МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Мегафакультет компьютерных технологий и управления

Факультет программной инженерии и компьютерной техники

Направление подготовки компьютерные технологии в дизайне

**ОТЧЁТ:**

По лабораторной работе №2.2

«Использование встроенных средств автоматизации в приложении»

По дисциплине «Информатика»

Студент группы:

Преподаватель:

Санкт-Петербург

2020

Оглавление

[Задание: 3](#_Toc54808774)

[Эскиз рисунка: 3](#_Toc54808775)

[Текст расширений: 3](#_Toc54808776)

[Итоговое изображение: 4](#_Toc54808777)

Задание 1

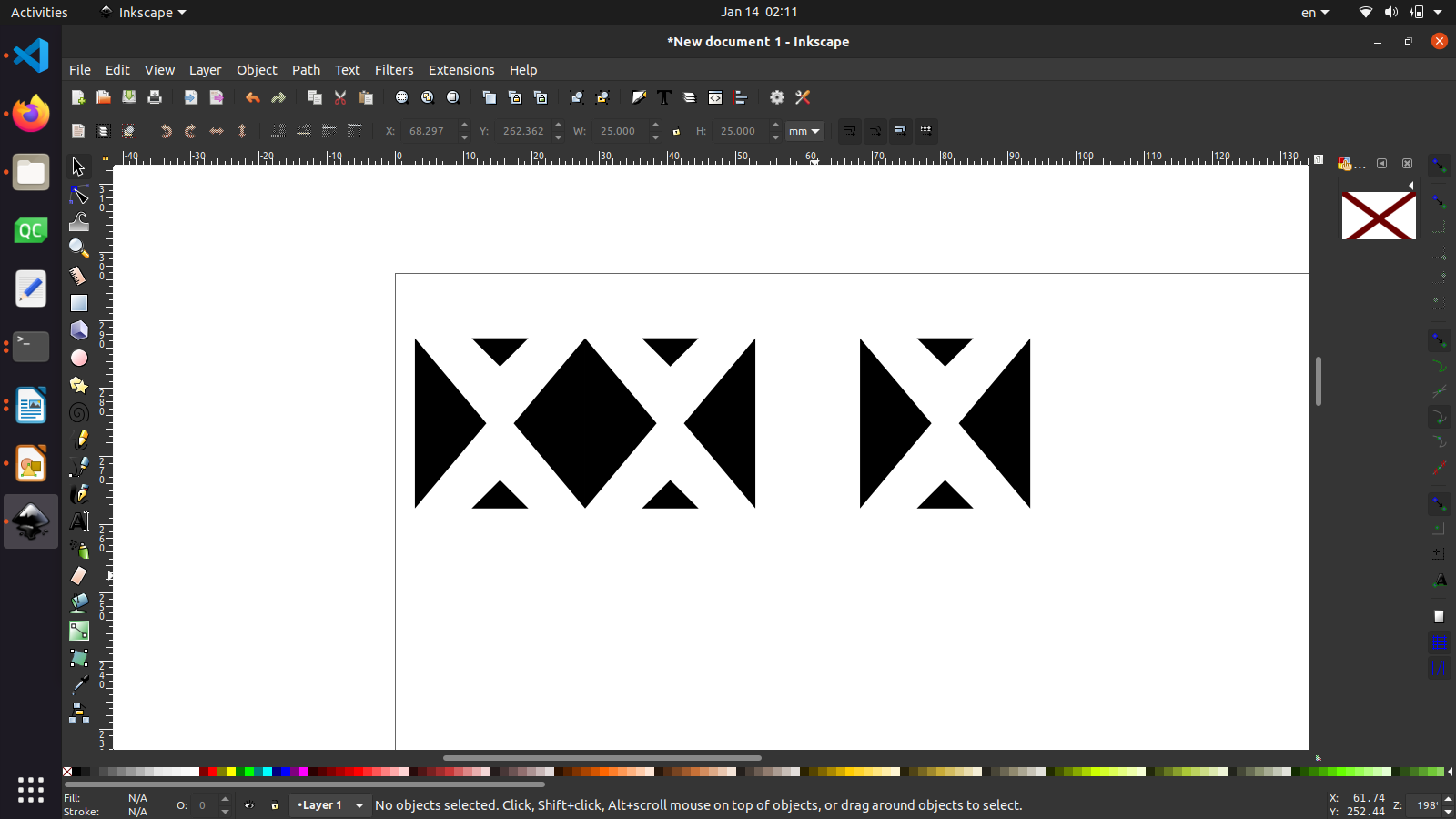
Текст задания:

Придумайте линейный (горизонтальный) орнамент и разработайте расширение Inkscape для его рисования.

Расширение должно запрашивать у пользователя:

* Количество раппортов
* Ширину раппорта
* Высоту раппорта

Эскиз орнамента:



Текст расширения (.inx):

<?xml version="1.0" encoding="UTF-8"?>

<inkscape-extension xmlns="http://www.inkscape.org/namespace/inkscape/extension">

<\_name>Raport\_new</\_name>

<id>org.ekips.filter.Raport\_new</id>

<dependency type="executable" location="extensions">raport.py</dependency>

<dependency type="executable" location="extensions">inkex.py</dependency>

<param name="count" type="int" min="1" max="10000" \_gui-text="Len rapport">24</param>

<param name="height" type="float" min="0.0" max="1000.0" \_gui-text="height rapport">20.0</param>

<param name="width" type="float" min="10.0" max="30.0" \_gui-text="width rapport">20.0</param>

<effect>

<object-type>all</object-type>

<effects-menu>

<submenu \_name="Render"/>

</effects-menu>

</effect>

<script>

<command reldir="extensions" interpreter="python">raport.py</command>

</script>

</inkscape-extension>

Текст расширения (.py):

#!/usr/bin/env python3

# coding=utf-8

import inkex

from simpletransform import computePointInNode

import simplestyle

class Raport(inkex.Effect):

def \_\_init\_\_(self):

inkex.Effect.\_\_init\_\_(self)

self.OptionParser.add\_option("-c", "--count",

action="store", type="int",

dest="count", default=24,

help="Count raport")

self.OptionParser.add\_option("-w", "--width",

action="store", type="float",

dest="width", default=20.0,

help="width")

self.OptionParser.add\_option("--height",

action="store", type="float",

dest="height", default=20.0,

help="height")

def ornament(self, coords\_left, coords\_right, coords\_top, coords\_bottom):

rapport\_str = ''

triangle\_left = 'M{},{} L{},{} L{},{} z '.format(

coords\_left['x1'],

coords\_left['y1'],

coords\_left['x2'],

coords\_left['y2'],

coords\_left['x3'],

coords\_left['y3']

)

triangle\_right = 'M{},{} L{},{} L{},{} z '.format(

coords\_right['x1'],

coords\_right['y1'],

coords\_right['x2'],

coords\_right['y2'],

coords\_right['x3'],

coords\_right['y3']

)

triangle\_top = 'M{},{} L{},{} L{},{} z '.format(

coords\_top['x1'],

coords\_top['y1'],

coords\_top['x2'],

coords\_top['y2'],

coords\_top['x3'],

coords\_top['y3']

)

triangle\_bottom = 'M{},{} L{},{} L{},{} z '.format(

coords\_bottom['x1'],

coords\_bottom['y1'],

coords\_bottom['x2'],

coords\_bottom['y2'],

coords\_bottom['x3'],

coords\_bottom['y3']

)

return rapport\_str + triangle\_left + triangle\_right + triangle\_top + triangle\_bottom

def effect(self):

count = self.options.count

width = self.options.width

height = self.options.height

parent = self.current\_layer

rapport\_str = ''

for i in range(1, count+1):

coords\_left = {

'x1' : width\*i-width,

'y1' : 0,

'x2' : width\*i-width/2 - 2,

'y2' : height / 2,

'x3' : width\*i-width,

'y3' : height,

}

coords\_right = {

'x1' : width\*i,

'y1' : 0,

'x2' : width\*i-width/2 + 2,

'y2' : height / 2,

'x3' : width\*i,

'y3' : height,

}

if i == 1:

x1 = width/3

x2 = width/2

x3 = width - width/3

else:

x1 = (width\*i - width) + width/3

x2 = width\*i - width/2

x3 = width\*i - width/3

coords\_top = {

'x1' : x1,

'y1' : 0,

'x2' : x2,

'y2' : height / 6,

'x3' : x3,

'y3' : 0,

}

coords\_bottom = {

'x1' : x1,

'y1' : height,

'x2' : x2,

'y2' : height - height / 6,

'x3' : x3,

'y3' : height,

}

rapport\_str += self.ornament(

coords\_left,

coords\_right,

coords\_top,

coords\_bottom

)

style = {'fill': 'black'}

path = {

'style':simplestyle.formatStyle(style),

'd': rapport\_str

}

inkex.etree.SubElement(parent, inkex.addNS('path','svg'), path )

if \_\_name\_\_ == '\_\_main\_\_':

Raport().affect()

Скриншот экрана с введенными данными раппорта и его отрисовкой

