University of Zagreb

Faculty of Electrical Engineering and Computing

Department of Applied Computing

**Instructions for performing laboratory exercises**

**and**

**tests via the Edgar system**



Zagreb, 2023.

Content

[1 Edgar system 1](#_Toc84151302)

[1.1 Login to the system 1](#_Toc84151303)

[1.2 Using Edgar for Introduction to Programming 2](#_Toc84151304)

[2 Exams 2](#_Toc84151305)

[2.1 Tests with program tasks 4](#_Toc84151306)

[2.2 Completion of writing and submission of the test for assessment 7](#_Toc84151307)

[3 Ticketing system 9](#_Toc84151308)

[3.1 View own tickets 10](#_Toc84151309)

# Edgar system

The Edgar system is used for automatic assessment of tests of theoretical and practical knowledge of the course *Introduction to Programming.* In this course, you will encounter programming tasks, where for a given problem it is necessary to write a program solution in the C programming language.

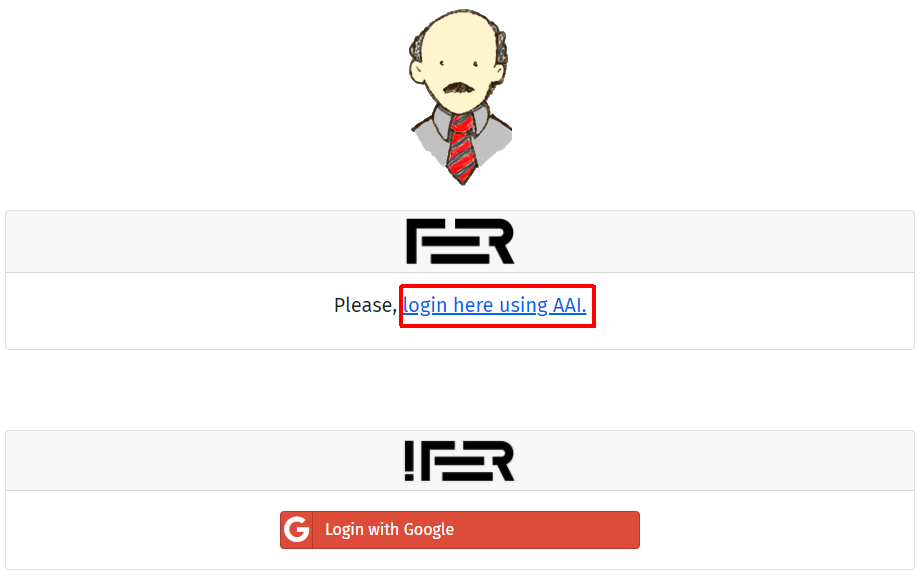
The labs contain program tasks that are solved and tested in the Notepad editor (++) or the VSC development environment, and the final solution is transferred (*copy-paste*) to the Edgar system.

You can access the Edgar system at: <https://edgar.fer.hr/>

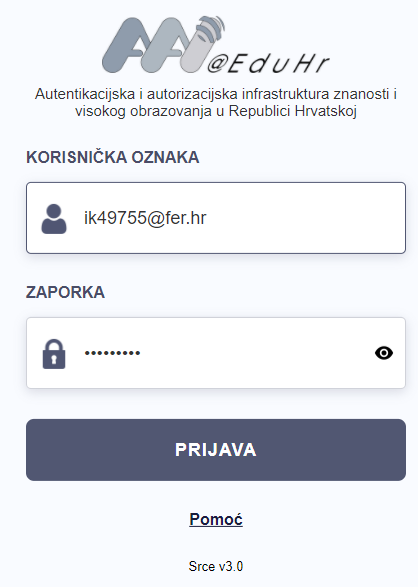
*Note:* The layout of the graphical user interface elements on the current version of the system Edgar may differ in part from the appearance of the user interface shown in the illustrations later in this document. In case of any ambiguities and/or difficulties when using the system, contact the assistant on duty.

## Login to the system

The username and password from the *AAI*-Edu system are used to log in to the system. You can access this loginmode by clicking on "*login here using AAI*", as shown in the figure:



Then you need to enter the AAI username and password in the login fields and click „PRIJAVA“:



*Note*: The username from the AAI-Edu system is in the form:

*your\_user\_name*@*your\_institution*.hr

## Using Edgar for Introduction to Programming

In order to get access to the tasks and tests in the course Introduction to Programming for the current academic year in Edgar, it is necessary to select the appropriate course and academic year from the drop-down menus on the right side of the main menu:

A screenshot of a computer

Description automatically generated

# Exams

**Note**: Official cheat-sheet printed on paper may be used in laboratory exercises. It will also be available in Edgar as a last question in the laboratory exercises and exams:

* [Cheat sheet for C](https://teams.microsoft.com/l/file/9CC0483D-04D2-4835-9B35-9AD224A7AF94?tenantId=ca71eddc-cc7b-4e5b-95bd-55b658e696be&fileType=docx&objectUrl=https%3A%2F%2Fferhr.sharepoint.com%2Fsites%2FIPRO2020%2FClass%20Materials%2FInstructions%2FCheatSheetC.docx&baseUrl=https%3A%2F%2Fferhr.sharepoint.com%2Fsites%2FIPRO2020&serviceName=teams&threadId=19:f5640862daa040309ccda4201c466ec8@thread.tacv2&groupId=753c7502-ec97-4f48-9130-d6896b3e521d)

Knowledge tests in laboratory exercises are accessed using a password. Knowledge tests are only available in a pre-set time intervals.

The system then asks the user to enter a password to access the exam. In the lab exercises, you will learn the passwords for the exams from the assistant on duty just before you start writing each exam.

A screenshot of a computer

Description automatically generated

The test is commenced by the option „*Start exam*“.

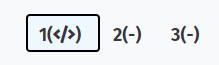
To practice working with the system at home, a *demo* test has been defined. This test is not graded and can be run an unlimited number of times. The password is **not needed**. The test is available under “Public exams”.

A screenshot of a computer

Description automatically generated



Links at the top of the screen are used to navigate between tasks in the tests. In addition to the ordinal number of each task, the status of the task is shown in parentheses. Next to the ordinal number of the questions that have not yet been answered, a *minus* sign is displayed. If the answer to the question is a software solution that has already been written, it will be marked with *</>*.

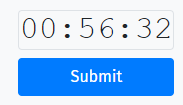


With each task, information is shown regarding the points that the student earns for correct, unanswered and incorrect answer.

A green and grey square sign

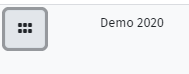
Description automatically generated

In the upper right corner there is a clock that counts down the time until the end of writing the test, a picture of the student and the "**Submit"** button which is used to submit the entire test for the final assessment (not individual tasks!).



The student is obliged to submit the test for assessment by the end of the countdown, but it will be possible to submit the test for assessment in a reasonable time after the expiration. This timeout, however, will be recorded in the system, so the test can be canceled later.

In the upper left corner (next to the test name) there is a button for selecting the layout of the screen:



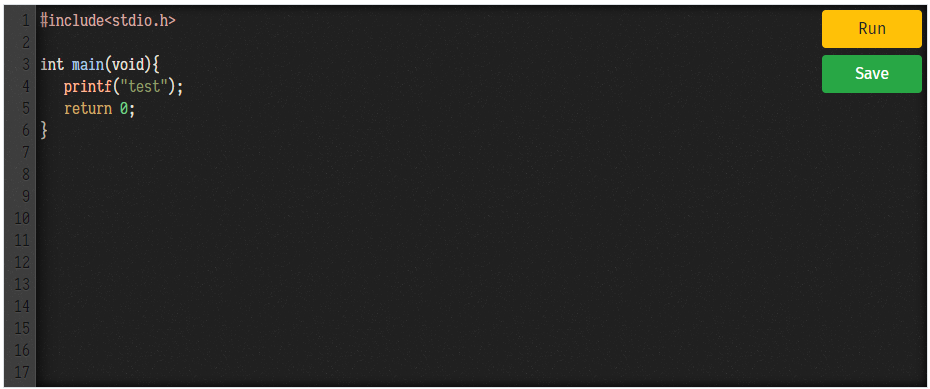
## Tests with program tasks

**Important:** The program task must first be prepared and tested outside Edgar. This means that before submitting a solution to the Edgar system, the program needs to be written (eg with the Notepad/++ editor or in the VSC), compiled (by calling the *gcc* compiler), all errors and warnings must be resolved, and the program should be thoroughly tested by executing against different test data. Only when you consider your solution to be entirely correct, you should *copy-paste* the code into Edgar.

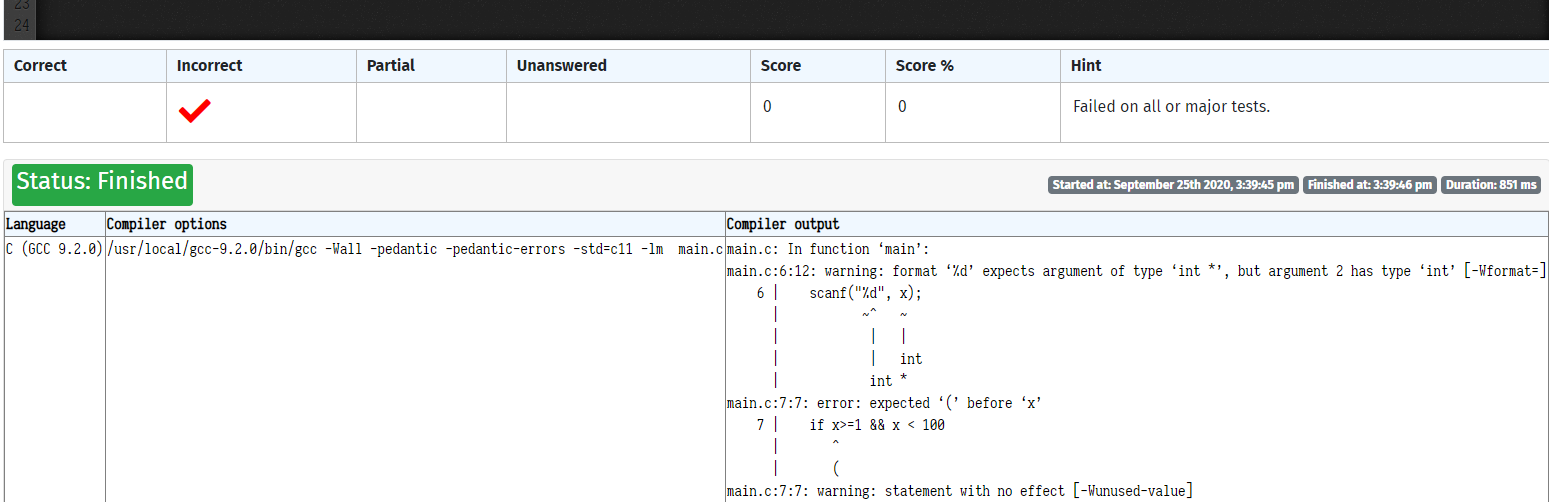
In tasks that require writing of a program, in the Edgar system, below the text of the task, there is a field to which the solution (program code) should be pasted.

The program code transferred to the Edgar system can be run in the following ways:

* by selecting the option „*Run*“
* while the program code field is in focus (the mouse pointer is in the field), by a key combination *Ctrl + Enter*



If errors or warnings occur during the compilation of the source code*,* the error/warning reported by the *gcc* compiler is shown*,* as illustrated by the example below in which there is a warning and a syntax error in the program.



If the compilation of the program is successful, the program will be run on several pre-prepared test cases. If the printout of the program for a case does not correspond to the expected printout, a certain percentage, determined in advance for that case, is deducted from the total number of points provided for the task. This percentage can be 100% if it is a typical test case, for example if a program that determines whether a given number is within the required interval does not work well for numbers inside or outside the interval, or less than 100% if the test case examines program behavior at the boundaries of the interval. The total deducted percentage of points for a task cannot exceed 100%.

The process of writing tasks in the Edgar system will be illustrated by one of the tasks from the demo test, which reads:

Write aprogram that prints the message as in the examples that follow, reads integers *a* and *b*, prints them separated by asingle space, and below that, prints their sum in anew line.

*Example 1:*

Hello·World!·Now·enter·a·and·b·>·15·20↵

15·20↵

35

*Example 2:*

Hello·World!·Now·enter·a·and·b·>·-5·20↵

-5·20↵

15

Mark `↵` in the examples indicates that a jump to new line (*line feed*) should be printed at that place or that the key *Enter* or *Return* should be pressed. Mark `·` indicates that a space should be printed at that place.

Try to *copy-paste* the following program into the solution field:

#include <stdio.h>

int main(void) {

int a, b;

printf("Hello World! Now enter a and b > ");

scanf("%d %d", &a, &b);

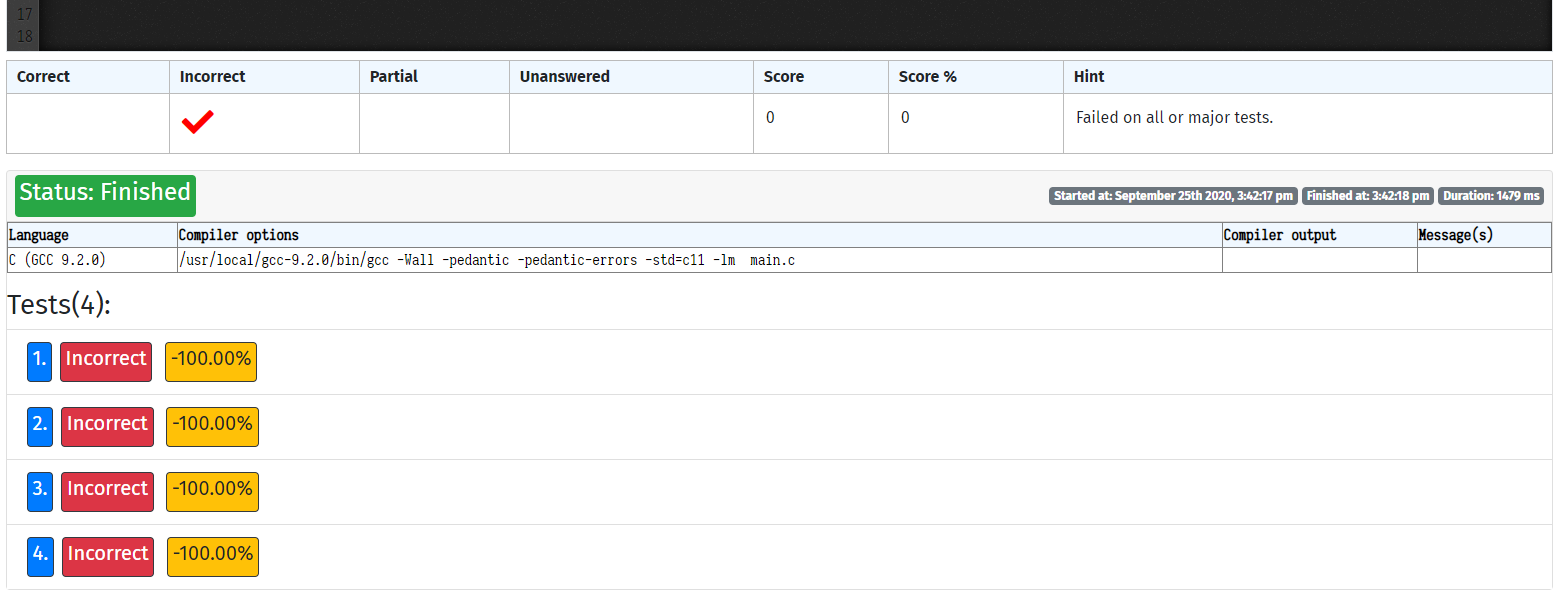
printf("%d %d", a, b);

return 0;

}

This program prints the message `Hello·World!·Now·enter·a·i·b·>·`, reads two integers and then prints them **separated by two spaces**. In addition, it does not print the sum.

If such a program is executed*,* Edgar will report that the result is incorrect for each of the four pre-prepared test cases:



Each row of the table corresponds to one test case. "*Incorrect"* indicates that the printout of the program was not as expected for that test case. The percentage next to the test case indicates what percentage of the predefined number of points for that task is lost if the solution does not pass that test. In this particular case, for any of the test cases in the example above, all points for that task are lost. So, in order to achieve 100% points on a task, the solution must be correct for all test cases prepared for that task.

Now try replacing the statement which prints the results with the following:

printf("%d %d %d", a, b, a + b);

The result of running the program will be

Hello·World!·Now·enter·a·and·b·>·15·20↵

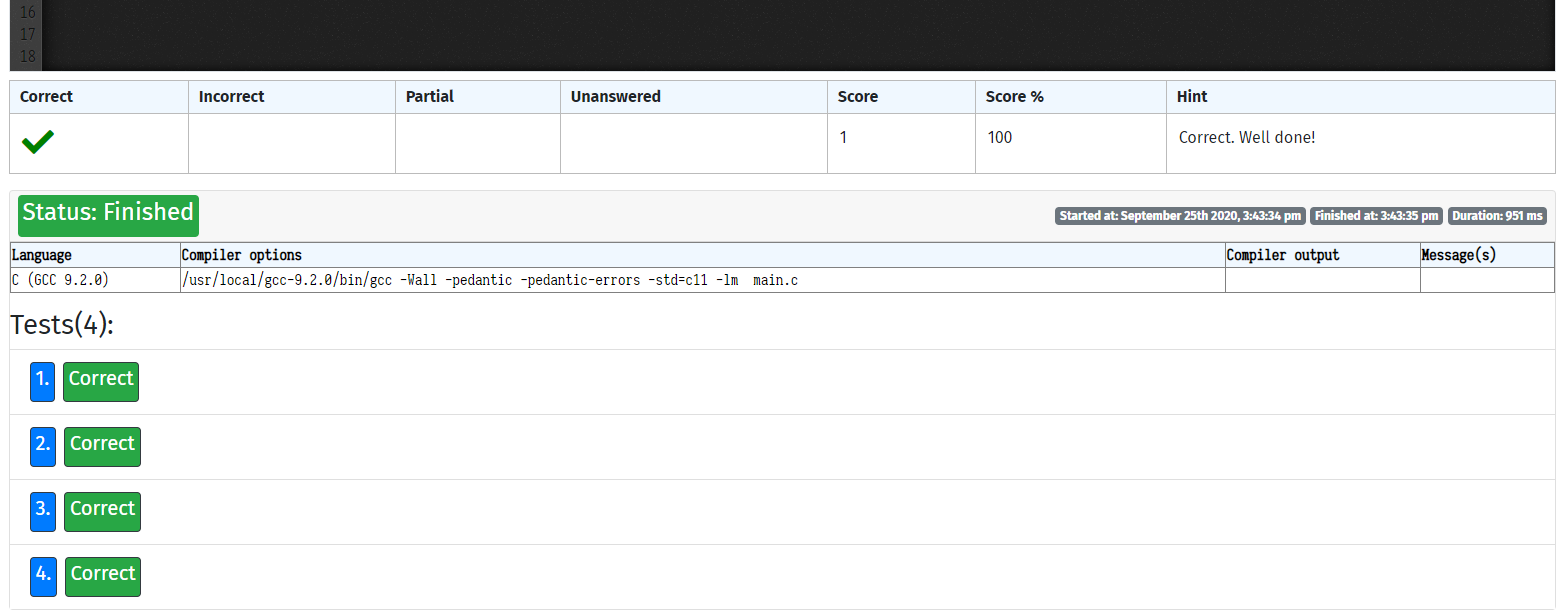
15·20·35

The program prints the correct values of the entered variables and correctly calculates and prints their sum, but this solution is also evaluated as incorrect in all cases, because there is no transition to a new line. One space more or less than required, or one jump to new line more or less than required, is enough for an almost correct solution to fail on all test cases.

Only if the print statement is corrected to

printf("%d %d\n%d", a, b, a + b);

the program will return the correct result for all test cases:



In the event of an unforeseen circumstance (eg power failure), Edgar allows the current contents of the solution field to be stored:

* **Explicitly -** the contents of the field can be explicitly saved by clicking the "Save" button in the upper right corner (below the "Run" button).
* **Implicitly** - Edgar automatically saves the contents of the field each time it is started with the "*Run*" option. Therefore, "*Save*" should only be used if you have written program code and have not run it.

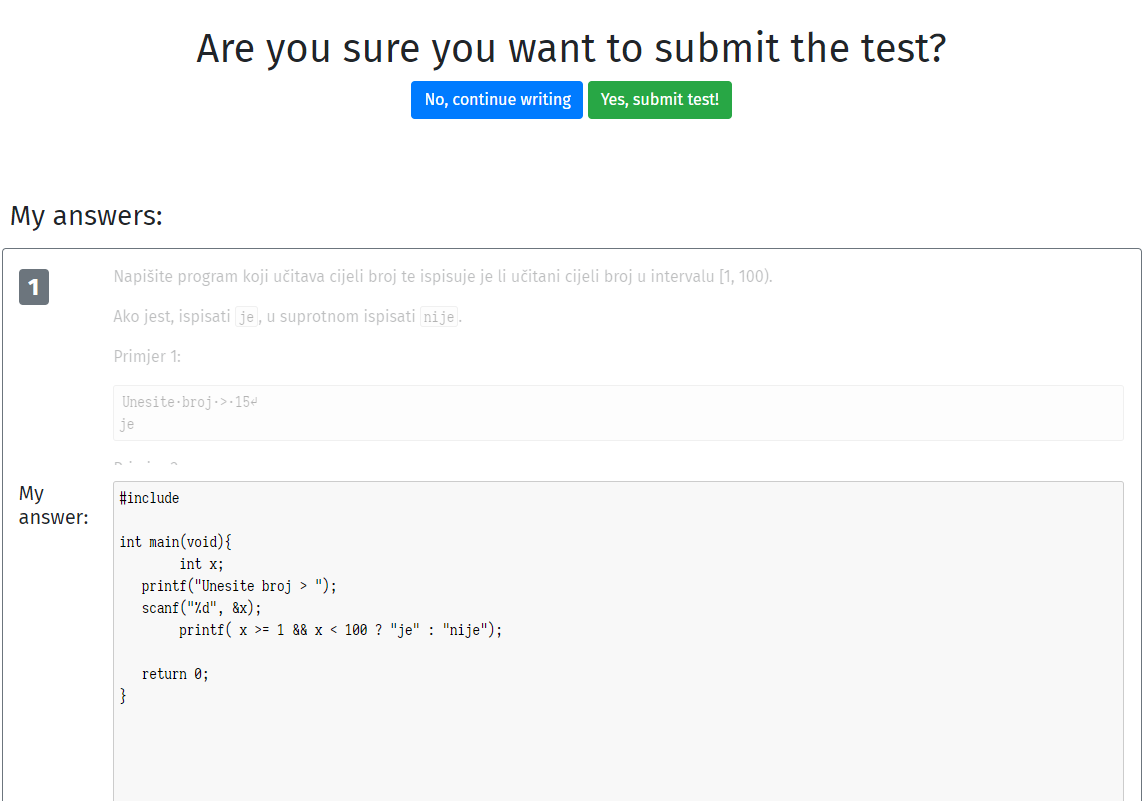
If an interrupt occurs, the test can later be re-accessed. By selecting the option "*Start exam (or continue ongoing exam*)" from the main menu, the system will display all tests that you have started but have not submitted, as well as the due time for the completion of the test *(Available until*). By entering the appropriate password, work on the interrupted test continues.

Graphical user interface, application

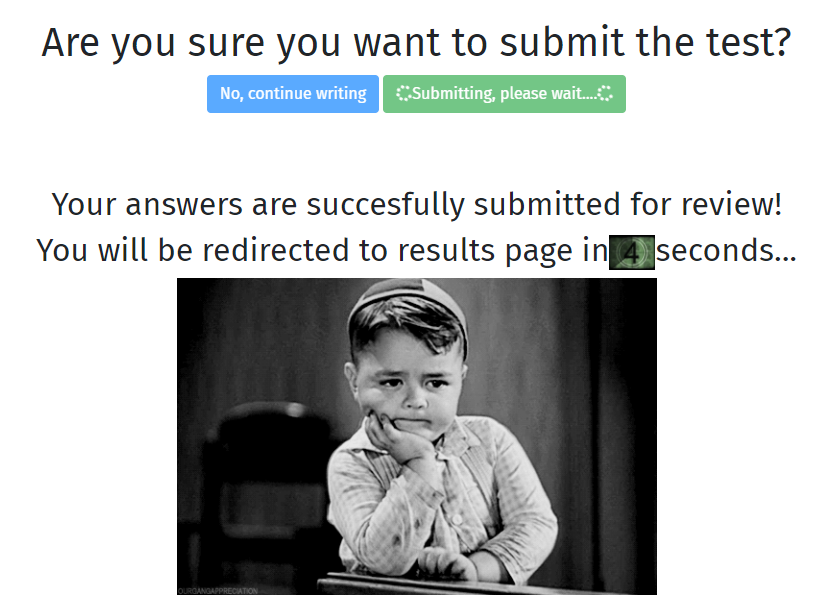
Description automatically generated

## Completion of writing and submission of the test for assessment

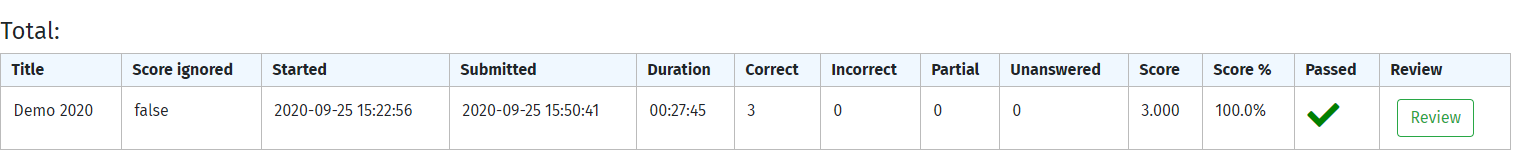
After completing all the tasks, the test should be submitted for the final assessment by selecting the "*Submit*" option in the upper right corner of the window. The submission is confirmed once again by selecting "*Yes, submit test!*", but can also be canceled by selecting "*No, continue writing*". Also, the screen will show your solutions to individual tasks, which should be visually inspected before final submission.



Then it is necessary to wait for the completion of the evaluation:

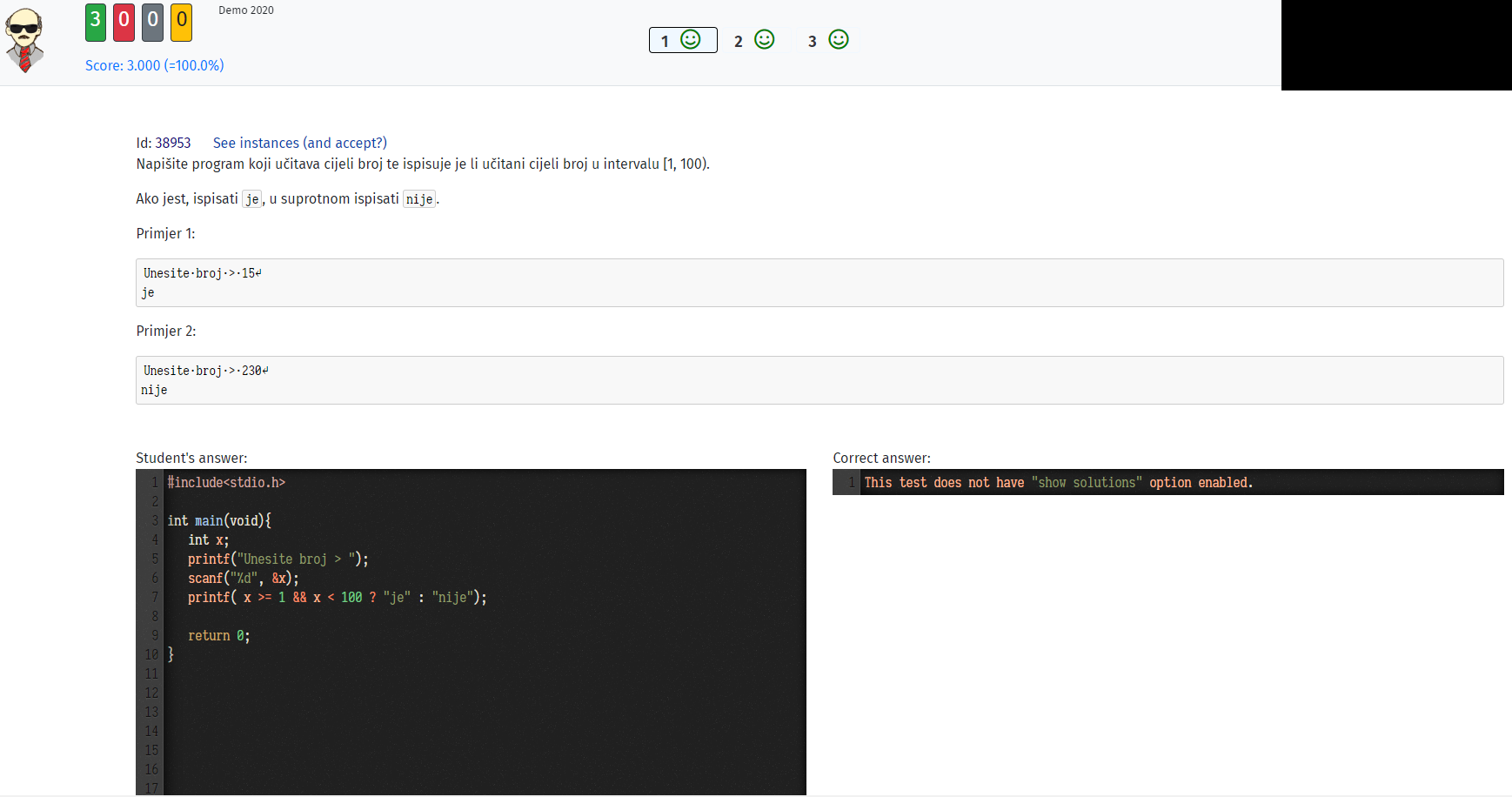


After the evaluation, the summary results of the test will be displayed. Within 10 minutes of submission, the system provides an overview of the assessment of individual tasks. The overview of individual tasks is accessed by selecting the option „*Review*“:



After the 10-minute period from the submission of the test, the review will be disabled, but the summary status of the test will still be displayed, as in the picture above, but without the option „*Review*“.

The solutions of individual tasks can be viewed on the following screen:



An overview of the **summary** results of all tests performed so far can be obtained at any time by selecting the option „*My previous exams*“.

# Ticketing system

In order to facilitate communication between students and teaching staff during the writing of tests, Edgar offers a support system called *Ticketing system*. In case of doubt about the accuracy of the solution or something else, the student can ask a question related to a specific task.

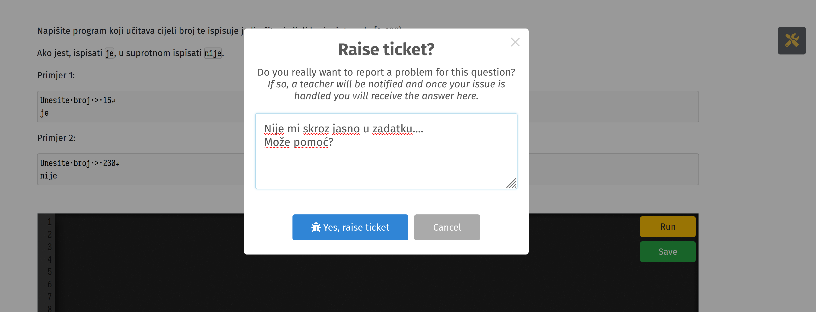
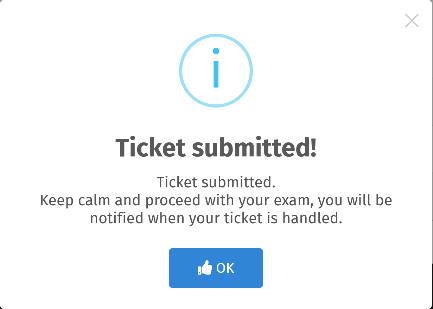
Using the system while writing the test is described below.

1. Click on the icon in the upper right corner of the question page

Graphical user interface, text, application

Description automatically generated

1. Enter a description of the problem in the window that appears and confirm it by clicking *Yes, raise ticket*. Then continue with the test*,*  without waiting for an answer.



1. The teaching staff will examine the problem as soon as possible and answer the student. While waiting for an answer, the student is not able to send additional messages related to the same task (he can send messages about other tasks) - the icon turns red, and an hourglass icon appears next to it.

Icon

Description automatically generated

After the the problem has been addresses, a message appears on the student's screen.

Graphical user interface, text, application, chat or text message

Description automatically generated

1. The student can then return to the appropriate question and check for the answer. The answer can be viewed by clicking on the green *cloud* icon. **Remark**: Only after the teaching staff answers the question related to the task, the student can ask the next question about the same task if necessary.

Graphical user interface, text, application

Description automatically generated

## View own tickets

Open and resolved tickets can be viewed the menu option ***My tickets.***

A picture containing graphical user interface

Description automatically generated

Graphical user interface, text, application

Description automatically generated

This option is useful when the question asked by the student cannot be answered quickly (in time

duration of the check) because it provides the possibility of subsequent review of the answers of the teaching staff. For example, there is a (fatal) mistake on the part of the teaching staff, and because of that, it is subsequent to all students answer recognized as correct.

**Note**: It is not possible to ask questions on the **ITP Demo** test.