_of_students),
reverse=True)
    result_streams.remove(sorted_streams[-1])
    sorted_streams[-1] = sorted(sorted_streams[-1], key=len, reverse=True)
    for i in range(len(sorted_streams[-1])):
        group = sorted_streams[-1].pop(-1)
        least_stream_index = result_streams.index(sorted_streams[-2 - i])
        result_streams[least_stream_index].append(group)
```

Figure 8. Fragment of the program code

8. Checking integrity: checks if the total number of groups in the original data and the resulting training flows match. If not, an exception is thrown.

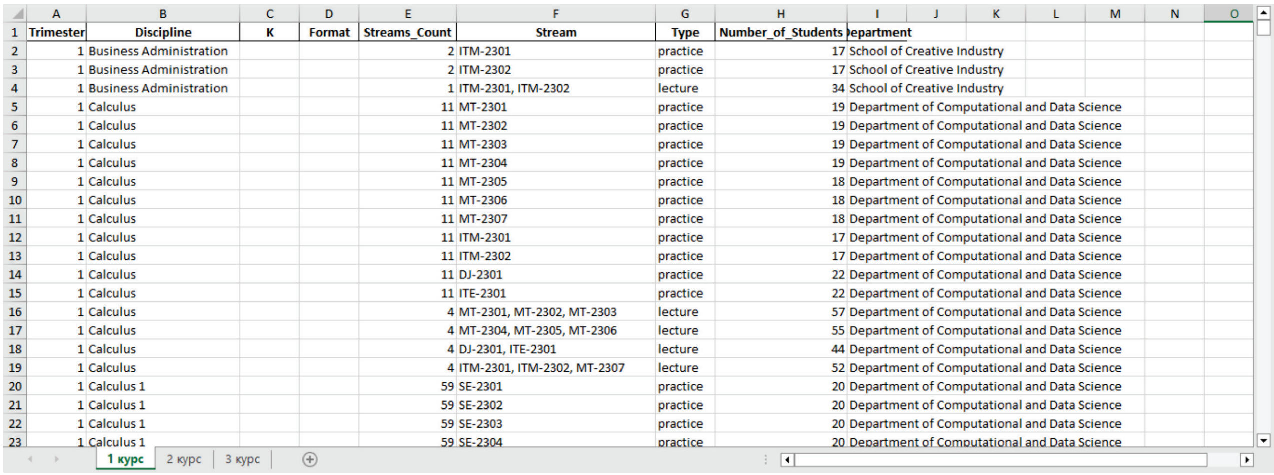
Thus, the *generate_streams* function generates and returns educational flows of lectures for one CC/UC discipline. In a similar way, this function creates educational flows for practical classes. Since the number of educational flows for seminar classes of CC/UC disciplines coincides with the number of groups taking this discipline, then in each educational flow of seminar classes there is exactly one group.

Next, the code for the functions *generate_lecture_streams_kv* and *generate_seminar_streams_kv* is written.

DOI: 10.37943/22VHYR8574

After initializing the educational flows, the results are formed as a nested dictionary *nested_dict*, which is then converted to a *DataFrame df*. In this DataFrame, the rows containing the lists are expanded, and information about the number of students in each educational flow is added.

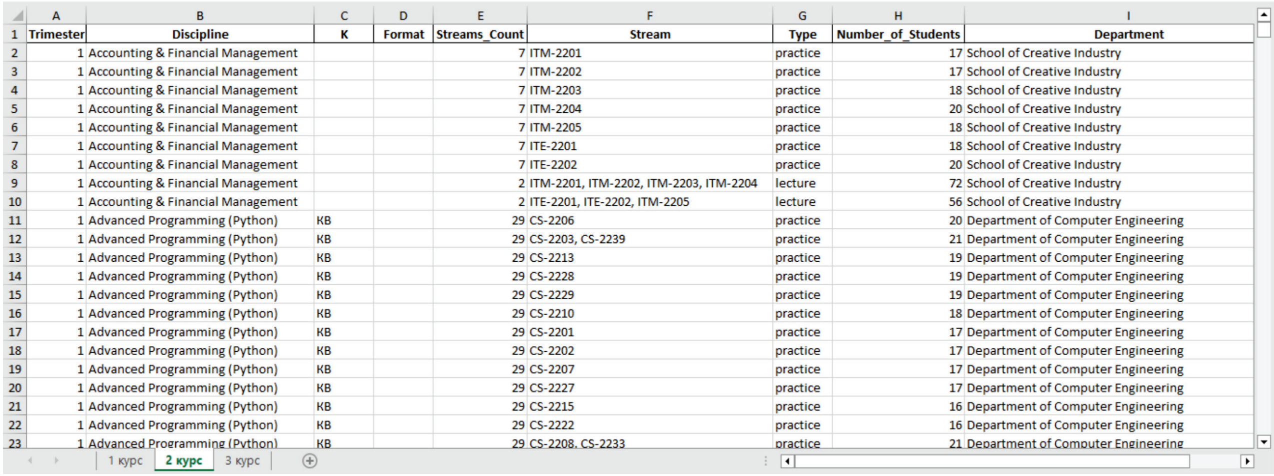
The result is written to a new file "Initialization of educational flows for undergraduate studies 2024-2025", and a separate sheet is created for each course in one table (Fig. 7-9).



Trimester	Discipline	K	Format	Streams_Count	Stream	Type	Number_of_Students	epartment
1	Business Administration			2	ITM-2301	practice	17	School of Creative Industry
1	Business Administration			2	ITM-2302	practice	17	School of Creative Industry
1	Business Administration			1	ITM-2301, ITM-2302	lecture	34	School of Creative Industry
1	Calculus			11	MT-2301	practice	19	Department of Computational and Data Science
1	Calculus			11	MT-2302	practice	19	Department of Computational and Data Science
1	Calculus			11	MT-2303	practice	19	Department of Computational and Data Science
1	Calculus			11	MT-2304	practice	19	Department of Computational and Data Science
1	Calculus			11	MT-2305	practice	18	Department of Computational and Data Science
1	Calculus			11	MT-2306	practice	18	Department of Computational and Data Science
1	Calculus			11	MT-2307	practice	18	Department of Computational and Data Science
1	Calculus			11	ITM-2301	practice	17	Department of Computational and Data Science
1	Calculus			11	ITM-2302	practice	17	Department of Computational and Data Science
1	Calculus			11	DJ-2301	practice	22	Department of Computational and Data Science
1	Calculus			11	ITE-2301	practice	22	Department of Computational and Data Science
1	Calculus			4	MT-2301, MT-2302, MT-2303	lecture	57	Department of Computational and Data Science
1	Calculus			4	MT-2304, MT-2305, MT-2306	lecture	55	Department of Computational and Data Science
1	Calculus			4	DJ-2301, ITE-2301	lecture	44	Department of Computational and Data Science
1	Calculus			4	ITM-2301, ITM-2302, MT-2307	lecture	52	Department of Computational and Data Science
1	Calculus 1			59	SE-2301	practice	20	Department of Computational and Data Science
1	Calculus 1			59	SE-2302	practice	20	Department of Computational and Data Science
1	Calculus 1			59	SE-2303	practice	20	Department of Computational and Data Science
1	Calculus 1			59	SE-2304	practice	20	Department of Computational and Data Science

1 курс | 2 курс | 3 курс

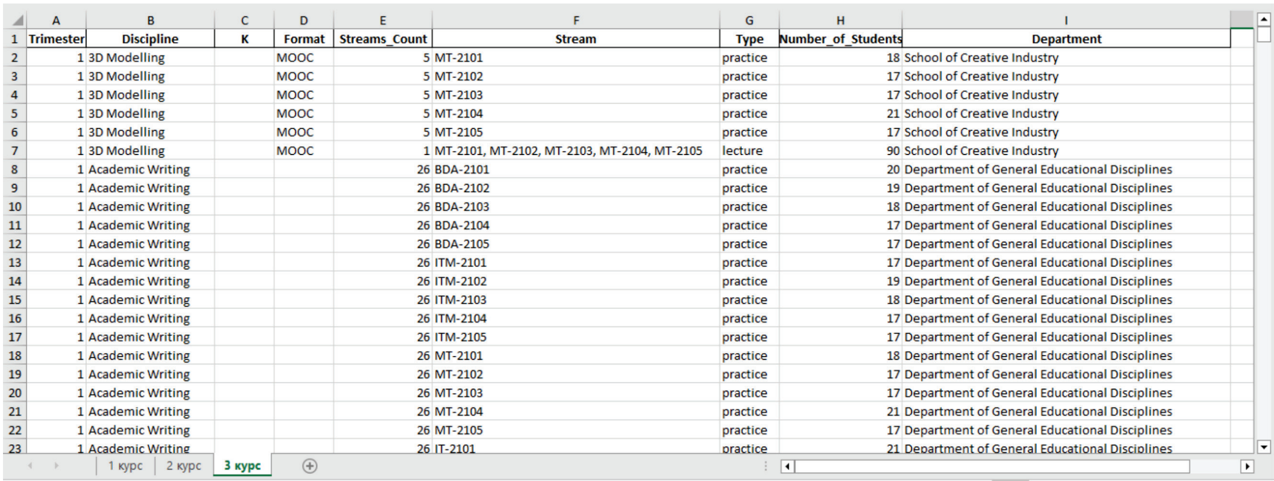
Figure 7. The result of the program for the 1st course



Trimester	Discipline	K	Format	Streams_Count	Stream	Type	Number_of_Students	Department
1	Accounting & Financial Management			7	ITM-2201	practice	17	School of Creative Industry
1	Accounting & Financial Management			7	ITM-2202	practice	17	School of Creative Industry
1	Accounting & Financial Management			7	ITM-2203	practice	18	School of Creative Industry
1	Accounting & Financial Management			7	ITM-2204	practice	20	School of Creative Industry
1	Accounting & Financial Management			7	ITM-2205	practice	18	School of Creative Industry
1	Accounting & Financial Management			7	ITE-2201	practice	18	School of Creative Industry
1	Accounting & Financial Management			7	ITE-2202	practice	20	School of Creative Industry
1	Accounting & Financial Management			2	ITM-2201, ITM-2202, ITM-2203, ITM-2204	lecture	72	School of Creative Industry
1	Accounting & Financial Management			2	ITE-2201, ITE-2202, ITM-2205	lecture	56	School of Creative Industry
1	Advanced Programming (Python)	KB		29	CS-2206	practice	20	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2203, CS-2239	practice	21	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2213	practice	19	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2228	practice	19	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2229	practice	19	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2210	practice	18	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2201	practice	17	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2202	practice	17	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2207	practice	17	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2227	practice	17	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2215	practice	16	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2222	practice	16	Department of Computer Engineering
1	Advanced Programming (Python)	KB		29	CS-2208, CS-2233	practice	21	Department of Computer Engineering

1 курс | 2 курс | 3 курс

Figure 8. The result of the program for the 2nd course



Trimester	Discipline	K	Format	Streams_Count	Stream	Type	Number_of_Students	Department
1	3D Modelling		MOOC	5	MT-2101	practice	18	School of Creative Industry
1	3D Modelling		MOOC	5	MT-2102	practice	17	School of Creative Industry
1	3D Modelling		MOOC	5	MT-2103	practice	17	School of Creative Industry
1	3D Modelling		MOOC	5	MT-2104	practice	21	School of Creative Industry
1	3D Modelling		MOOC	5	MT-2105	practice	17	School of Creative Industry
1	3D Modelling		MOOC	1	MT-2101, MT-2102, MT-2103, MT-2104, MT-2105	lecture	90	School of Creative Industry
1	Academic Writing			26	BDA-2101	practice	20	Department of General Educational Disciplines
1	Academic Writing			26	BDA-2102	practice	19	Department of General Educational Disciplines
1	Academic Writing			26	BDA-2103	practice	18	Department of General Educational Disciplines
1	Academic Writing			26	BDA-2104	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	BDA-2105	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	ITM-2101	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	ITM-2102	practice	19	Department of General Educational Disciplines
1	Academic Writing			26	ITM-2103	practice	18	Department of General Educational Disciplines
1	Academic Writing			26	ITM-2104	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	ITM-2105	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	MT-2101	practice	18	Department of General Educational Disciplines
1	Academic Writing			26	MT-2102	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	MT-2103	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	MT-2104	practice	21	Department of General Educational Disciplines
1	Academic Writing			26	MT-2105	practice	17	Department of General Educational Disciplines
1	Academic Writing			26	IT-2101	practice	21	Department of General Educational Disciplines

1 курс | 2 курс | 3 курс

Figure 9. The result of the program for the 3rd course

Scientific Journal of Astana IT University
ISSN (P): 2707-9031 ISSN (E): 2707-904X
VOLUME 22, JUNE 2025

The program "Initialization of educational flows and groups" when distributing the teaching load of the university is a full-fledged application for personal computers. Users can download the program to their computer and interact with it through a graphical interface. Thus, this program helps to optimize the process of distributing the teaching load between the university teachers, ensuring effective and fast uniform initialization of academic groups. Using the program allows to reduce the workload of the Academic department employees several times, automating this task.

Methods and Materials

The objects of the rersarch are business processes for the distribution of educational flows academic groups of Astana IT University.

To write the functional model, data analysis methods and operations research methods were used [16]. Empirical and quantitative methods were used to write the program [17]. As empirical methods, an assessment was made of the analysis of the questionnaires of teachers and the university administration to optimize the calculation of educational flows and groups, as well as their uniform distribution. As quantitative methods, methods of mathematical analysis and discrete mathematics using finite sets series with Boolean variables, as well as function approximation methods were used [18]. For writing this innovative program for educational flows distribution we use the mathematical model and calculations of the university educational work volume distribution which were described in detail in the work [18]. Note that the presented models were tested on the basis of university data in the Excel environment according to the intra-university scientific project of Astana IT University for the 2024-2025 academic year [19], [20].

The Python programming language was used to develop the program. Python was chosen due to its conciseness and the presence of convenient built-in libraries for working with .xlsx documents.

The following libraries were used to write the program: Pandas (used for data processing), Tkinter (provides functionality for writing the interface), PyInstaller (collects the necessary modules and libraries into one executable .exe file).

The development of this program and methods of applications will be useful both for beginners in the Python programming and for professional developers creating their startups.

Results

The program for initializing educational flows and groups when distributing the educational workload provides the following functional capabilities:

1. Receiving input data from the user: using a convenient graphical interface, the user can easily upload the data necessary for the program to work.

2. Processing input data: the program counts the number of students in each academic group, links disciplines to the list of groups for each course and generates the names of academic groups by educational flows based on the data uploaded by the user.

3. Distribution of groups by educational flows for each discipline depending on the discipline component: special algorithms have been developed for initializing educational flows and groups by disciplines and different types of academic work (lectures and practical/laboratory classes) depending on the discipline component.

4. Outputting the result in an easy-to-read form: the program writes the distribution result to a new Excel file, concisely placing information about each educational flow on the separate line. The resulting file is open to changes and adjustments by the Academic department and the Registrar's office.

Discussion

It should be noted that it is importance of modeling the key task of business processes for optimizing educational flows formation, by means of which the automated system is built. The following achievements were selected as criteria for the optimal formation of educational flows and groups: maximizing of the classroom fund distribution; the maximum of available distribution of the students number in each flow; uniform distribution of students across all flows of one discipline; minimization of educational flows of each discipline for each type of educational work, taking into account the distribution of all remaining vacancies and due to the limited classroom fund. All input data, criteria and restrictions are described by context diagrams of the functional model. This approach ensures the fair distribution of educational flows, as well as the effective use of the potential for educational departments for its further development.

Note that thanks to the developed program, the response time to the request was no more than 5 seconds and more than 4700 lines of educational flows were generated. The application has successfully passed testing and implementation at the university has been launched. Moreover, it is expected that thanks to the implementation of this innovative product, the university will be able to increase the optimization of educational flows thanks to the developed algorithm. It is important to note that the results of the research were achieved during the intra-university scientific project of Astana IT University for the 2024-2025 academic year, a copyright certificate for the software product was received. The results of the paper can serve as a basis for further research in this area and be useful for the development of a modern education system.

Conclusion

In this research work, the new methodology for solving complex optimization problems that arise when modeling and optimizing the educational work volume distribution at the university has been developed. It should be noted that comparative experiments under labor-intensive and time-limited conditions confirm the effectiveness of this methodology in solving the problems of distributing educational flows and academic groups of educational programs, which in turn contributes to the implementation of high-quality software. Thus, the proposed algorithm for planning educational flows of the university will allow not only to take into account the tasks that must be performed in a strictly defined period (for example, planning for the next academic year), but also to carry out an even distribution of educational flows during the academic year for dependent and independent reasons. And the functional model of business processes built on its basis will be able to carry out an optimized distribution of educational flows in an automated mode, which will significantly reduce time, eliminate errors, and, if necessary, quickly make amendments in the future to the plan of the university teaching load and the academic schedule.

Based on the optimization model, the software product was written for initializing educational flows and academic groups for Astana IT University, facilitating the work of the Academic department in distributing the teaching load of the university teachers by automating one of the stages of this process. It should be noted that the program is effective and applicable for any universities, observing the criteria, specifics and limitations of the university itself. The main application belongs to the field of higher education.

A graphical interface has been developed for the application on the computer, which is one main window that accepts input data and starts the algorithm. The program supports processing of .xlsx files, the result is also written in .xlsx format, which is the Excel table.

Acknowledgment

The work was carried out with the support of grant funding from the intra-university project of Astana IT University on the topic "Optimization of business processes for distributing the teaching load of teaching staff at Astana IT University" (№ 0124РКИ0477).

References

[1] Gaete, M., & González-Araya, M.C. (2024). Assessment of the academic load in a curriculum through an optimization model: case study of a master program. *International Conference on Operations Research and Enterprise Systems*, 269–274. https://doi.org/10.5220/0012365400003639

[2] Soltan, G., Zunimova, G., & Sarsenbayeva, G. (2020). The Algorithm for Designing Competency Oriented Educational Programs Based on the Data Analysis of Academic Processes. Proceedings - 2020 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT, 488–491. https://doi.org/10.1109/USBEREIT48449.2020.9117787

[3] Ishanov, Ye., & Magauova, A. (2021). Development of information and management competence of the heads of the structural unit of a modern university. *Al-Farabi Kazakh National University*, 68(3), 138–158. https://doi.org/10.26577/JES.2021.v68.i3.12

[4] Saule, K., Indira, U., Aleksander, B., Gulnaz, Z., Zhanl, M., Madina, I., & Györök, G. (2018). Development of the information and analytical system in the control of management of university scientific and educational activities. *Acta PolytechnicaHungarica*, 15(4), 27–44. https://doi.org/10.12700/APH.15.4.2018.4.2

[5] Kargin, D., Merzadinova, G., Safarov, R., & Mahmutov, B. (2021). On the degree of influence of the teaching load on the scientific and innovative activity of scientists using the example of ENU. *Bulletin of L.N. Gumilyov Eurasian National University*, 1(134), 67–73. https://doi org/10.32523/2616-6895-2021-134-1-67-74

[6] Blair, B. L., Slagle, D. R., & Williams, A. M. (2021). Institutional and programmatic determinants for graduate public affairs' online education: Assessing the influence of faculty workload. *Teaching Public Administration*, 40(2), 181-198. https://doi.org/10.1177/01447394211017326 (Original work published 2022)

[7] Kumargazhanova, S., Erulanova, A., Soltan, G., Suleimenova, L., & Zhomartkyzy, G. (2018). System of indicators for monitoring the activities of an educational institution. Proceedings - 2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT, 179–182. https://doi.org/10.1109/USBEREIT.2018.8384580

[8] Nieto, Y., García-Díaz, V., Montenegro, C., & Crespo, R. (2019). Supporting academic decision making at higher educational institutions using machine learning-based algorithms, Soft Computing, 23(12), 4145–4153. https://doi.org/10.1007/s00500-018-3064-6

[9] Vanessa N., Y., Diaz, V.G., & Montenegro, C.E. (2016). Academic decision-making model for higher education institutions using learning analytics. *4th International Symposium on Computational and Business Intelligence, ISCBI,* 27–32. https://doi.org/10.1109/ISCBI.2016.7743255

[10] Rakhmetullina, Z., Soltan, G., Mukasheva, R., Mukhamedova, R., & Tezekpayeva, S. (2021). Functional and architectural solution of a software package for the analysis of educational data. International Congress of Advanced Technology and Engineering, ICOTEN. https://doi.org/10.1109/ICOTEN52080.2021.9493505

[11] Kumargazhanova, S., Soltan, G., Zhomartkyzy, G., & Suleymenova, L. (2019). Analytical monitoring model of educational system.ACM International Conference Proceeding Series, 14, 1-5. https://doi.org/10.1145/3330431.33304

[12] Chigambayeva, D., Goryakin, M., &Batova, P. (2024). Optimization models for teaching load formation of university teachers, *IEEE AITU: Digital Generation*International Conference Proceedings, 146–149. https://doi.org/10.1109/IEEECONF61558.2024.10585368

[13] Myriam, R.G., & Marcela G.-A. (2024). Assessment of the Academic Load in a Curriculum Through an Optimization Model: Case Study of a Master Program. *Conference: 13th International Conference on Operations Research and Enterprise Systems*, https://doi.org/10.5220/0012365400003639

[14] Nageswaran, M., Ruzana, I., & Noorlizawati A.R. (2021). Programming Subject Allocation Model for Private School Teachers. *Open International Journal of Informatics*, 9(2), 50-57. https://www.researchgate.net/publication/357168840
[15] Syadatul, S., Najihan, A., & Nurul, H. (2022). The Department Team Teaching Assignment Problem Using Zero-One Integer Programming. *Journal of Computing Research and Innovation*, 5(1). https://doi.org/10.24191/jcrinn.v5i1.156
[16] Mushabe, C., Abreh, M., & Mills, C. (2022). Reconceptualizing faculty workload as a measure of human resource motivation and performance in higher education. *Social Sciences & Humanities Open*, 6(1), https://doi.org/10.1016/j.ssaho.2022.100279
[17] Stacey, M., McGrath-Champ, S. & Wilson, R. (2023). Teacher attributions of workload increase in public sector schools: Reflections on change and policy development. *J Educ Change,* 24, 971–993. https://doi.org/10.1007/s10833-022-09476-0
[18] Chigambayeva, D., & Soltan, G. (2025). Constraction of distribution models of the university educational work volume, *AITU Scientific Journal*, 5–18. https://doi.org/ 10.37943/21HIVR6985
[19] Lai, A., Mohamed, S. (2021). Course allocation among lecturers using Python. *Universiti Malaysia Terengganu Journal of Undergraduate Research,* 3(4). 127-136. https://doi.org/10.46754/umtjur.v3i4.246
[20] Bogdanchikov, A., Zhaparov, M., & Suliyev, R. (2013). Python to learn programming. Journal of Physics: Conference series 423 (2013) 012027. https://doi.org/10.1088/1742-6596/423/1/012027