Міністерство освіти і науки УКРАЇНИ

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Кафедра програмного забезпечення комп’ютерних систем

**Модульна контрольна робота**

з дисципліни

**«Розроблення мобільних застосунків»**

*Варіант №10*

***Виконав:*** студент 3 курсу

групи КП93

спеціальність 121 – Інженерія

програмного забезпечення

Інюшев Артем Владиславович

Київ – 2020

**Завдання 1**

За допомого Canvas або її аналогів реалізувати рух круга, який має радіус R0 по кривій за варіантом.

Варіант No10. Фігури Ліссажу

https://uk.wikipedia.org/wiki/%D0%A4%D1%96%D0%B3%D1%83%D1%80%D0

%B8\_%D0%9B%D1%96%D1%81%D1%81%D0%B0%D0%B6%D1%83

Формула фігури Ліссажу:

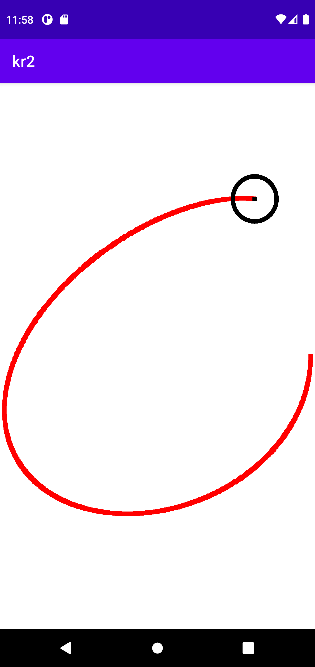
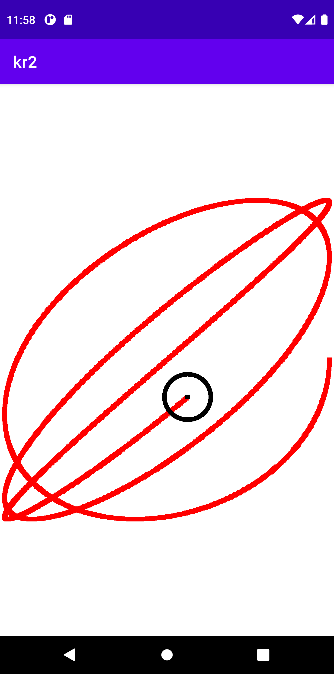
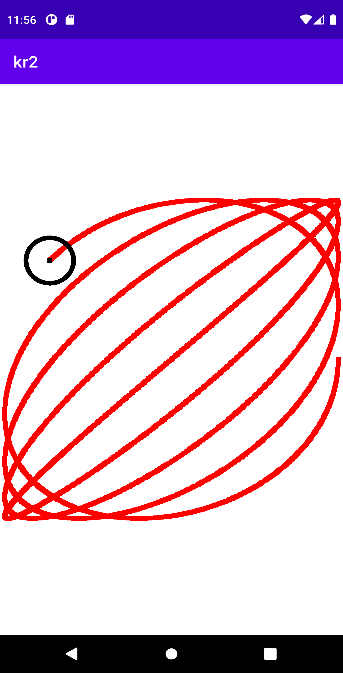
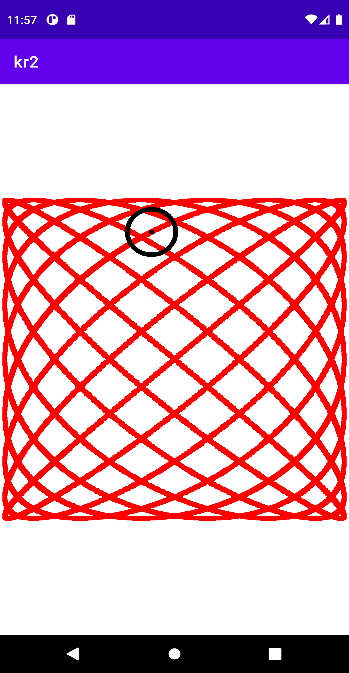
{ x(t) = A\*sin(at + delta)

{ y(t) = B\*sin(bt)

Для цієї формули я обрав такі параметри:

A = B = 700; a = 9; b = 8; delta = pi / 2;

**Зображення роботи програми:**

**   **

**Програмний код:**

|  |
| --- |
| MyView.java |
| package com.example.kr2;  import android.content.Context;  import android.graphics.Canvas;  import android.graphics.Color;  import android.graphics.Paint;  import android.view.View;  import java.util.ArrayList;  class Point{  public float X;  public float Y;  public Point(float x, float y){  X = x;  Y = y;  }  }  public class MyView extends View {  private float a = 9;  private float b = 8;  private double delta = Math.PI / 2;  private float t = 0;  private ArrayList<Point> \_points = new ArrayList<Point>();  public MyView(Context context) {  super(context);  }  @Override  protected void onDraw(Canvas canvas) {  super.onDraw(canvas);  Paint paintPoint = new Paint();  paintPoint.setColor(Color.RED);  paintPoint.setStyle(Paint.Style.STROKE);  paintPoint.setStrokeWidth(20);  for(Point point : \_points){  canvas.drawPoint(point.X, point.Y, paintPoint);  }  float x = (float) (700 \* Math.sin(a \* t + delta) + canvas.getWidth() / 2);  float y = (float) (700 \* Math.sin(b \* t) + canvas.getHeight() / 2);  Point newPoint = new Point(x, y);  \_points.add(newPoint)  t += 0.001;  Paint paintCircle = new Paint();  paintCircle.setColor(Color.BLACK);  paintCircle.setStyle(Paint.Style.STROKE);  paintCircle.setStrokeWidth(20);  canvas.drawPoint(x, y, paintCircle);  canvas.drawCircle(x, y, 100, paintCircle);  invalidate();  }  } |

**Завдання 2**

Створити мобільний застосунок, який зливає дві картинки з

коефіцієнтом 0,5. Інтерфейс додатку повинен містити три зображення: два

вхідних, третє – вихідне.

**Зображення роботи програми:**



**Програмний код:**

|  |
| --- |
| MyView.java |
| package com.example.kr2;  import androidx.appcompat.app.AppCompatActivity;  import android.graphics.Bitmap;  import android.graphics.BitmapFactory;  import android.graphics.Color;  import android.os.Bundle;  import android.widget.EditText;  import android.widget.ImageView;  public class MainActivity extends AppCompatActivity {  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  //setContentView(new MyView(this)); // task1  setContentView(R.layout.activity\_main); // task2  startMerge(); // task2  }  private void startMerge() {  double ratio = 0.5;  mergeImages(  BitmapFactory.decodeResource(getResources(), R.drawable.glass),  BitmapFactory.decodeResource(getResources(), R.drawable.car),  ratio  );  }  private void mergeImages(Bitmap source1, Bitmap source2, double ratio) {  Bitmap result = Bitmap.createBitmap(source1.getWidth(),  source1.getHeight(), source1.getConfig());  for (int x = 0; x < source1.getWidth(); x++) {  for (int y = 0; y < source1.getHeight(); y++) {  int color1 = source1.getPixel(x, y);  int color2 = source2.getPixel(x, y);  result.setPixel(x, y, Color.argb(  (int) (Color.alpha((color1)) \* ratio +  Color.alpha((color2)) \* (1 - ratio)),  (int) (Color.red((color1)) \* ratio + Color.red((color2))  \* (1 - ratio)),  (int) (Color.green((color1)) \* ratio +  Color.green((color2)) \* (1 - ratio)),  (int) (Color.blue((color1)) \* ratio +  Color.blue((color2)) \* (1 - ratio))  ));  }  }  ImageView mergedView = (ImageView) findViewById(R.id.mergedImage);  mergedView.setImageBitmap(result);  }  } |