



Софийски университет „Св. Климент Охридски“

Факултет по математика и информатика

Курсов проект

по Мобилни приложения

Тема: How fractal is your Droid?

Изготвил: Елена Валентинова Георгиева, ФН: 81555, Компютърни науки, курс 3, поток 1, група 1

Ръководител: ас. Христо Христов

Дата: 30.06.2019

Проверка:

1. Алгоритъм и функционалност

Проектът реализира приложение за Android, което визуализира множеството на Манделброт, дефинирано с формулата:

$$(1) \quad F(Z) = Z^2 + C$$

в равнината на комплексните числа, тоест множеството от точки C в комплексната равнина, за които редицата $\{Z_0, Z_1, Z_2, \dots, Z_n, Z_{n+1}, \dots\}$, построена посредством горната формула, е ограничена, като итерирането започва от $Z_0 = 0$. Може да се докаже, че това изискване е еквивалентно на изискването всички елементи на горната редица да са ограничени в рамките на кръг с радиус 2:

$$C \in M, \text{ дефинирано с формула (1) } \Leftrightarrow \lim_{n \rightarrow \infty} \sup |Z_n| \leq 2$$

С други думи, открием ли член на редицата, който надвишава 2 по абсолютна стойност, знаем, че тя клони към безкрайност, т.е. е неограничена и съответното комплексно число C не принадлежи на множеството на Манделброт. Този факт е в основата на Escape time алгоритъма, който програмата използва, за да генерира множеството: всеки пиксел (x,y) , $x \in [0,width]$, $y \in [0,height]$, се изобразява в точка C от комплексната равнина с координати (c_{re}, c_{im}) , след което с тази точка C програмата итериращо по координатно формула (1), докато или поредното Z_n надхвърли 2 по абсолютна стойност, с което C излиза от множеството на Манделброт (escape condition), или се надхвърли определен максимален брой итерации, зададен предварително, без C да напусне търсеното множество, т.е. в този случай C принадлежи на множеството на Манделброт. На този принцип всички пиксели от растера с размери $width$ и $height$ биват отбелязани като

принадлежащи или не принадлежащи на множеството на Манделброт, като им се асоциира и съответен цвят (черно за точките от множеството, друг цвят за тези извън него).

Приложението удовлетворява следните функционални изисквания:

- (o) При On-tap/On-click върху изображението, приложението реализира Zoom-in;
- (o) Бутон за Zoom-out/Reset;
- (o) Запазване в галерията на устройството и изпращане като attachment на готовото изображение;

2. Архитектура и реализация на приложението

Приложението използва 2 Activity компонента – MainActivity.java и DrawActivity.java.

При стартиране се отваря екран, подканващ потребителя да избере измежду 7 цвята, кодирани в res/values/colors.xml файла в ARGB формат. На всеки избор на цвят от страна на потребителя приложението реагира с изпращане на explicit Intent за смяна на контекста и стартиране на DrawActivity.java, предавайки посочения от потребителя цвят като Extra key-value pair.

Класът DrawActivity.java реализира изчисляването, генерирането и изобразяването на фрактала върху canvas чрез ImageView обект и графичните Android класове Bitmap, Canvas, Paint.

След като генерира изображението, приложението достъпва shared/external storage директорията на устройството и съхранява

там готовото изображение в .png формат в поддиректория /mandelbrot, която създава. Осъществява се достъп до media provider на устройството посредством класа посредник MediaStore и изображението се включва в галерията на устройството.

Приложението реализира zoom in и zoom out при on touch/on click върху изображението чрез ZoomControls widgets, които реагират на съответния MotionEvent.

При натискане на бутона SEND IMAGE приложението изпраща implicit Intent с действие ACTION_SEND, прикачайки изображението като Intent.EXTRA_STREAM и останалата информация за изпращане на email като Intent.EXTRA_* key-value pairs към Intent обекта. Въз основа на действието (action) и данните (data) на Intent обекта в процеса на Intent resolution PackageManager-ът на системата претърсва <intent-filter> информацията в манифестите на всички инсталирани приложения и показва на потребителя списък с приложенията в системата, които са в състояние да изпълнят действието от Intent обекта (в случая email apps).

В AndroidManifest.xml MainActivity е обявена за parent activity на DrawActivity, така че потребителят може многократно да се връща към първоначалния екран на приложението, да избира нов цвят и да генерира поредица от изображения в различни цветове.

```
<activity android:name=".DrawActivity"
          android:parentActivityName=".MainActivity">
  <meta-data
    android:name="android.support.PARENT_ACTIVITY"
    android:value=".MainActivity" />
</activity>
```

Използвани Android APIs:

android.content.Intent
android.graphics.Bitmap
android.graphics.Canvas
android.graphics.Color
android.graphics.Paint
android.graphics.drawable.Drawable
android.net.Uri
android.os.Environment
android.provider.MediaStore
android.support.v7.app.AppCompatActivity
android.os.Bundle
android.view.MotionEvent
android.view.View
android.widget.Button
android.widget.ImageView
android.widget.Toast
android.widget.ZoomControls

Използвани Java classes:

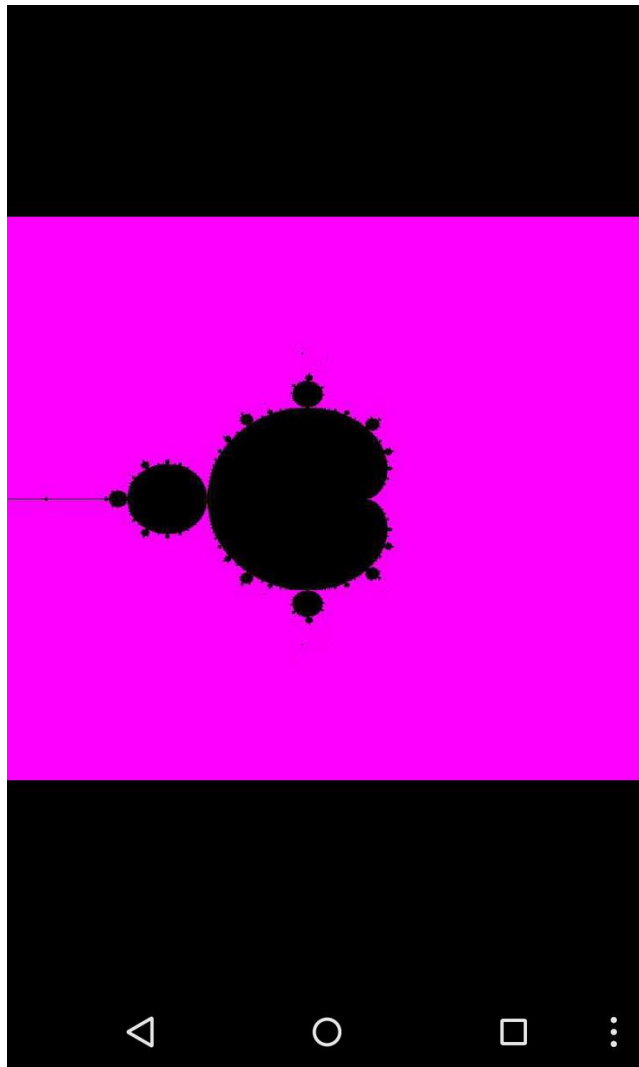
java.io.File
java.io.FileOutputStream
java.io.OutputStream
java.text.SimpleDateFormat
java.util.Date

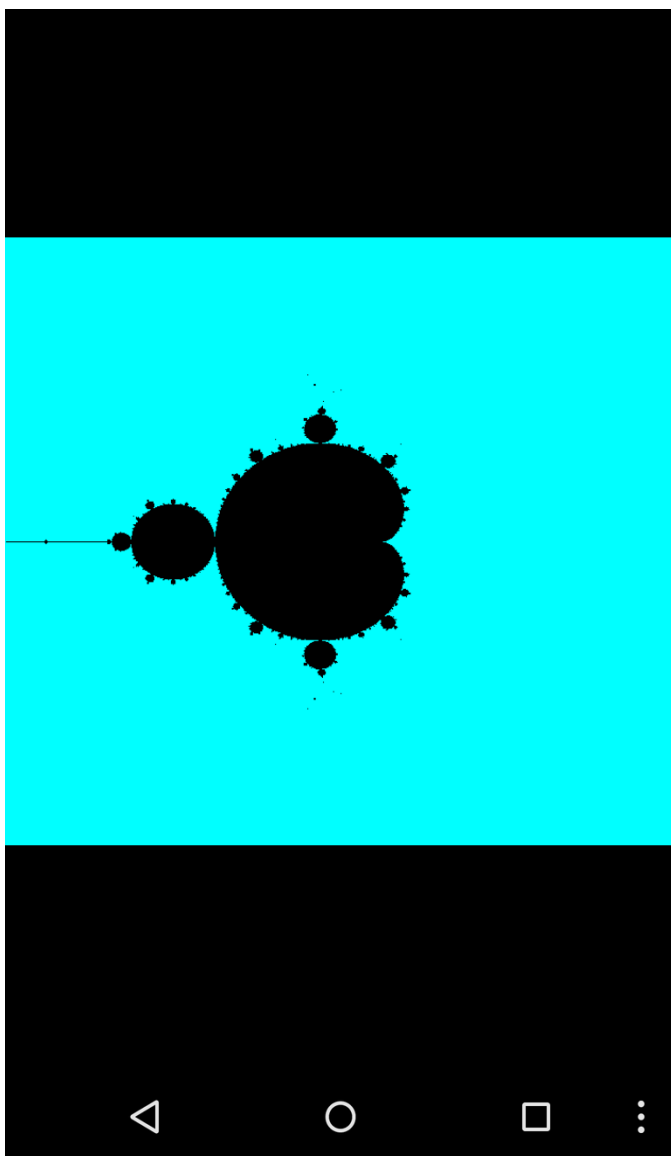
Reference:

<https://developer.android.com/guide>

Mandelbrot, B.B. The Fractal Geometry Of Nature. San Francisco 1982.

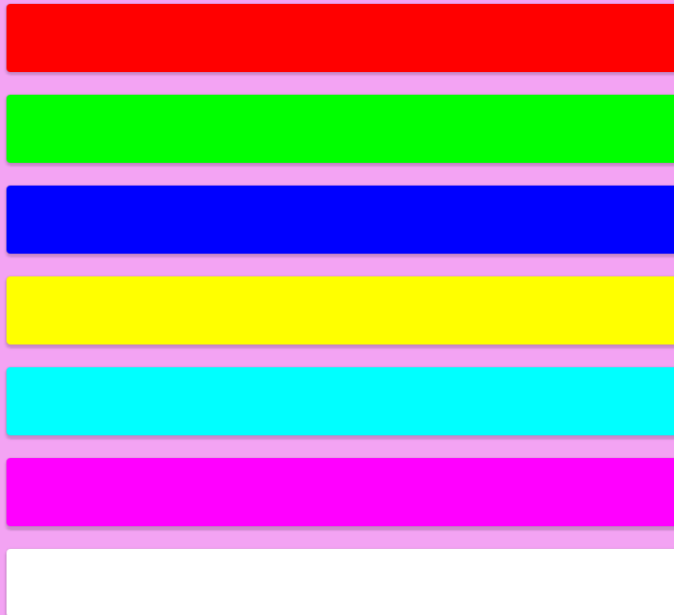
3. Резултати от работата на приложението (Screenshots)





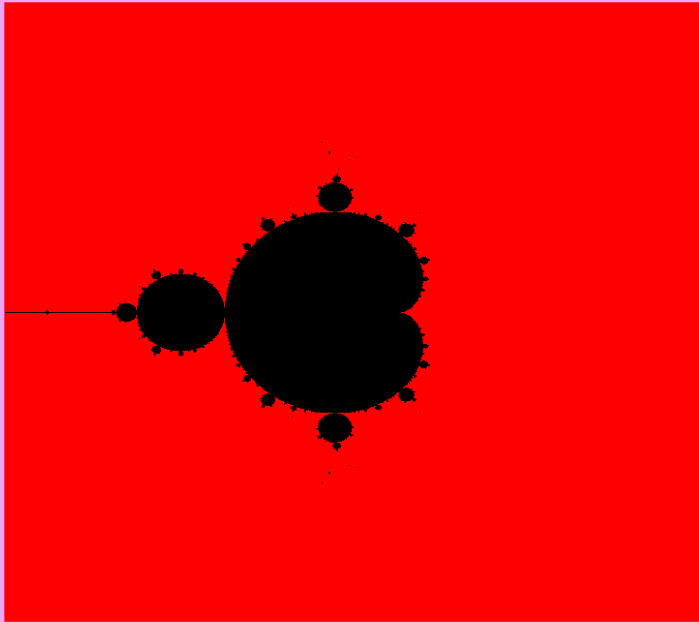
Mandelbrot

Pick a color:



← Mandelbrot

Mandelbrot set:

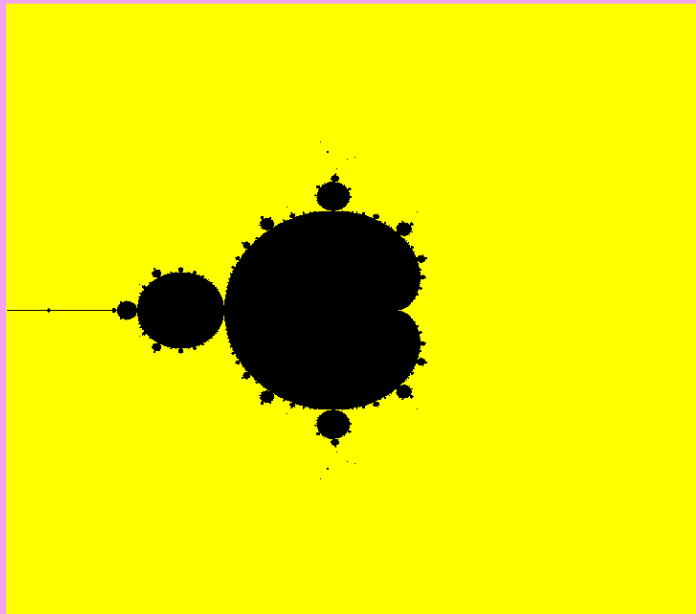


Generated image stored to: /storage/
sdcard/mandelbrot/
g_img_20190625_210515.png

SEND IMAGE

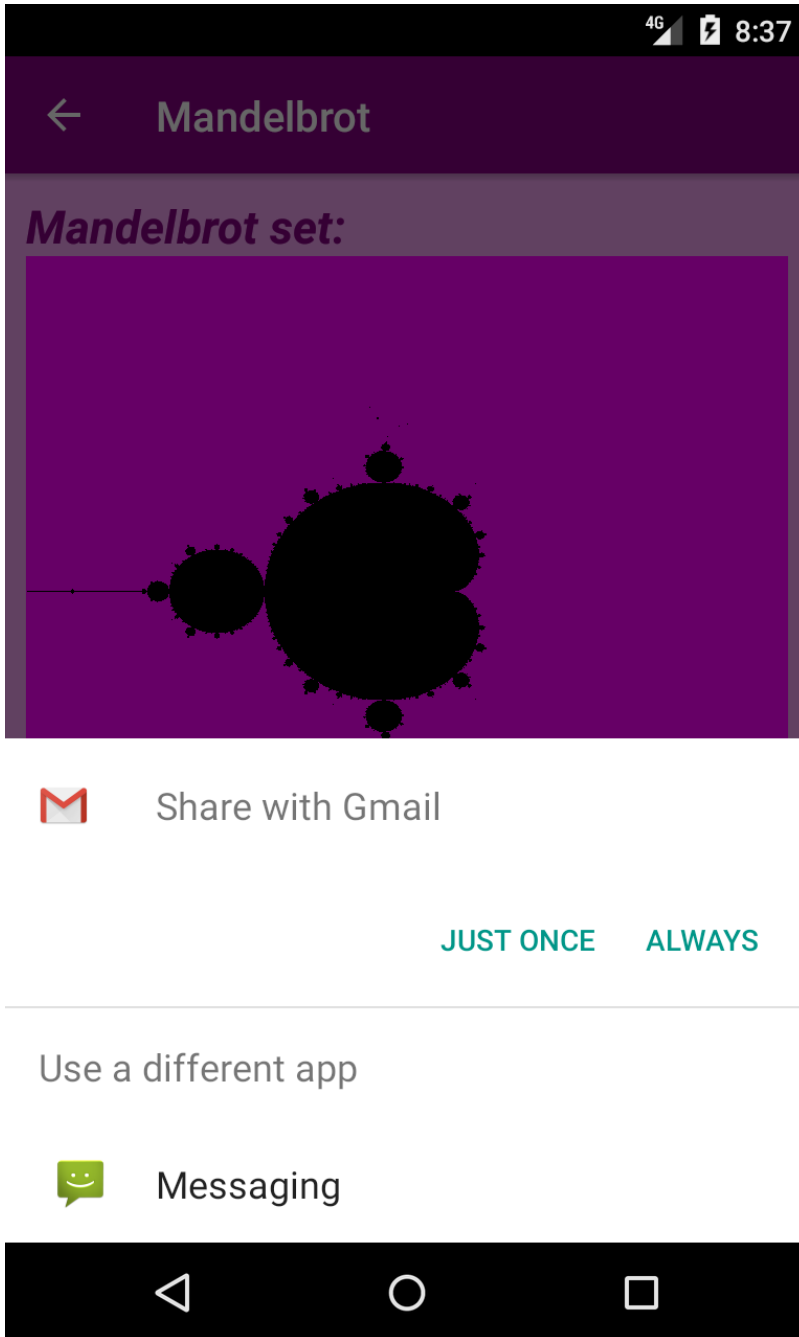
← Mandelbrot

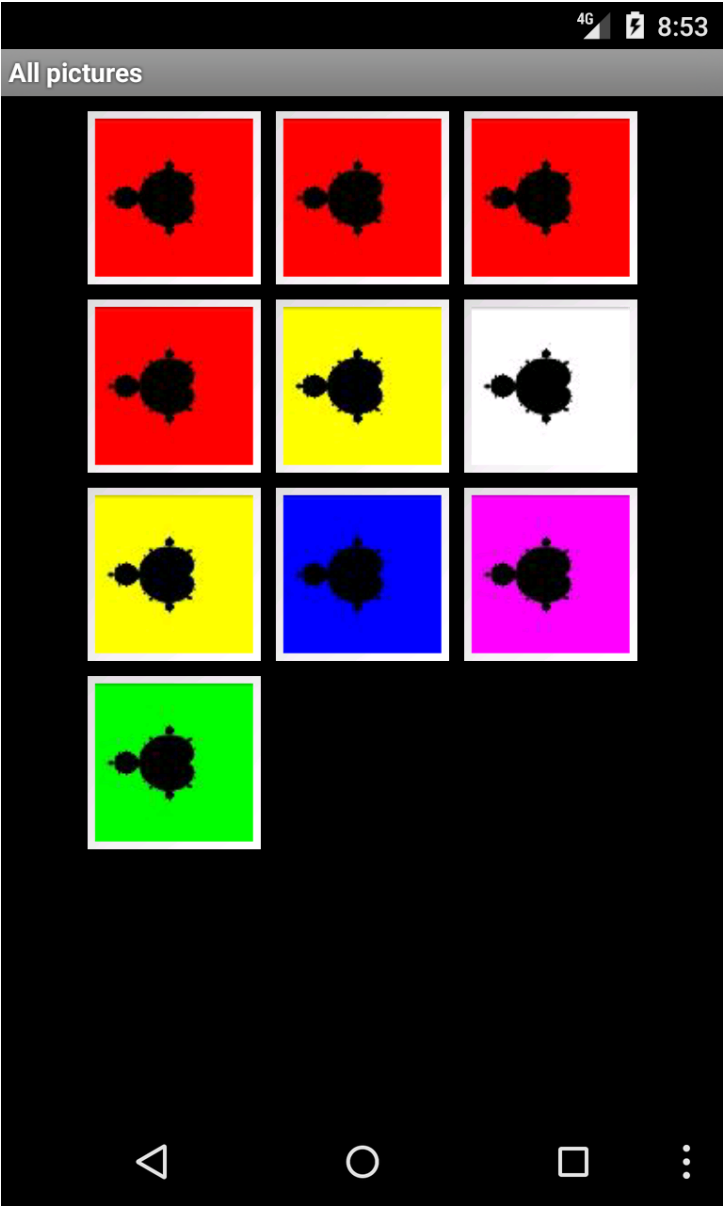
Mandelbrot set:



GENERATE IMAGE

SEND IMAGE





← Mandelbrot

