Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего профильного образования «Национальный исследовательский университет ИТМО»

Мегафакультет компьютерных технологий и управления

Факультет программной инженерии и компьютерной техники

Образовательная программа: «Компьютерные технологии в дизайне»

«Использование встроенных средств автоматизации в приложении»

Отчёт

Студент: Кешишев Максим Юрьевич

Группа: P3166

Преподаватель: Климов Игорь Викторович

Санкт-Петербург, 2023

Оглавление

[Текст задания 3](#_Toc151139187)

[Эскиз орнамента с координатами 3](#_Toc151139188)

[Текст расширения 3](#_Toc151139189)

[Файл inx 3](#_Toc151139190)

[Файл py 4](#_Toc151139191)

[Снимки результатов 6](#_Toc151139192)

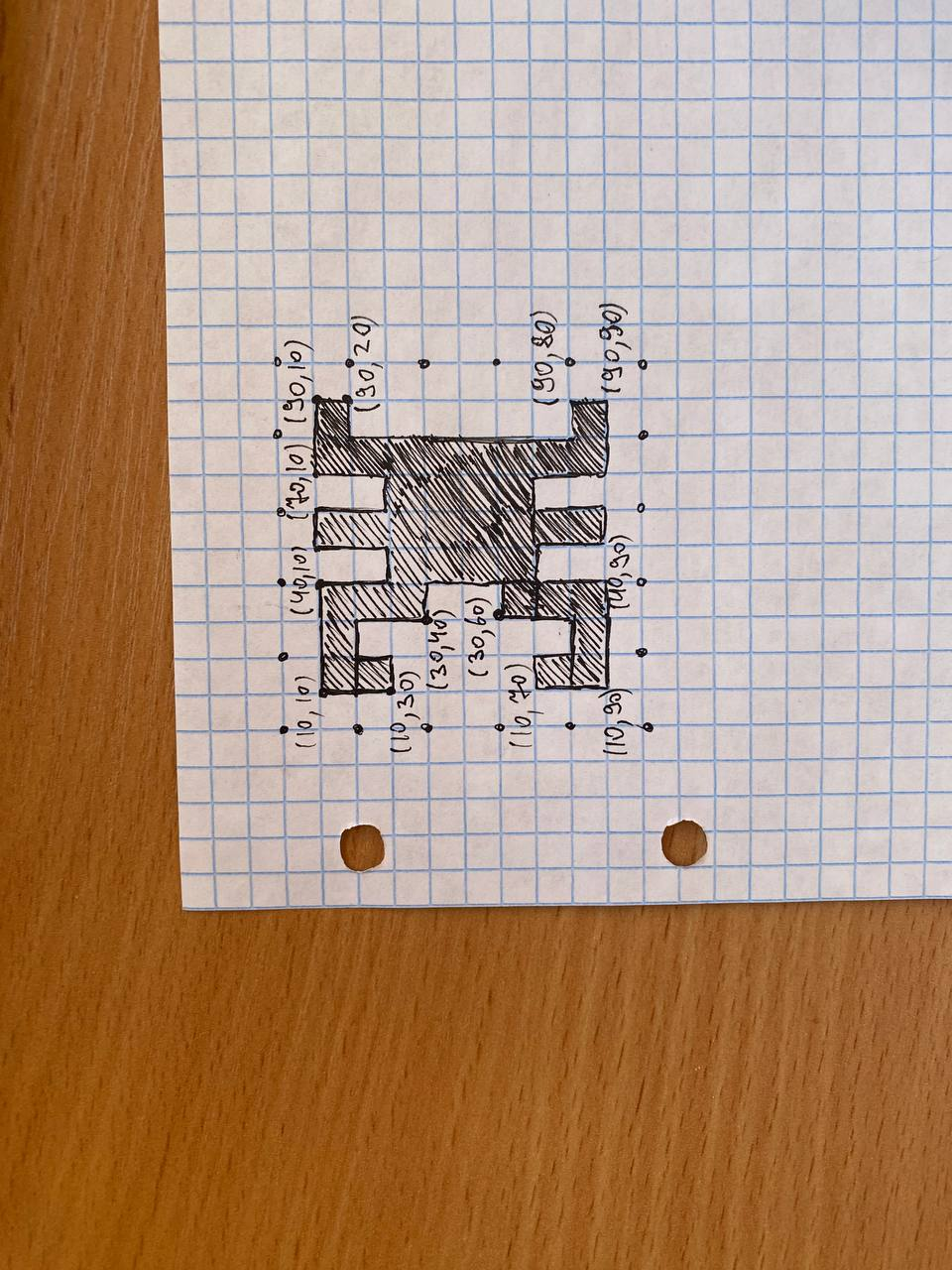
Текст задания

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

Расширение должно запрашивать у пользователя:

1. Количество раппортов
2. Высоту раппорта
3. Ширину раппорта

Эскиз орнамента с координатами



Текст расширения

Файл inx

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

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

<name>Ornament</name>

<id>ornament\_</id>

<param name="number" type="int" min="1" max="1000" gui-text="Number of rapports:">10</param>

<param name="height" type="float" min="0.01" max="10000" gui-text="Height:">100.0</param>

<param name="weight" type="float" min="0.01" max="10000" gui-text="Weight:">100.0</param>

<effect>

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

<effects-menu>

<submenu name="Render"/>

</effects-menu>

</effect>

<script>

<command location="inx" interpreter="python">\_ornament\_.py</command>

</script>

</inkscape-extension>

Файл py

#!/usr/bin/env python

# coding=utf-8

import sys

import inkex

from lxml import etree

def draw\_SVG\_ornament(ct, hg, wg, parent):

xk = wg / 100

yk = hg / 100

style = {'fill': '#C20030'}

for i in range (0, ct):

elem = parent.add(inkex.PathElement())

elem.update(\*\*{

'style': style,

'inkscape:label': 'Ornament',

'd': 'M ' + str(0\*xk + wg\*i) + ',' + str(0\*yk) +

'M ' + str(100\*xk + wg\*i) + ',' + str(100\*yk) +

'M ' + str(10\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(30\*yk) +

' L ' + str(50\*xk + wg\*i) + ',' + str(30\*yk) +

' L ' + str(50\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(60\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(60\*xk + wg\*i) + ',' + str(30\*yk) +

' L ' + str(70\*xk + wg\*i) + ',' + str(30\*yk) +

' L ' + str(70\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(90\*xk + wg\*i) + ',' + str(10\*yk) +

' L ' + str(90\*xk + wg\*i) + ',' + str(20\*yk) +

' L ' + str(80\*xk + wg\*i) + ',' + str(20\*yk) +

' L ' + str(80\*xk + wg\*i) + ',' + str(80\*yk) +

' L ' + str(90\*xk + wg\*i) + ',' + str(80\*yk) +

' L ' + str(90\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(70\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(70\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(60\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(60\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(50\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(50\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(10\*xk + wg\*i) + ',' + str(90\*yk) +

' L ' + str(10\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(20\*xk + wg\*i) + ',' + str(70\*yk) +

' L ' + str(20\*xk + wg\*i) + ',' + str(80\*yk) +

' L ' + str(30\*xk + wg\*i) + ',' + str(80\*yk) +

' L ' + str(30\*xk + wg\*i) + ',' + str(60\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(60\*yk) +

' L ' + str(40\*xk + wg\*i) + ',' + str(40\*yk) +

' L ' + str(30\*xk + wg\*i) + ',' + str(40\*yk) +

' L ' + str(30\*xk + wg\*i) + ',' + str(20\*yk) +

' L ' + str(20\*xk + wg\*i) + ',' + str(20\*yk) +

' L ' + str(20\*xk + wg\*i) + ',' + str(30\*yk) +

' L ' + str(10\*xk + wg\*i) + ',' + str(30\*yk) +

' z'

})

return elem

class ornament(inkex.EffectExtension):

def add\_arguments(self, pars):

pars.add\_argument("--number", type = int, default = 10, help="Number of rapports")

pars.add\_argument("--height", type = float, default = 100.0, help="Height")

pars.add\_argument("--weight", type = float, default = 100.0, help="Weight")

def effect(self):

centre = self.svg.namedview.center

grp\_transform = 'translate' + str(centre)

grp\_name = 'Group Name'

grp\_attribs = {inkex.addNS('label', 'inkscape'): grp\_name, 'transform': grp\_transform}

grp = self.svg.get\_current\_layer()

ct = self.options.number

hg = self.svg.unittouu(str(self.options.height) + 'px')

wg = self.svg.unittouu(str(self.options.weight) + 'px')

draw\_SVG\_ornament(ct,hg,wg,grp)

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

ornament().run()

Снимки результатов

