What are possible predictors for learning scripts-based programming in Leaphy for students?
Research Paper

Tim Zegeling[1] and Amos Daniël Roth[1] Flavio Sanchez Troise[2] Víctor Moya Alcaraz [4]

[1]Hogeschool Utrecht, University of Applied Sciences, Netherlands
[2]ESAIP Aix-en-Provence, École d'Ingénieur, France
[3]Universidad Politécnica de Valencia, Spain

tim.zegeling@student.hu.nl, daan.roth@student.hu.nl, flavio.sanchez-troise@student.hu.nl, victor.moyaalcaraz@student.hu.nl

- Title - Abstract - Introduction - Background/Lit review - Methods - Results - Discussion - Conclusion - Acknowledgment - References

1 Abstract

From the lectures abstract is : • Summary of entire paper Need to answer to: • What was the problem? • What was your research method? • What was your major findings, • What are your conclusions?

In this paper we have researched what possible predictors are for learning script-based programming using Leaphy robotics for younger students

Learning script-based programming can be a difficult task for younger people. This research is focused on possible predictors for learning script-based programming based on programming with Leaphy, an educational robot for students. In this research students got assignments and the students were observed when making the assignments, by making recordings of their screens. The different factors were taken into account when observing the research.

The research had multiple research methods. The data was collected by the following research methods: Observation, interviews and surveys. We have researched the difference between gender, the difference in age, motivation, previous skill level in block-based programming, motivation and confidence in programming.

The data that was collected, was analyzed with quantitative method to research the relation between the different predictors. A multiple linear regression analysis was used.

We think our results are able to be used in determining the predictors for learning script-based programming for younger students.

(add findings and conclusion) (add the lit.review part) (maybe add recommendations)

2 Introduction

Defines the research problem. After the introduction, the reader knows why you are investigating this particular topic, and what you are going to investigate. - about 4 paragraphs - Start with societal / scientific relevance - Narrow down to your specific case (2 paragraphs) - Final paragraph : what are you going to do

The acquisition of script-based programming skills presents significant challenges for students, and it has never been more crucial for children to learn programming. In today's society, where technology plays an increasingly prominent role, programming skills have become essential for future success.

The main goal of this research is to investigate potential predictors for learning script-based programming by utilizing Leaphy, an educational robot.

Identifying these predictors will assist educators in developing improved teaching materials for programming instruction to students. This, in turn, will enhance the effectiveness of programming education and yield better outcomes among students.

To achieve this, students were instructed to complete a form about their personal possible predictors and to do programming assignments that has been analyzed with to research the relation between the different predictors.

3 Background

The numbers of students learn about programming and new technologies in the classrooms is increasing. In this research we have researched for how can we help teachers improve their course material and increase the effectiveness of this material by researching what possible predictors influence their learning process. Mainly age, gender, motivation, current skill level, previous experience in peer-to-peer education, grade, experience in block-based programming, previous completed programming assignments as these are the predictors we could research in our short amount of time and limited previous research experience.

Finding these predictors will help teachers create better teaching materials for teaching programming to stu-

dents. This should increase the effectiveness in teaching programming, and allow for better results from the students. If block-based programming has a big influence on learning text-based programming for example, it might be worth keeping this in the educational material.

We found some previous research regarding this topic and have described it in our literature research section.

A few predictors like programming experience, self-esteem/confidence, age, gender and motivation are also used in this study. We have used these predictors to see if we had similar results.

Though most studies were taken on a bigger scale, with multiple lessons. We have limited our research to one lesson.

We hope our results can be used in education by allowing teachers to make better educational material backed by the results of our research paper.

4 Hypothesis(has to disappear as a section, maybe we can modify a little and move it to the discussion part)

We suspect that predictors like motivation and previous experience will have a greater effect than the others, but that these factors combined will amplify each other because we hypothesise that all these predictors are beneficial. This is because programming often has a bit of a barrier to break through. Previous experience and motivation will make this barrier smaller.

5 Literature review(has to be incorporate in the background)

We will discuss possible literature to aid us in our research.

Transitioning from Block-Based to Text-Based Programming Languages[1] is a paper relevant for us because of the research in transitioning from block-based programming to text-based programming. They found in this paper that certain block-based environments allow the students to complete the same tasks faster than their peers who learned using text-based environments. Because we will include this effect of previous experience in block-based programming as a predictor for learning and this paper discusses this including weaknesses and suggestions for improvements.

Transitioning from introductory block-based and text-based environments to professional programming languages in high school computer science classrooms[2] is a paper discussing the transition from introductory block-based and text-based programming to professional programming languages in school computer science classrooms. Some highlights of this study that are useful to our research are:

- Students who learned with a block-based programming environment outperformed peers who used an isomorphic text-based tool.
- There were no significant differences in attitudes between block-based and text-based learners 10 weeks into learning Java.
- The modality of the introductory tool (block or text-based) had no effect on student outcomes after 10 weeks of learning Java.

Student ability and difficulties with transfer from a block-based programming language into other programming languages: a case study in Colombia[3] is relevant for us since it is a study with children about block-based and text-based programming therefore we used it to determine how we can ease the transition and make more relevant and effective materials for teaching.

Investigating the Transition from Block-based to Text-based Programming Techniques in Secondary Education in Greece[4] is relevant for us in order to determine methods for researching the predicting factors.

No tests required: comparing traditional and dynamic predictors of programming success[5] is relevant for us because it includes possible predictors, and their research is similar to that of us.

Programming: factors that influence success[6] is relevant to us similar to the paper above. It helps us find predictors and correlations.

Predictors of success in a first programming course[7] is another paper discussing predictors and programming used to come up with predictors and information.

6 Research design and methods

6.1 Research approach

In this research project we have taken a quantitative approach in collecting data from students. We collected data using surveys, with Microsoft Forms. The students filled in the survey before they started the assignments, so that we had all the information about the predictors for learning script-based programming for the students. Those possible predictors are gender, age, motivation, interest, self confidence, previous experience in block based programming and previous experience in script based programming. We also collected results from assignments that students did. The students got assignments that got harder by every assignment. We used the assignments for this research by analyzing how good the students had done the assignments. We could see how long it took for them to finish the assignments and how many assignments they had finished.

The quantitative data consists of data we collect using the surveys before the assignment, and the data during the assignments.

It was initially planned that the students for this study would have been from the stakeholder's primary school, but due to problems with the few number of students that have participated, we needed to include other students for our study to do a quantitative data analysis. So we have done

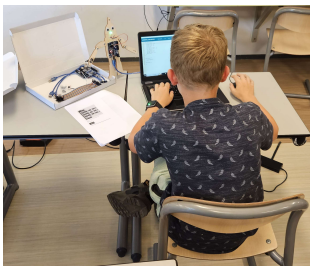


Figure 1. Research setup

6.2 Research methods, techniques and tools

The students are going to fill in the surveys and complete the assignments on a laptop computer where there will be a recording software, named OBS (Open Broadcaster Software), that will record their screens from the beginning until the end. In that way, the data will be anonymous, and it will let us find out which survey is related to which result without using personal information like names. For the older students the was observation live. By recording the data at the moment they did the survey and the assignments.

We used digital surveys Microsoft Forms to send out the surveys to the students. The survey is used to assess the current possible predictors.

We also gave assignments to the students and collect their results. We collected how many of the assignments the students were able to complete in the time frame, which is 30 minutes. We wrote down when these assignments were completed.

The data collected from these surveys is used to generate diagrams using Python and matplotlib, and then to do statistical analysis.

6.3 Assignments

To measure the ability of students to learn script-based programming there were used assignments. Those assignments were programming assignments that started with some basic assignments that progressed in difficulty. The form of those assignments was some block-based code that the students had to replicate and then they had to go to the script-based code to modify it. There were seven assignments.

The starting assignment was an exercise similar to the "hello world" exercise. For this first assignment the student needed to burn a led on the robot. The further the student got with the assignments the harder the assignments got. In the end the children needed knowledge about if statements and variables to pass the assignments.

6.4 Scope

For the research, Leaphy Flitz was used for students to learn programming, because Flitz is a robot that is easy to use for students. The students were given assignments that they tried to finish within the time limit of 30 minutes.

We have limited ourselves to two subject groups. One of the groups had students from 12-13 and the other group had older students from 20-24. The group with students from 12-13 are in secondary school. The group with students from 20-24 are international students. We used older students, because then we could use age as predictor. The total number of students was 14.

6.5 Research data management

We have anonymized this data and stored this as an excel spreadsheet or a csv file. This is stored in our Microsoft Teams files location assigned to us by the Hogeschool Utrecht. We have also stored this data in our private GitHub repository along with the source code to analyze the data.

We have also analyzed the data through the generated diagrams to have some answers for our research.

7 Results

A multiple linear regression analysis was used to see how the predictors from **??** affect the learning performance.. The overall regression was statistically significant (R2 = 0.797).

It was found that interest significantly predicted learning performance (= 5.97, p = -.197) in a positive way. A high self confidence seemed to have negative impact on the learning performance (= -5.534, p = 0.159).

It was found that motivation did not significantly predict exam score (= -0.175, p = 0.947). Age, block programming and code programming only seem to have milder effects compared to interest and self confidence but can still affect the end results.

The fitted regression model was LP (Learning Perfor-

Category	Coef	Standard error	t	p
Intercept	-59.5171	37.730	-1.577	0.166
age	3.2356	1.562	2.072	0.084
motivation	-0.1754	2.508	-0.070	0.947
interest	5.9699	4.115	1.451	0.197
block programming	2.4891	1.313	1.896	0.107
code programming	2.8111	1.547	1.817	0.119
self confidence	-5.5356	3.441	-1.609	0.159

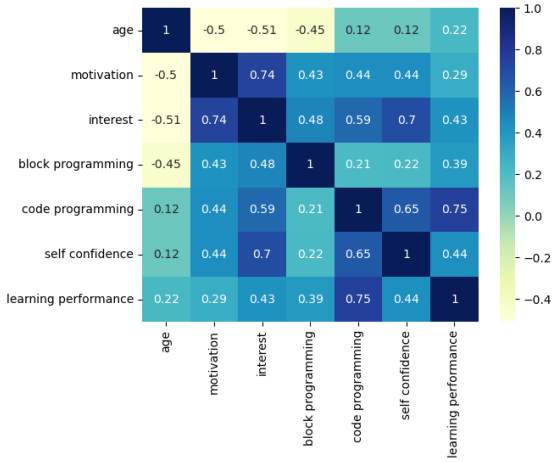
Table 1. Results

mance):

$$\begin{aligned} LP = -59.5171 &+ 3.2356 * \text{Age} \\ &+ -0.1754 * \text{Motivation} \\ &+ 5.9699 * \text{Interest} \\ &+ 2.4891 * \text{Block programming experience} \\ &+ 2.8111 * \text{Code programming experience} \\ &+ -5.5356 * \text{Confidence} \end{aligned}$$

In the heatmap2 is is shown that previous code programming experience heavily correlates with the total. More so than interest and self confidence.

Figure 2. Correlation Heatmap



8 Discussion

The aim of this study was to examine what individual student factors are for learning script-based programming for students. These factors predict the effectiveness in learning script based programming. To be precise, we compared the effectiveness of the student making block based and script based assignments with previous recorded predictors. We compared age, previous experience in block based programming, previous experience in code based programming, motivation and interest in technology and the students confidence in completing the assignments and researched whether this data has influence in learning to program based on the results of the assignments.

We discovered that previous experience in block-based and code-based programming is the main factor in predicting success. This fits in with the following papers: (Add feedback from the discussions from the social robotics event as per advice in class)

We discovered that interest and self confidence played the biggest parts in influencing the learning performance of the student. The interest having a significant influence could be explained by having more general technical knowledge. Though it is not visible in the formula itself, the heatmap shows a big correlation between interest and motivation.

One of the reasons why a high confidence might negatively affect the score is the Dunning–Kruger effect. Further research should perhaps analysis the previous experience of the student and compare it with the self confidence. A quick look at the heatmap shows previous block programming experience had a low correlation with self confidence while previous code programming experience showed a high correlation.

References

[1] Paul Denny Luke Moors, Andrew Luxton-Reilly. Transitioning from block-based to text-based programming languages. Online: https://ieeexplore.ieee.org/abstract/document/8753414, Accessed: 04-13-2023.

[2] Uri Wilensky David Weintrop. Transitioning from introductory block-based and text-based environments to professional programming languages in high school computer science classrooms. Online: https://www.sciencedirect.com/science/article/abs/pii/S036013151930199X, Accessed: 04-13-2023.

[3] Valeria Guerrero-Bequis Alejandro Espinal, Camilo Vieira. Student ability and difficulties with transfer from a block-based programming language into other programming languages: a case study in colombia. Online: https://www.tandfonline.com/doi/abs/10.1080/08993408.2022.2079867, Accessed: 04-13-2023.

[4] Christos Douligeris Eleni Seralidou. Investigating the transition from block-based to text-based programming techniques in secondary education in greece. Online: https://ej-eng.org/index.php/ejeng/article/view/2753, Accessed: 04-13-2023.

[5] Jamie L. Godwin Christopher Watson, Frederick W.B. Li. No tests required: comparing traditional and dynamic predictors of programming success. Online: https://dl.acm.org/doi/abs/10.1145/2538862.2538930, Accessed: 04-13-2023.

[6] Ronan Reilly Susan Bergin. Programming: factors that influence success. Online: https://dl.acm.org/doi/abs/10.1145/1047344.1047480, Accessed: 04-13-2023.

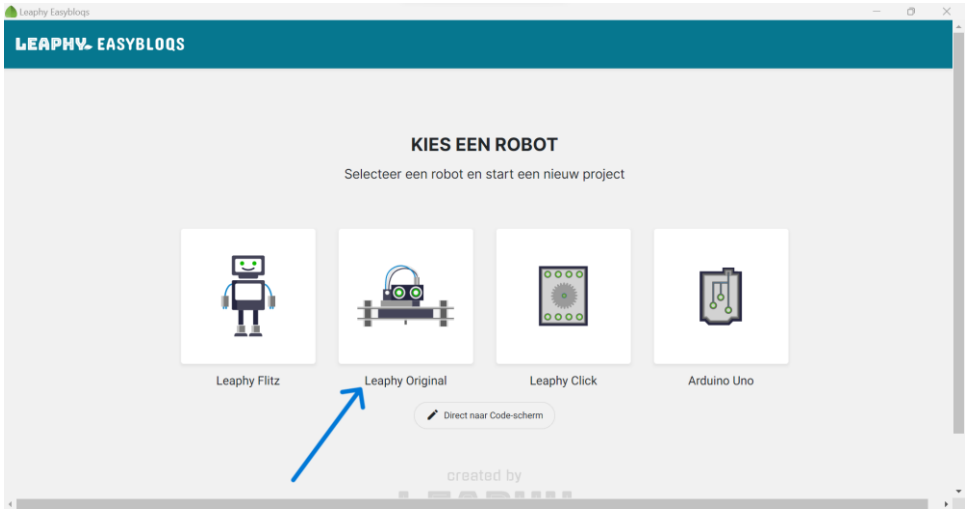
[7] Bob Baker Ilona Box Quintin Cutts Michael de Raadt Patricia Haden John Hamer Margaret Hamilton Raymond Lister Marian Petre Ken Sutton Denise Tolhurst Jodi Tutty Sally Fincher, Anthony Robins. Predictors of success in a first programming course. Online: https://research.usq.edu.au/item/9y1qv/predictors-of-success-in-a-first-programming-course, Accessed: 04-13-2023.

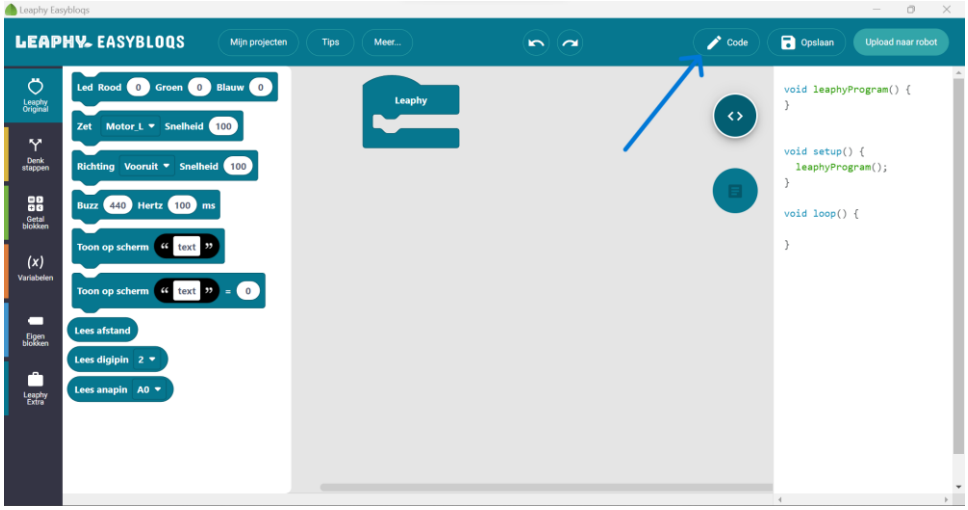
9 Appendix

Teaching childeren programming: How?

Assignments:

First assignment:





Start with something simple, maybe hello world like program? Just turn on a led or something.

Om de led te laten branden op de Leaphy robot moet er gebruik worden gemaakt van de setLed() functie. Deze setLed() functie moet worden toegevoegd in ‘void leaphyProgram(){}’. Om de setLed() functie te gebruiken moeten er 3 parameters worden meegeven: setLed(rood, groen, blauw). Als je de led een rode kleur wilt geven moet de parameter ‘rood’ op 255 worden gezet en moeten de parameter ‘groen’ en parameter ‘blauw’ allebei op 0 worden gezet, zoals in het voorbeeld hieronder:

```
setLed(255, 0, 0);
```

Voeg deze functie nu toe aan de ‘leaphyProgram(){}’ tussen de twee accolades (een accolade is ‘{‘), zoals hieronder:

```
void leaphyProgram() { setLed(0, 0, 0); }
```

Druk nu op de ‘Upload naar robot’ knop en zie of de led op de robot rood wordt.

Teach them loops? The Leaphy software needs a main loop to work.

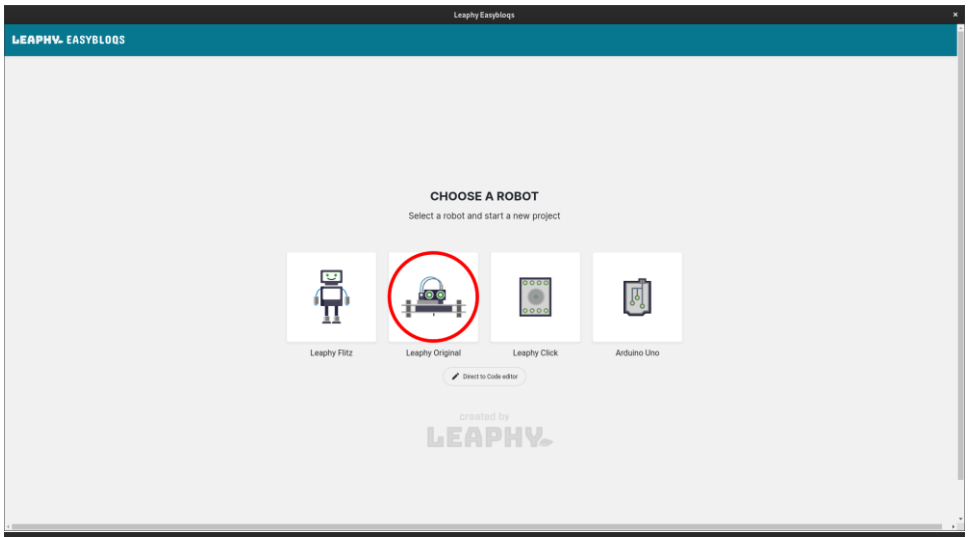
```
for (int count = 0; count < 10; count++) { }
```

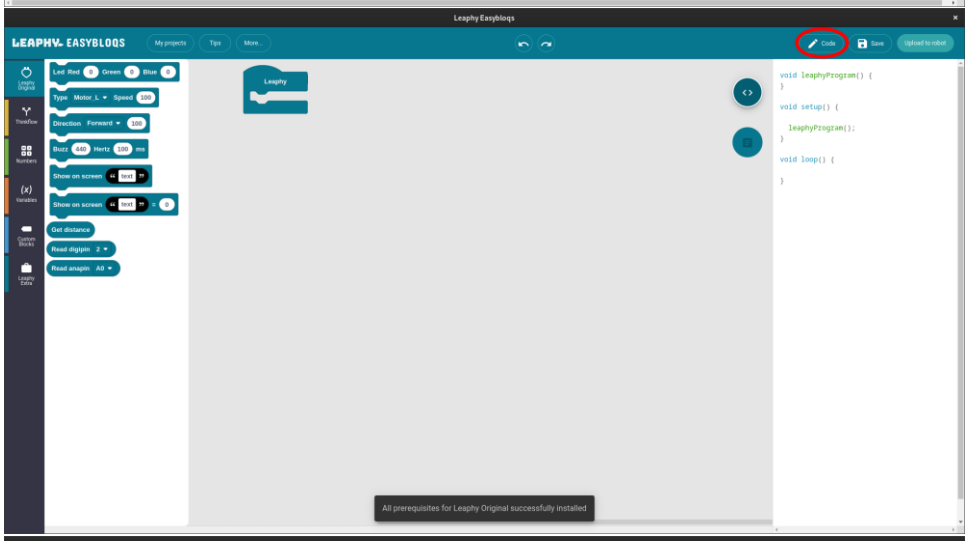
Maybe teach basics of variables? Though Oliver told us variables are hard for beginners, we kind need them for if else statements.

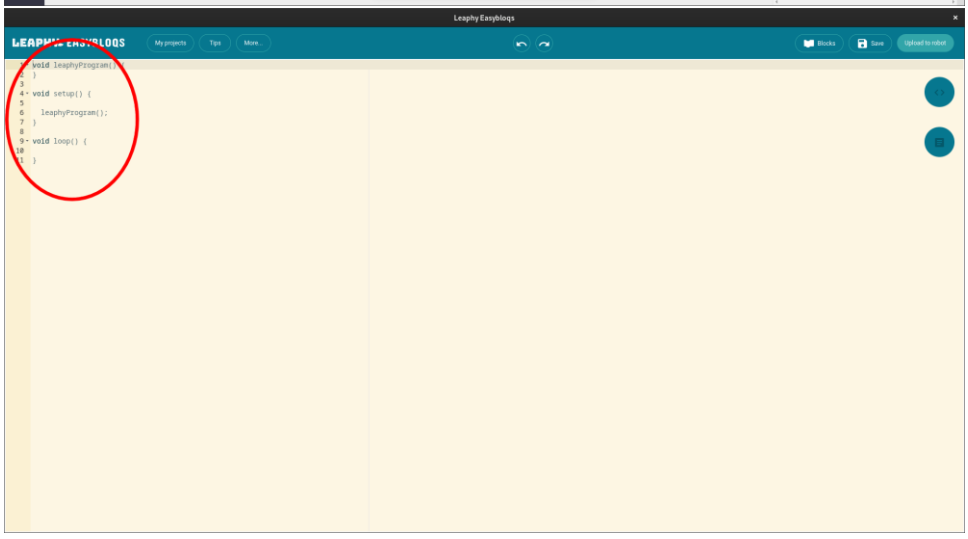
Teach them if else statements maybe?

```
if (false) { } else { }
```

Make the assignments a bit harder by combining basic math with the if else statements maybe?


Leaphy Easybloqs
LEAPHY EASYBLOQS
CHOOSE A ROBOT
Select a robot and start a new project
Leaphy Flitz
Leaphy Original
Leaphy Click
Arduino Uno
Direct to Code editor
created by
LEAPHY


Leaphy Easybloqs
LEAPHY EASYBLOQS
My projects
Tips
More...
Code
Save
Upload to robot
Led Red 0 Green 0 Blue 0
Type Motor_L Speed 100
Direction Forward 100
Buzz 440 Hertz 100 ms
Show on screen text
Show on screen text = 0
Get distance
Read digipin 2
Read anapin A0
Leaphy
void leaphyProgram() {
}
void setup() {
leaphyProgram();
}
void loop() {
}
All prerequisites for Leaphy Original successfully installed


Leaphy Easybloqs
My projects
Tips
More...
Blocks
Save
Upload to robot
void leaphyProgram() {
}
void setup() {
leaphyProgram();
}
void loop() {
}

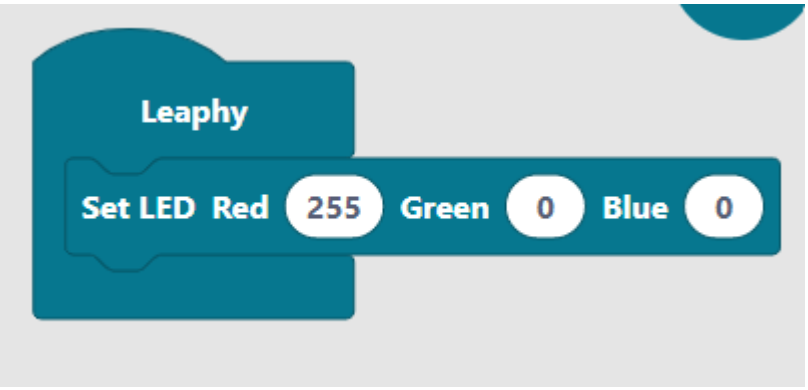
Level 1 – LED laten branden.

In this task, you will flash an RGB LED (light) to get familiar with the Easybloqs environment and the Arduino. RGB stands for Red, Green, Blue.

Level 1.1 – Neus led laten branden

Connect the LED as shown in the diagram below. The legs of the LED-have two different lengths. The long leg must be put in box at 13 and the short leg in GND.

Open the EasyBloqs environment, create the following setup and upload it to the Arduino:



If everything has been done correctly, this LED will light up red.

Level 1.2 – Kleur aanpassen

The following can be seen on the right side of the EasyBloqs environment:

```
void leaphyProgram() {
  analogWrite(3, 255);
  analogWrite(5, 0);
  analogWrite(6, 0);
}


void setup() {

  leaphyProgram();
}

void loop() {

}
```

The important thing for now is the setLed(255, 3, 3);

This is a piece of code that is responsible for making the LED light up.

“3” is the first parameter of the function. This indicates to which pin/cell to write. The second parameter is “255”. This indicates what should be written to this pin. This has a direct relationship with what has been filled in for the blocks in the previous assignment.

Change Red to 0, and Green to 255.

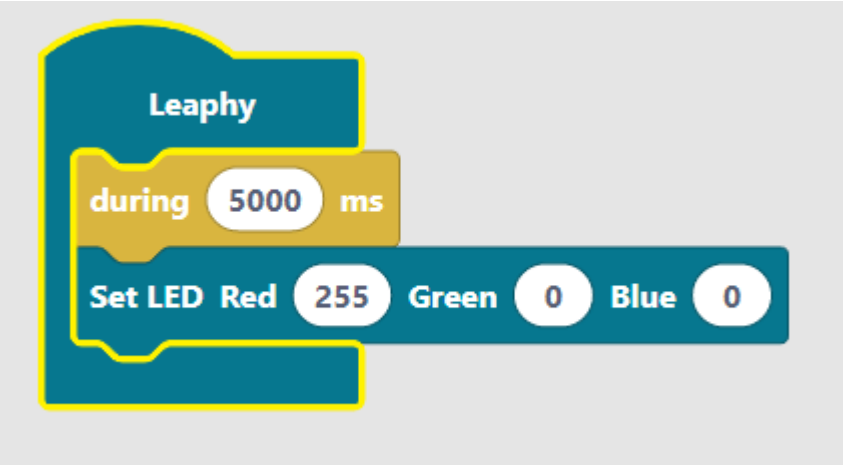
Write down what happens.

Level 2 – Vertragingen en patronen

In this task, we will use the delay function to make the Robot blink in a pattern. The delay function is a delay function. By adding delay(5000); to the code, it will wait half a second before continuing.

Level 2.1 Vertraging

In this task, we will leave the LED off for a while before turning it on. Create the following setup and upload it to the robot:




```
void leaphyProgram() {
  delay(5000);
  analogWrite(3, 255);
  analogWrite(5, 0);
  analogWrite(6, 0);
}


void setup() {

  leaphyProgram();
}

void loop() {

}
```

Click the "Code" button at the top right.

Look at the script, and change the code so that instead of 5 seconds, it waits 10 seconds before the robot turns on its lamp. (ms explain)

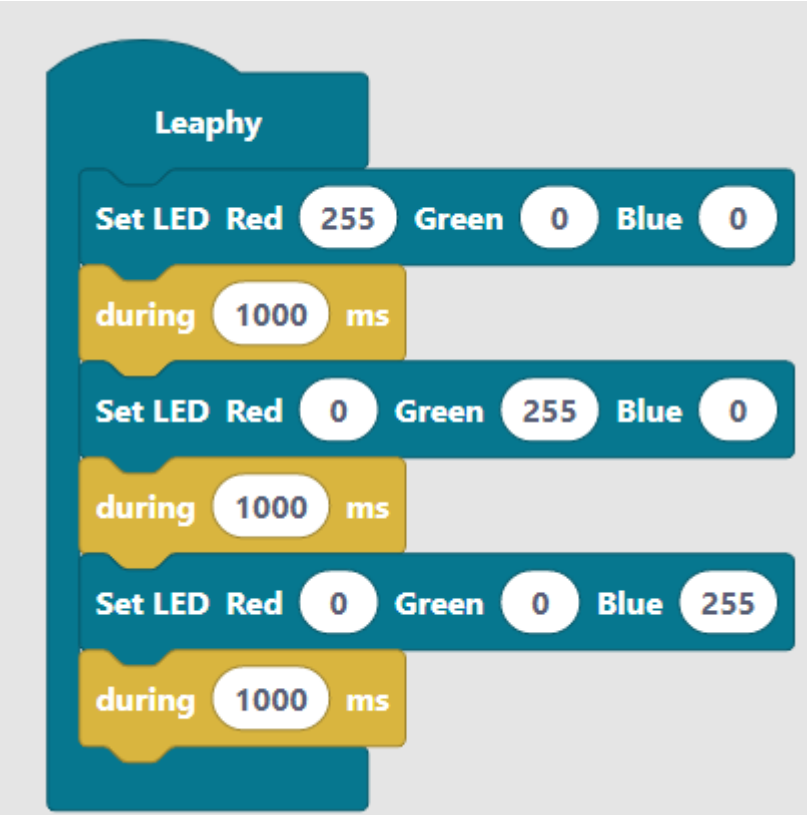
Level 2.2

Create a pattern using the delay + high and low

In this task we are going to try to make the LED blink in different colors. You can do this by setting the Digipin to true(1) and false(0) several times in a row. Then you have to enter the number of milliseconds that the LED should be on or off. Below you can see what it should look like:

In this task, we are going to try to make the "Neuslamp" (find on easyblocs the english word for that block) flash different colors. You can do this by modifying the "Neuslamp" block.

Then you need to enter the number of milliseconds that the LED should turn on or off. Below you can see what it should look like:



Upload this to the robot.

The LED now flashes every 1000 milliseconds. 1000 milliseconds is equivalent to 1 second. Now try changing the time in the code that the LED blinks. By changing the code, make sure that the LED blinks every 2 seconds (2000 milliseconds) instead of every 1 second.

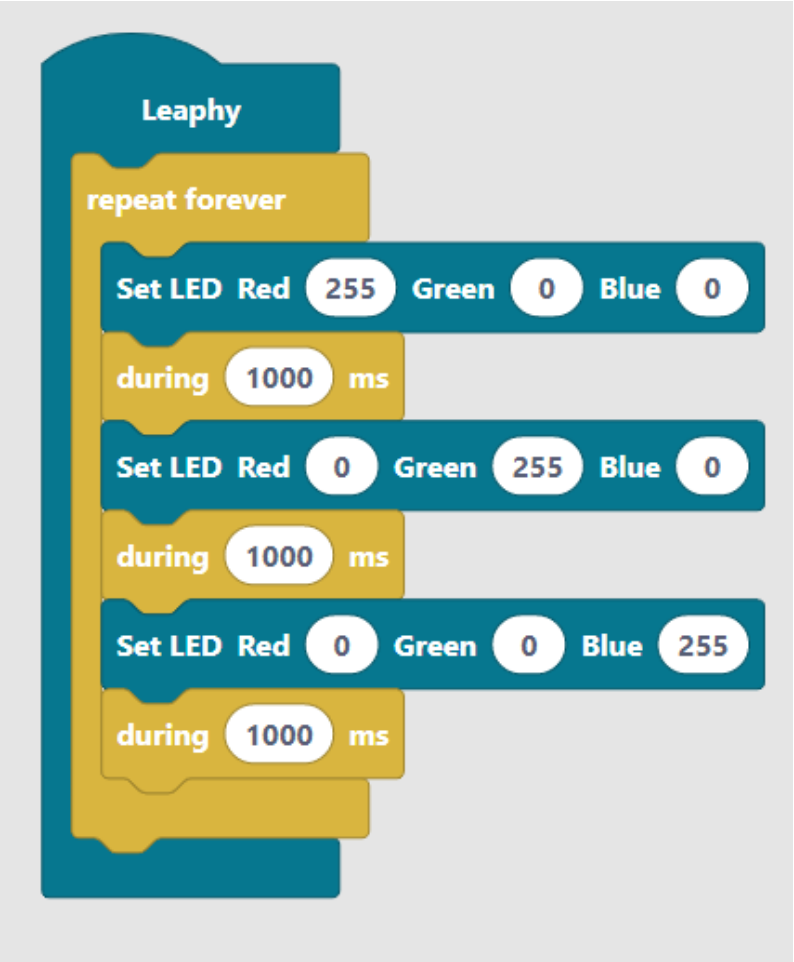
In what colors does the LED blink? Write this down.

Level 3 – While loop

In this task you will learn what a while loop is, by changing the pattern of the previous task.

Level 3.1

In this exercise we are going to change the pattern of the previous exercise so that we need less code. We do this through a While loop. For the While loop we need the "Repeat forever" block. We will use the “Repeat forever” block in Easybloqs. Below you can see what it should look like:



If you look in the code you will see “while(true)”.

--------- na 3 of 4 geen blokken meer gebruiken --------

Level 3.2

Go to the source code and change the pattern to green, red, blue.

Level 4 – Variables

You can think of a variable in programming as a box in which you can store things. Just like in real life you can put stuff in a box and take it out when you need it, so in programming you can store data in a variable and use it again later.

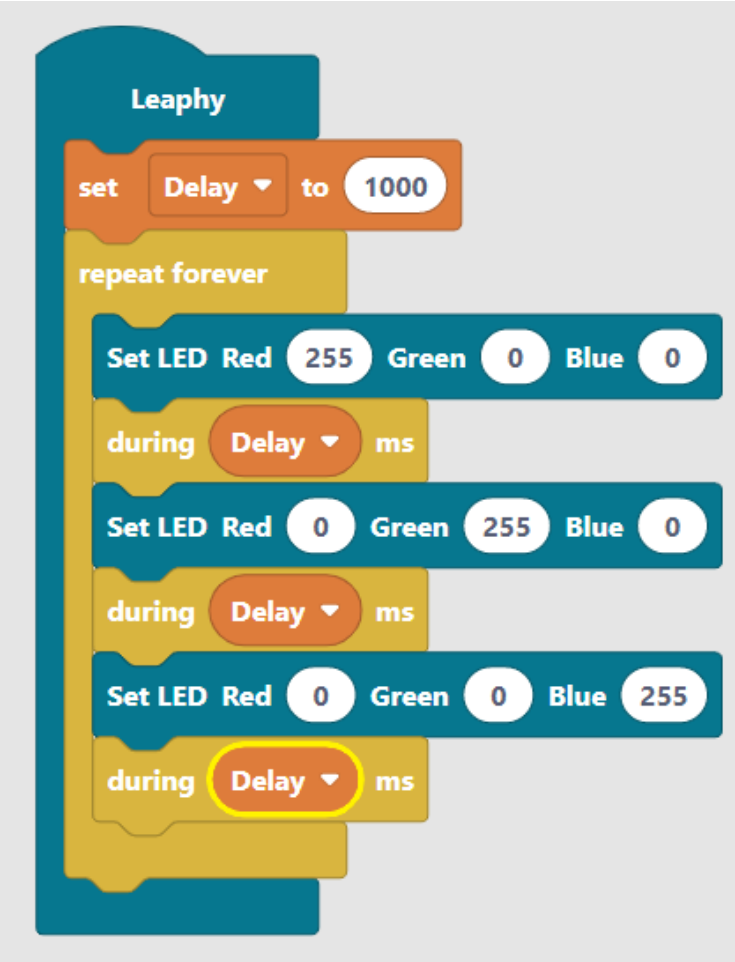
Level 4.1 - Loops

Use the setup from the previous command, and make the following adjustments.

Click on the “Variables” section on the right side of the screen.

Click on “Create Variable…”.

Name this variable “delay” and click OK.



Change the value of the delay using the "set ... to ..." block to a different value.

Level 4.2 - Loops

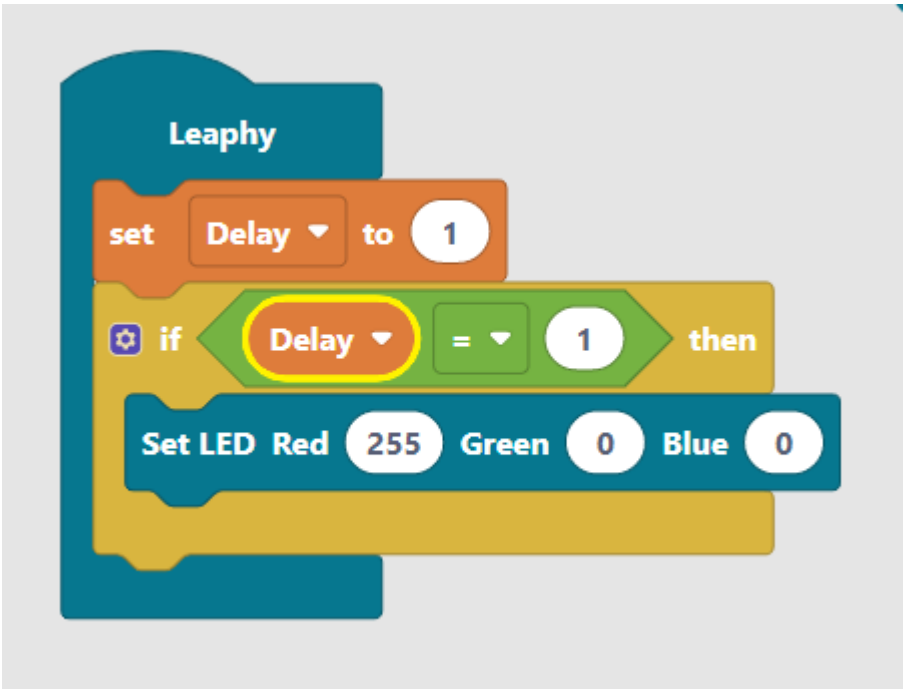
Add an extra variable. Change the delay of the pattern so that it waits 2 seconds once, and waits 4 seconds once.

Level 5 – If else

An if else statement in programming is like a special command for a robot to perform different actions depending on the situation.

Level 5.1 – If else

Make the following setup: The yellow goats can be found in the “Think Steps” section. Create a variable called “choice”



When you upload this, the robot's nose will turn red.

Study the code, and see if you can understand the logic.

```
double keuze;

void leaphyProgram() {
  keuze = 1;
  if (keuze == 1) {
    analogWrite(3, 255);
    analogWrite(5, 0);
    analogWrite(6, 0);
  }
}


void setup() {

  leaphyProgram();
}

void loop() {

}
```

Level 5.2 - Code

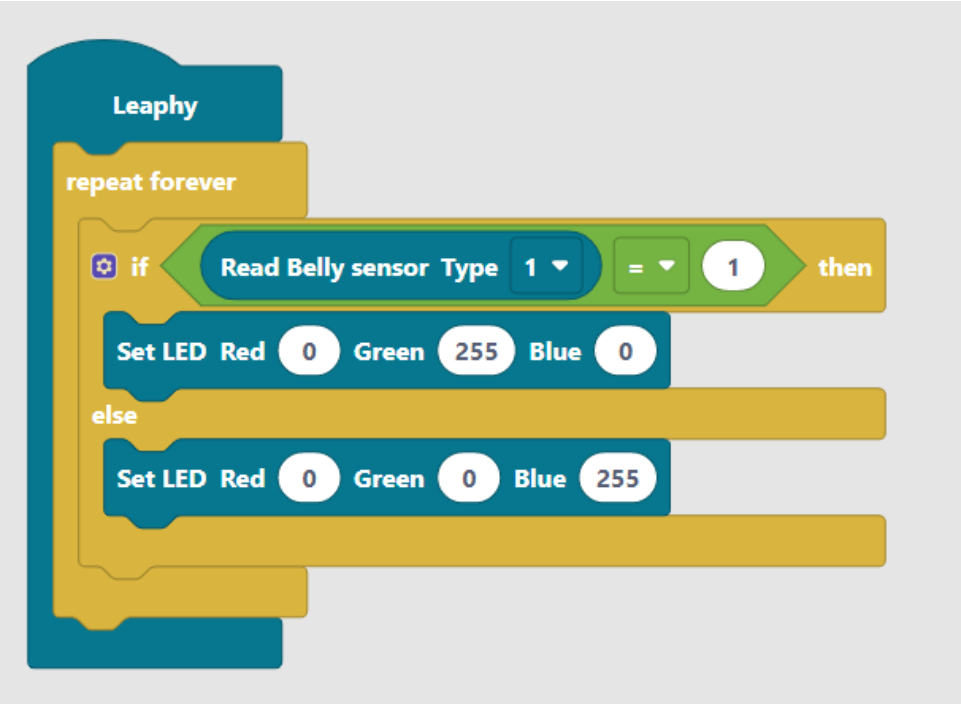
Go to the code section of EasyBloqs. Change the variable's value and see what happens or doesn't happen.

---------- Only code

Level 6 – Belly sensor

Create the following program and test if the belly sensor works:

(When “Lees buiksensor soort 1” doesn,t work, change it to “Lees buiksensor soort 2”)



Check the code, it should be something like this.

```
void leaphyProgram() {
  while (true) {
    if (digitalRead(10) == 1) {
      analogWrite(3, 0);
      analogWrite(5, 255);
      analogWrite(6, 0);
    } else {
      analogWrite(3, 0);
      analogWrite(5, 0);
      analogWrite(6, 255);
    }
  }
}
```

Compare the code with the blocks. Change the code in a way that the LED becomes red when you activate the buiksensor /belly sensor.