

-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 №8

Виконав: студент 3 курсу
групи КП-83
спеціальність 121 – Інженерія
програмного забезпечення
Коваль Андрій Олександрович

Київ – 2020

ЗМІСТ

1. Завдання до контрольної роботи.....	2
2. Програмна реалізація задачі.....	3

Завдання до контрольної роботи

1. За допомогою *Canvas* або її аналогів реалізувати рух круга, який має радіус R_0 по Астроїді.
2. Створити мобільний застосунок, який зливає дві картинки з коефіцієнтом 0,5. Інтерфейс додатку повинен містити три зображення: два вхідних, третє – вихідне.

Програмна реалізація задачі

Перехід до завдань відбувається у нових Activity

MainActivity.java

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onTask1(View v) {
        Intent in = new Intent();
        startCertainActivity(in, AstroidaActivity.class);
    }

    public void onTask2(View v) {
        Intent in = new Intent();
        startCertainActivity(in, Task2Activity.class);
    }

    private void startCertainActivity(Intent in, Class<?> cls) {
        in.setClass(this, cls);
        startActivity(in);
    }
}
```

Завдання 1

AstroidaActivity.java

```
public class AstroidaActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new AstroidaView(this));
    }

    public class AstroidaView extends View {

        Paint paint;
        int R = 500;
        int smallR = 20;
        int smallCircleColor = Color.BLACK;
        float t = 0;
        float dt = 0.01f;
        float cx, cy;
        int strokeWidth = 10;
        int astroidaColor = Color.RED;
        ArrayList<Point> points;

        int pointIndexForCircle = 0;

        boolean isInitied = false;
        boolean astroidaIsDrawen = false;

        public AstroidaView(Context context) {
            super(context);
        }

        private void init(Canvas canvas) {
            paint = new Paint();
            this.points = new ArrayList<Point>();
            this.isInitied = true;
            this.cx = canvas.getWidth() / 2.0f;
            this.cy = canvas.getHeight() / 2.0f;
            paint.setColor(astroidaColor);
            paint.setStrokeWidth(strokeWidth);
            // calculate points
            while (t < Math.PI * 2) {
                double x = R * Math.pow(Math.cos(t), 3);
                double y = R * Math.pow(Math.sin(t), 3);
                t += dt;
                Point point = new Point((int)x, (int)y);
                points.add(point);
            }
        }

        @Override
        protected void onDraw(Canvas canvas) {
            if (!this.isInitied) {
                this.init(canvas);
            }
            drawAstroida(canvas);
            drawPoint(canvas);
        }
    }
}
```

```

        pointIndexForCircle++;
        if (pointIndexForCircle >= points.size()) {
            pointIndexForCircle = 0;
        }
        invalidate();
    }

    private void drawAstroidea(Canvas canvas) {
        if (points.isEmpty()) {
            return;
        }

        for (Point p : points) {
            canvas.drawPoint(p.x + cx, p.y + cy, paint);
        }
    }

    private void drawPoint(Canvas canvas) {
        Point p = points.get(pointIndexForCircle);
        paint.setColor(smallCircleColor);
        canvas.drawCircle(p.x + cx, p.y + cy, smallR, paint);
        paint.setColor(astroideaColor);
    }
}

```

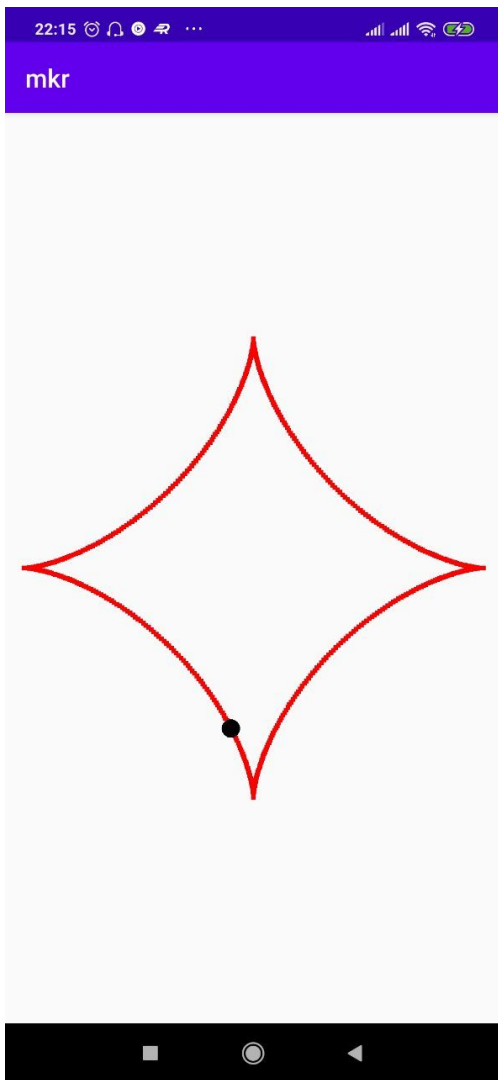


Рис 1-1. Рух точки по астроїді

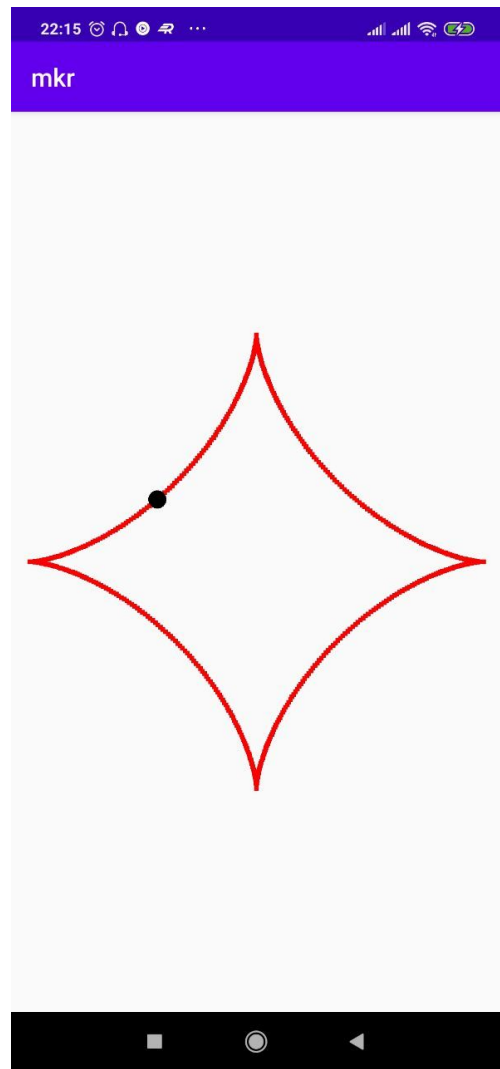


Рис 1-2. Рух точки по астроїді

Завдання 2

Завдання було доповнено змінний коефіцієнтом прозорості

Task2Activity.java

```
public class Task2Activity extends AppCompatActivity {

    int img1ResId = R.mipmap.mac_foreground;
    int img2ResId = R.mipmap.student_foreground;

    double alpha = 0.5;

    int width, height;

    Bitmap bm1, bm2;
    int[] pixels1, pixels2;

    ImageView imageView1, imageView2, result;
    TextView aplhaTextView;
    SeekBar seekBar;

    @RequiresApi(api = Build.VERSION_CODES.O)
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_task2);
        init();
        setupEventListeners();
        combineImages();
    }

    private void init() {
        this.imageView1 = findViewById(R.id.imageView1);
        this.imageView2 = findViewById(R.id.imageView2);
        this.result = findViewById(R.id.resultView);
        this.aplhaTextView = findViewById(R.id.aplhaTextView);

        bm1 = BitmapFactory.decodeResource(getResources(), img1ResId);
        bm2 = BitmapFactory.decodeResource(getResources(), img2ResId);
        this.imageView1.setImageBitmap(bm1);
        this.imageView2.setImageBitmap(bm2);

        width = bm1.getWidth();
        height = bm1.getHeight();

        pixels1 = new int[width * height];
        pixels2 = new int[pixels1.length];

        bm1.getPixels(pixels1, 0, width, 0, 0, width, height);
        bm2.getPixels(pixels2, 0, width, 0, 0, width, height);

        this.seekBar = findViewById(R.id.seekBar);
        this.seekBar.setProgress((int)(this.alpha * 100));
    }

    @RequiresApi(api = Build.VERSION_CODES.O)
    private void combineImages() {
        this.aplhaTextView.setText(Double.toString(alpha));
        int[] dst = new int[pixels1.length];
```



```

        for (int i = 0; i < pixels1.length; i++) {
            Color color1 = Color.valueOf(pixels1[i]);
            Color color2 = Color.valueOf(pixels2[i]);
            int r = (int)(color1.red() * alpha * 255) + (int)((1f - alpha) *
color2.red() * 255);
            int g = (int)(color1.green() * alpha * 255) + (int)((1f - alpha) *
color2.green() * 255);
            int b = (int)(color1.blue() * alpha * 255) + (int)((1f - alpha) *
color2.blue() * 255);
            dst[i] = Color.argb(255, r, g, b);
        }

        Bitmap dstBitmap = Bitmap.createBitmap(width, height,
Bitmap.Config.ARGB_8888);
        dstBitmap.setPixels(dst, 0, width, 0, 0, width, height);
        result.setImageBitmap(dstBitmap);
    }

    private void setupEventListeners() {
        this.seekBar.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            @RequiresApi(api = Build.VERSION_CODES.O)
            @Override
            public void onProgressChanged(SearchBar seekBar, int progress, boolean
fromUser) {
                alpha = progress / 100.0;
                combineImages();
            }

            @Override
            public void onStartTrackingTouch(SearchBar seekBar) {
            }

            @Override
            public void onStopTrackingTouch(SearchBar seekBar) {
            }
        });
    }
}

```

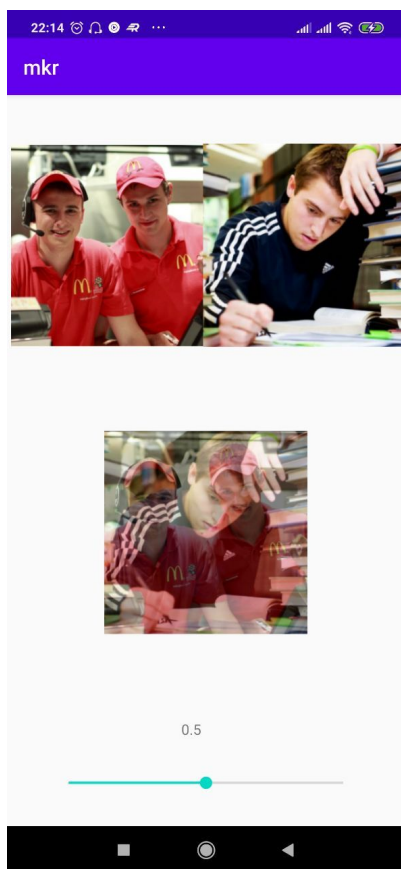


рис 2-1, поєднання
зображень при лямбда
0.5

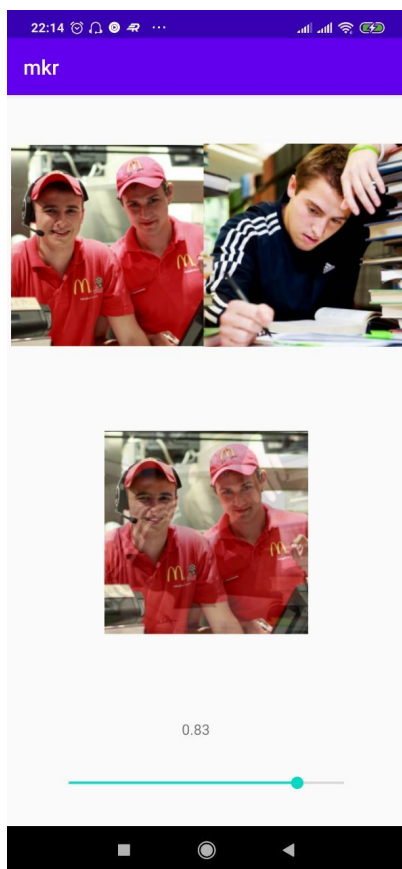


рис 2-2, поєднання
зображень при лямбда
0.83

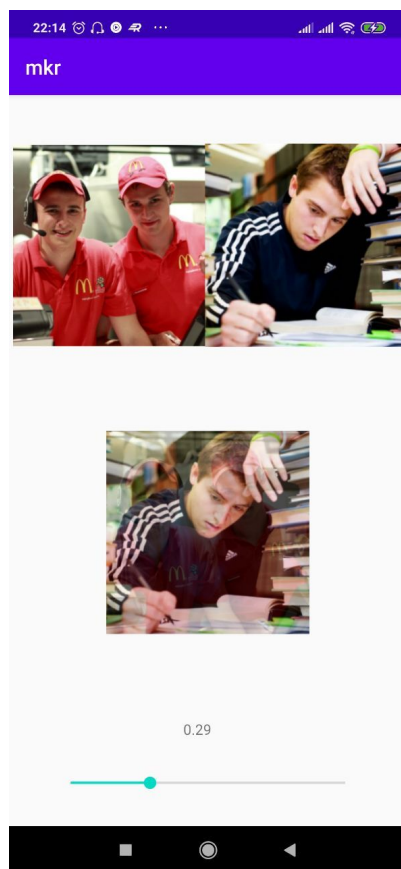


рис 23, поєднання
зображень при лямбда
0.29