

IT mokytojų konferencija
„Informacinės technologijos: mokymo(si) kokybės gerinimas“
Vilnius, 2016-01-07



PROGRAMUOJAME VAIZDĄ: „Processing“ programavimo pradmenų modulyje

Antranas Budriūnas, Klaipėdos Vydūno gimnazija

Pranešimo tikslai

- Atkreipti dėmesį į programavimo mokymo(si) sunkumus, patrauklumo stoką
- Nurodyti „Processing“ pasirinkimo priežastis
- Pasiūlyti taikymo būdus ir priemones
- Apžvelgti praktinius rezultatus
- Pateikti išvadas ir padrašinti išmėginti

Sunkumai, mokantis C++

- C++ kalba neskirta programavimo pradmenims
- Pirmosios programos rašymui jau reikia daug žinių
- Programos išvestis tik tekstinė
- Negriežta klaidų kontrolė

C++ ir Java kodas

	main.cpp	Labas.java
	<pre>#include <iostream> int main() { std::cout << "Labas rytas!" << endl; return 0; }</pre>	<pre>public class Labas { public static void main(String[] args) { System.out.println("Labas rytas!"); } }</pre>

Grynoji *Java* kalba pradedančiajam programuotojui nėra paprastesnė už C++

C++ ir Processing kodas

	C++ kalba (CPP failas)	Processing kalba (PDE failas)
Išvestis į tekstinį terminalą	<pre>#include <iostream> int main() { std::cout << "i = " << i << "; << endl; return 0; }</pre>	<pre>println("i =", i + ";");</pre>
Animuota grafika	<p>Pagrindinio ugdymo programoje numatytais priemonėmis padaryti neįmanoma</p>	<pre>int x; void setup() { x = 0; } void draw() { point(x, 50); x++; }</pre>

Baziniai duomenų tipai

Skirtumai pabraukti

Duomenų tipai	C++	Processing (Java)
Loginis	bool	<u>boolean</u>
Baitas	byte	byte
Sveikasis	int	int
Rašmuo (simbolis)*	char	char
Ilgas sveikasis	long	long
Slankaus kablelio (trupmeninis)	float	float
Dvigubo tikslumo trupmeninis	double	double
<i>Rašmenų eilutė</i>	<i>string</i>	<u><i>String</i></u>

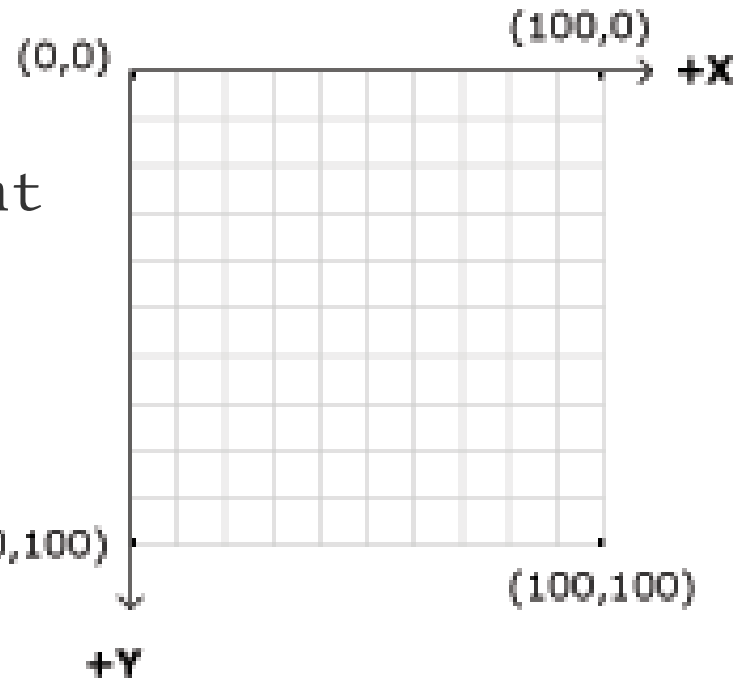
Kodėl „Processing“?



- Sukurta programavimo pradmenims ugdyti, plačiai naudojama
- Pagrindiniai programos kodo elementai užrašomi lygiai taip pat, kaip C++ kalboje
- Statiniam vaizdui (arba be vaizdo išvesties) nebūtinos pagrindinės funkcijos `setup()` ir `draw()`
- Yra žinynas, daug mokomosios medžiagos (anglų k.), pavyzdžių
- Naudojama dideliems (30.000 eilučių ir daugiau), profesionaliems medijų meno ir duomenų vizualizavimo projektams
- Pritaikyta daugeliui operacinių sistemų ir kompiuterio platformų (architektūrų)

Vaizdo programavimo ypatumai

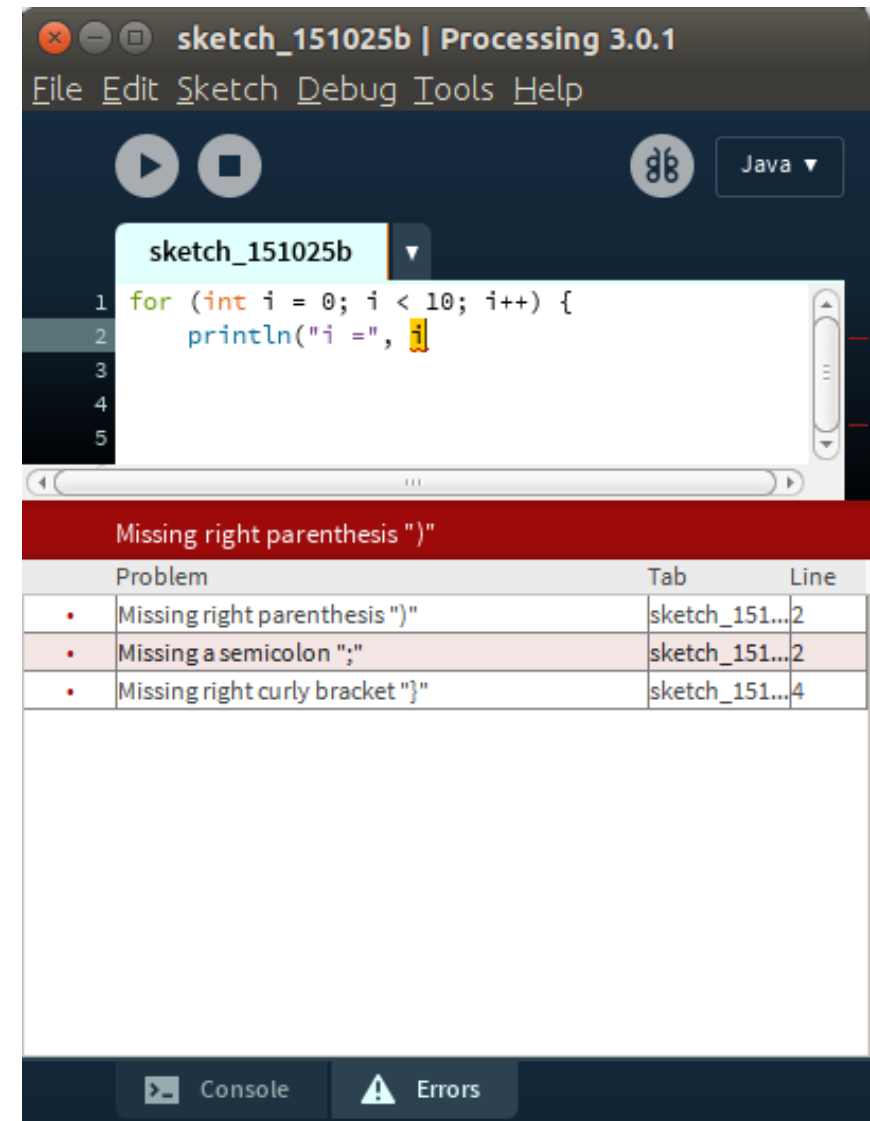
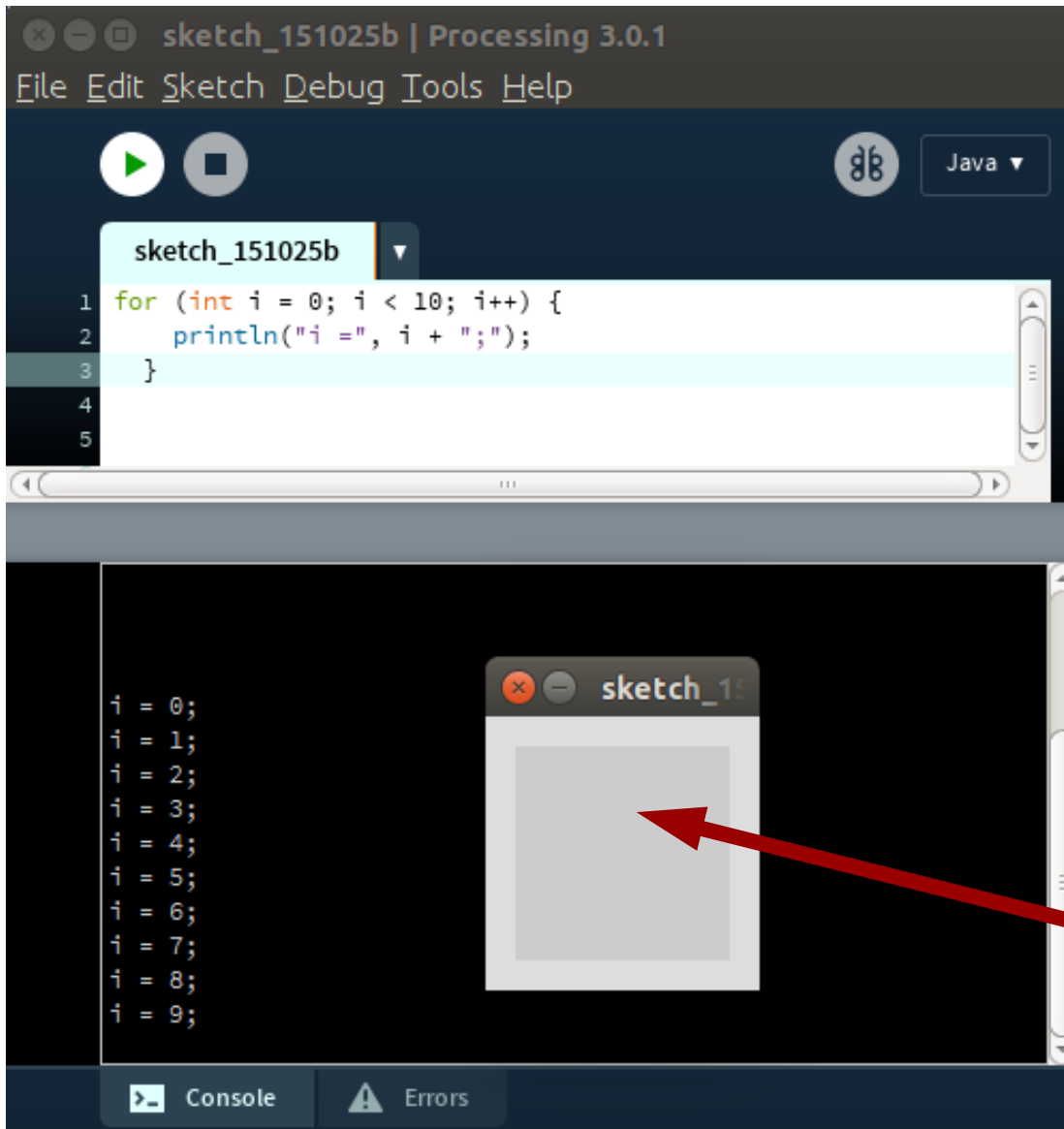
- Koordinačių sistema
 - sistemos kintamieji `width`, `height`
- Funkcijos
 - Pagrindinės (kaip C++ `main()` funkcija)
 - `setup()` // pradiniai veiksmai
 - `draw()` // kadrai (ciklas)
 - Grafinių priemonių
 - `point(x, y)` // taškas
 - `line(x1, y1, x2, y2)` // linija
 - `rect(x1, y1, x2, y2)` // stačiakampis
 - `stroke(r, g, b)` // pieštuko spalva
 - ...



Priemonės ir būdai

- PDE programavimo terpė
- Pavyzdžiai ir šablonai
- Mokymo(si) etapai:
 1. Užduotys be vaizdo išvesties
 2. Statinio vaizdo programavimas
 3. Animacijos, žaidimų, video programavimas
 4. Erdvinio (trimačio) vaizdo programavimas

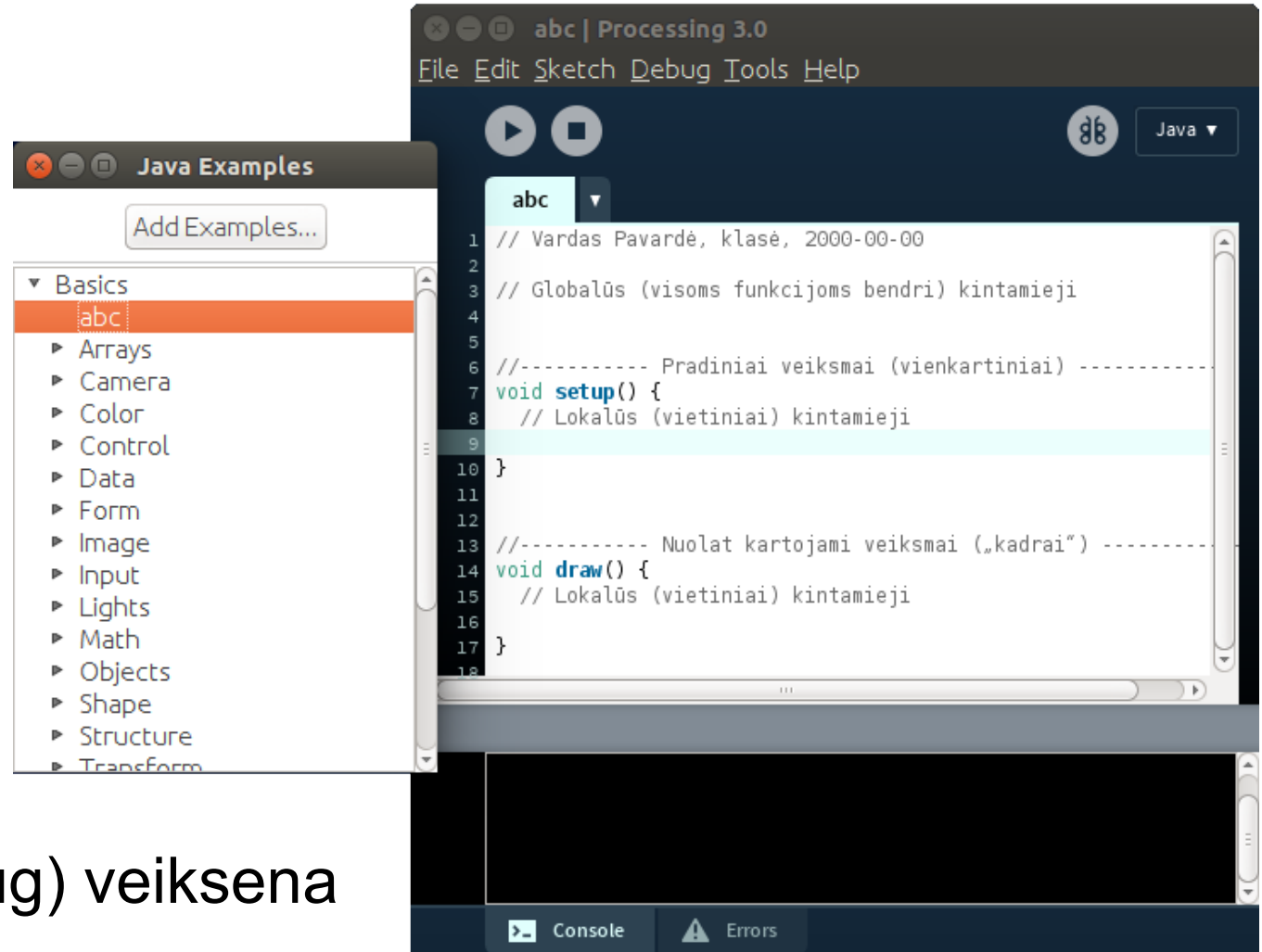
PDE Processing Development Environment



Kad grafinēs išvesties langas nebūtu rodomas, programos pradzioje irašyti:
`this.surface.setVisible(false);`

PDE galimybės

- Šablonai
- Rašant kodą aptinkamos ir rodomos klaidos
- Automatinis kodo formatavimas
- Derinimo (Tweak) veikseną
- Taisymo (Debug) veikseną



Šablonas „abc“, įrašytas į pavyzdžių katalogą

Užduočių pavyzdžiai (1)

Kintamieji, reikšmės priskyrimas, tiesiniai algoritmai

```
int vietaX = 10,  
    vietaY = 10,  
    dydis = 40;
```

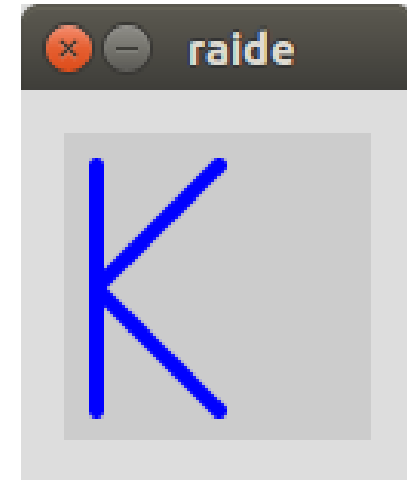
```
stroke(0, 0, 255); // pieštuko spalva
```

```
strokeWeight(5); // pieštuko storis
```

```
line (vietaX, vietaY, vietaX, vietaY + dydis * 2);
```

```
line (vietaX, vietaY + dydis, vietaX+dydis, vietaY);
```

```
line (vietaX, vietaY + dydis, vietaX+dydis, vietaY +  
dydis * 2);
```



Užduočių pavyzdžiai (2)

Veiksmų kartojimas

```
for(int i = 0; i < 10; i++) {  
  stroke(random(0, 255), random(0, 255), random(0, 255));  
  strokeWeight(i * 10);  
  point(i * 10, i * 10);  
}
```



Užduočių pavyzdžiai (3)

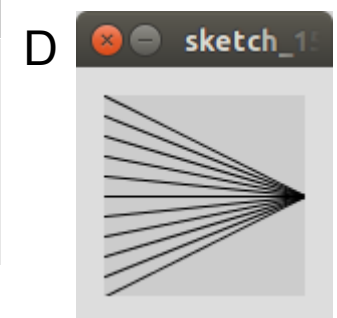
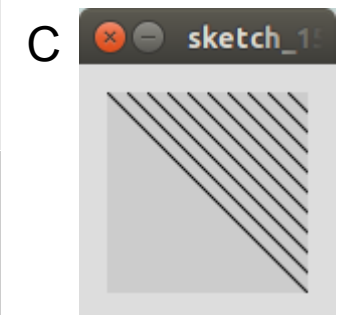
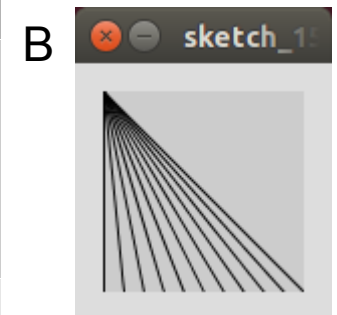
Veiksmų kartojimas ir šakojimas

```
int i = 0, // linijos ilgis
    t = 0; // linijos vieta
while (t < height) {
    line (0, t, i, t);
    if (t < height / 2) i = i + 10;
    else i = i - 10;
    t = t + 10;
}
```



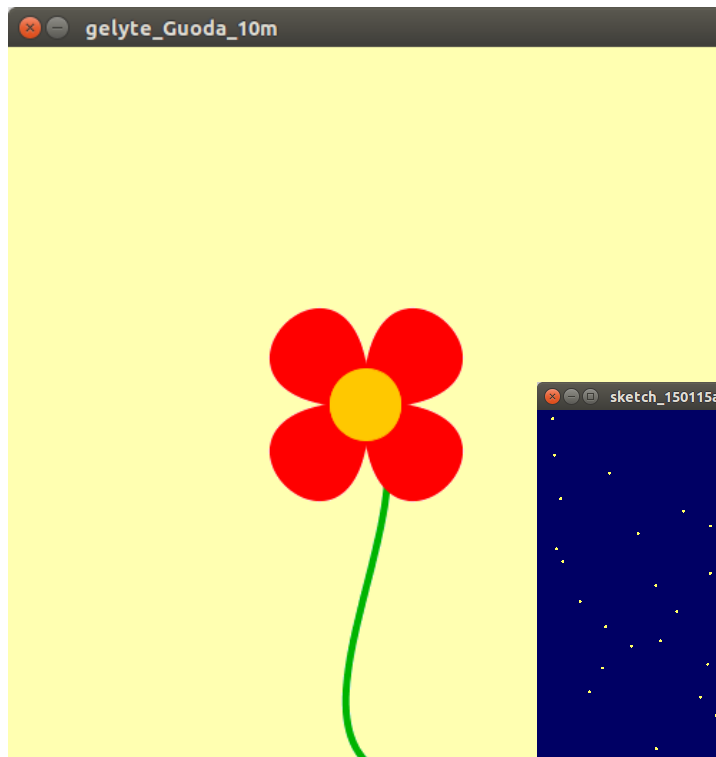
Testo pavyzdys

Eil. Nr.	Programos kodas	Kuris piešinys?
1.	<pre>int i = 0; while(i < width) { line(i, 0, width, height - i); i = i + 10;}</pre>	
2.	<pre>for (int i = height; i >= 0; i = i - 10) { line(0, i, width, height / 2); }</pre>	
3.	<pre>int i = 0; while(i < height) { line(0, i, width, i); i = i + 10;}</pre>	
4.	<pre>for (int i = 0; i <= height; i = i + 10) { line(0, 0, i, height); }</pre>	



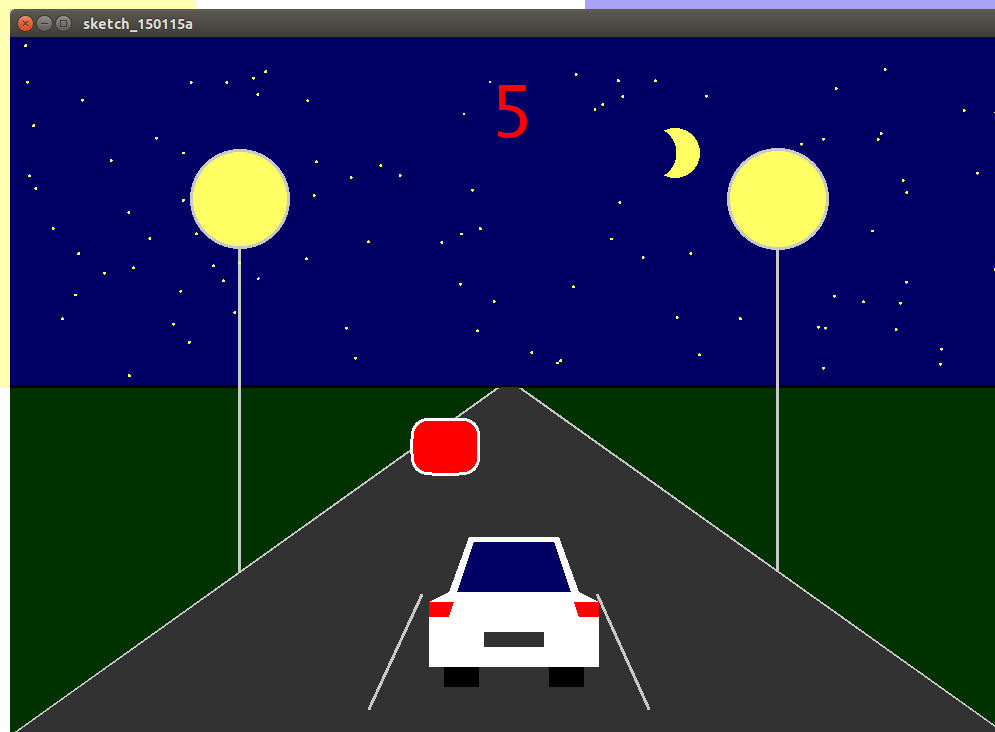
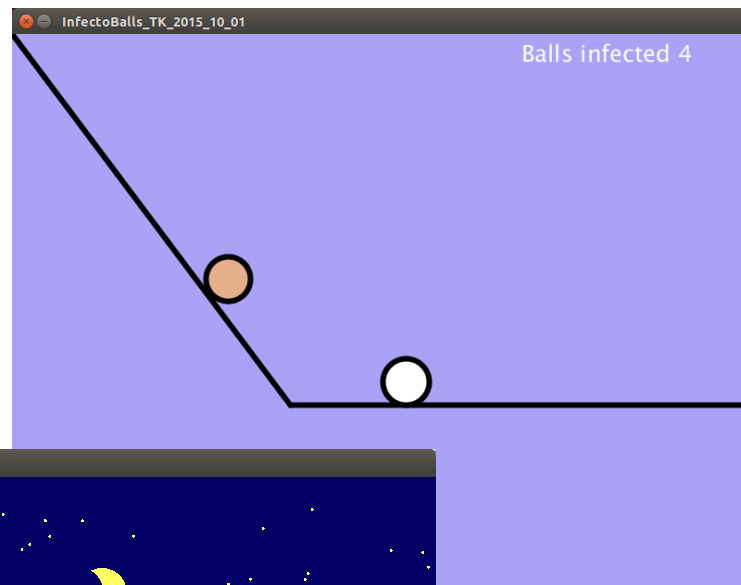
Mokinių darbai

Ekrano nuotraukos:



Augantis stiebelis ir išsiskleidžiantis gėlės žiedas.
Guoda Š. ir Tomas D.

Simuliacija „Užkrečiami rutuliai“.
Tomas K..



Žaidimas: važiuojant išvengti kuo daugiau kliūčių.
Tomas D. ir Jokūbas V.

Išvados

- „Processing“ privalumai
 - Greitai gaunamas patrauklus (vaizdinis) rezultatas
 - Patogi eksperimentams
 - Tinka integracijai bei STEAM ugdymui
 - Lengva diferencijuoti užduotis
 - Gabiausieji gali naudoti visas *Java* priemones
 - Programėles glaima įterpti į tinklalapius, naudojant *processing.js*
 - Lengvas perėjimas prie C++ kalbos
- Sunkumai pradedant:
 - orientacija dvimatėje teigiamų koordinačių sistemoje
 - reikia įsidėmėti funkcijų parametrų paskirtį ir jų eiliškumą

Smagių atradimų!

- www.processing.org
- www.openprocessing.org – projektų galerija

Bendraukime:

antanas.budriunas@vyduno.lt