



# IB mokinio baigiamojo pažymio skaičiuoklė (Final grade calculator for IB learner)

Ainas Puidokas TB1



XX TARPTAUTINĖ KONFERENCIJA „MOKSLEIVIŲ MATEMATIKOS, INFORMACINIŲ  
TECHNOLOGIJŲ IR EKONOMIKOS PROJEKTINIAI DARBAI“  
2021-03-05

## Tikslas

ir

## Uždaviniai

- Sukurti funkcionalią programą, kuri galėtų apskaičiuoti IB studentų egzaminų gautų pažymių suminius vidurkius.
  - Padaryti „*easy-to-understand*“ aplinką bet kokiam „*user*“ darant didelius mokinių sąrašus su jų gautais rezultatais.
- Sukurti mokinių vidurkių skaičiavimo algoritmą ir pritaikyti IB vertinimo formulę.
  - Išmokti naudotis įvairiais GUI (Grafinės sąsajos) komponentais ir naudojantis IDE (Integruota kūrimo aplinka) sukurti galutini produktą.
  - Testuoti programos funkcionalumą ir efektyvumą.

# Apie programos kūrimą

- Rašyta **Java** kalba
- Naudotas „*Intelij IDEA*“ IDE (Integruota kūrimo aplinka)
- GUI (Grafinė sąsaja) kūrimui naudojama JavaFX biblioteka
- Praleista apie 30+ val. kuriant šios programos GUI
- Programa bendrai sudaryta iš 850 eilučių

```
<BorderPane fx:controller="com.AssessmentCalculator.Controller"
  xmlns:fx="http://javafx.com/fxml">

  <top>

    <HBox fx:id="topbox" spacing="15" alignment="top_right"
      style="-fx-background-color: grey; -fx-border-color: black; -fx-border-width: 2">
      <padding>
        <Insets top="10" right="5" bottom="10"/>
      </padding>
      <Button fx:id="aboutbutton" text="About" prefWidth="100" textFill="darkblue"
        style="-fx-background-color: darkgrey; -fx-border-color: black"
        onAction="#onAboutClicked">
        <font>
          <Font name="Arial bold" size="14"/>
        </font>
        <graphic>
          <ImageView>
            <Image url="@/toolbarButtonGraphics/general/Information24.gif"/>
          </ImageView>
        </graphic>
      </Button>

      <Button fx:id="helpbutton" text="Help" prefWidth="100" textFill="black"
        style="-fx-background-color: darkgrey; -fx-border-color: black"
        onAction="#onHelpClicked">
        <font>
          <Font name="Arial bold" size="14"/>
        </font>
      </Button>
    </HBox>
  </top>
</BorderPane>
```

# Naudoti programos kūrimui GUI komponentai

```
Adomas Počius SL 6.0 6.0 0.0 7.0
Ainas Butkus SL 5.0 7.0 0.0 7.0
Ainas Puidokas SL 7.0 4.0 0.0 5.0
Benas Ramanauskas SL 4.0 6.0 0.0 6.0
Gintarė Kazlauskaitė SL 7.0 6.0 0.0 4.0
Ieva Urbonaitė HL 5.0 7.0 7.0 6.0
Jokūbas Kučinskas HL 5.0 4.0 6.0 6.0
Jonas Žemaitis SL 3.0 3.0 0.0 5.0
Kamilė Kazlauskaitė HL 7.0 4.0 5.0 6.0
Lukas Adomaitis SL 4.0 6.0 0.0 7.0
Milita Savickaitė HL 2.0 2.0 2.0 2.0
Nojus Balčiūnas HL 4.0 5.0 5.0 5.0
Paulius Paulauskas HL 6.0 5.0 5.0 6.0
Sofija Sakalauskaitė SL 5.0 4.0 0.0 5.0
Tomas Šimkevičius SL 5.0 3.0 0.0 4.0
```

Button

Vbox

TextField

ChoiceBox

RadioButton

Hbox

TextArea

TableView

```
@Override
public void initialize(URL url, ResourceBundle rb) {

    name.setCellValueFactory(new PropertyValueFactory<>(s: "studentName"));
    level.setCellValueFactory(new PropertyValueFactory<>(s: "Level"));
    p1.setCellValueFactory(new PropertyValueFactory<>(s: "P1"));
    p2.setCellValueFactory(new PropertyValueFactory<>(s: "P2"));
    p3.setCellValueFactory(new PropertyValueFactory<>(s: "P3"));
    internal.setCellValueFactory(new PropertyValueFactory<>(s: "Internal"));
    averages.setCellValueFactory(new PropertyValueFactory<>(s: "Averages"));

    dataTable.setItems(observableList);
    dataTable.setEditable(true);

    name.setCellFactory(TextFieldTableCell.forTableColumn());
    level.setCellFactory(TextFieldTableCell.forTableColumn());
    p1.setCellFactory(TextFieldTableCell.forTableColumn(new DoubleStringConverter()));
    p2.setCellFactory(TextFieldTableCell.forTableColumn(new DoubleStringConverter()));
    p3.setCellFactory(TextFieldTableCell.forTableColumn(new DoubleStringConverter()));
    internal.setCellFactory(TextFieldTableCell.forTableColumn(new DoubleStringConverter()));

    while (load.hasNext()) {

        String name = load.next();
        String surname = load.next();
        String fullName = name + " " + surname;
        String level = load.next();
        double p1 = Double.parseDouble(load.next());
        double p2 = Double.parseDouble(load.next());
        double p3 = Double.parseDouble(load.next());
        double internal = Double.parseDouble(load.next());

        Calculator newData = new Calculator(fullName, level, p1, p2, p3, internal, Averages: 0);
        dataTable.getItems().add(newData);
        dataTable.getColumns().get(6).setVisible(false);
    }
}
```

# Išvados

- Programa lanksčiai atlieką savo užduotį bei ją atlieką efektyviai.
- Viskas išsaugoma bei pakeičiama, todėl galima laikyti, kaip skaitmenini mokinių sąrašą.
- Mokytojai galėtų naudotis programą savo darbe, jei nori greitai ir efektyviai sudaryti skaitmeninius mokinių sąrašus, per kuriuos galėtų greitai apskaičiuoti egzaminų vidurkius bei palyginti mokinių pasiekimus.
- Gana plačiai pagilinau savo programavimo žinias **Java** kalba bei išmokau sukurti programos GUI (Grafinė sąsaja).
- Sužinojau, kad norint sukurti kokybiškai atrodančią programą, kuri veikia be priekaištų nėra visai lengvas dalykas ir tam reikia įdėti labai daug laiko bei darbo.

# Klausimai





**About:** Find out information about the program

**Add:** Add a new data column for writing in new details

**Clear:** Clear out selected row and delete it

**Check:** Calculate the averages after writing all the details in

**Close:** Close the application

To use the application, fill in the blank details up to any desirable amount of columns, then click the check button to calculate the averages

**Student Full Name:** Enter the full name of a student

**Level:** Enter class level of the student (either SL or HL)

**Paper1:** Enter for both levels grade of Paper1

**Paper2:** Enter for both levels grade of Paper2

**Paper3:** Enter only for HL level grade of Paper3 (For SL write in 0)

**Internal:** Enter for both levels grade of internal assessment

OK

Full Name	Level	P1	P2	P3	Internal	Average grades
Adomas Počius	SL	5.0	7.0	0.0	7.0	6.33
Ainas Dvidokas	SL	7.0	4.0	0.0	5.0	5.33
Benas	SL	4.0	6.0	0.0	6.0	5.33
Gintarė	SL	7.0	6.0	0.0	4.0	5.67
Ieva Urbanaitė	HL	5.0	7.0	7.0	6.0	6.25
Jokūbas	HL	5.0	4.0	6.0	6.0	5.25
Jonas Žemaitis	SL	3.0	3.0	0.0	5.0	3.67
Kamilė	HL	7.0	4.0	5.0	6.0	5.5
Lukas	SL	4.0	6.0	0.0	7.0	5.67
Milėta Savickaitė	HL	2.0	2.0	2.0	2.0	2.0
Nojus	HL	4.0	5.0	5.0	5.0	4.75
Paulius	HL	6.0	5.0	5.0	6.0	5.5
Sofija	SL	5.0	4.0	0.0	5.0	4.67
Tomas	SL	5.0	3.0	0.0	4.0	4.0
Marta	HL	7.0	5.0	5.0	5.0	5.5
Laurynas Macernis	HL	4.0	6.0	7.0	5.0	5.5

Asmens duomenys

Full Name

HL

P1 (1-7)

P2 (1-7)

P3 (1-7)

Internal (1-7)

**Results:**

TB Average is 5.13 or 73.32%  
 HL Average is 5.03 or 71.88%  
 SL Average is 5.22 or 74.6%

Worst SL Student was Jonas Žemaitis with an average of 3.67  
 Worst HL Student was Milėta Savickaitė with an average of 2

Best SL Student was Adomas Počius with an average of 6.33  
 Best HL Student was Ieva Urbanaitė with an average of 6.25

Paper 1 Average for the class was 5.06, Paper 2 Average - 4.88  
 Paper 1 Average for SL was 5.11, Paper 2 Average - 5  
 Paper 1 Average for HL was 5, Paper 2 Average - 4.75, Paper 3 Average - 5.5

Hardest paper for the class - p2,  
 for SL - p2, for HL - p2

Easiest paper for the class - p3,  
 for SL - p1, for HL - p3

Full Name	Level	P1	P2	P3	Internal
Adomas Pocius	SL	6.0	6.0	6.0	7.0
Ainas Butkus	SL	5.0	7.0	0.0	7.0
Ainas Pocius	SL	7.0	4.0	0.0	5.0
Benas Pocius	SL	4.0	6.0	0.0	6.0
Gintaras Pocius	SL	7.0	6.0	0.0	4.0
Ieva Urnaitis	HL	5.0	7.0	7.0	6.0
Jokūbas Pocius	HL	5.0	4.0	6.0	6.0
Jonas Pocius	SL	3.0	3.0	0.0	5.0
Kamile Pocius	HL	7.0	4.0	5.0	6.0
Lukas Pocius	SL	4.0	6.0	0.0	7.0
Milita Pocius	HL	2.0	2.0	2.0	2.0
Nojus Pocius	HL	4.0	5.0	5.0	5.0
Paulius Pocius	HL	6.0	5.0	5.0	6.0
Sofija Pocius	SL	5.0	4.0	0.0	5.0
Tomas Pocius	SL	5.0	3.0	0.0	4.0
Matas Pocius	HL	7.0	5.0	5.0	5.0

Asmens duomenys

## Results:

Laurynas Mačernis

HL

4

6

7

5

 Grey Mode  
 Green Mode

Add

Clear

Check

Save And Exit