

Федеральное агентство связи
Сибирский государственный университет телекоммуникаций и
информатики

Кафедра прикладной математики и кибернетики (ПМ и К)

Лабораторная работа «Компас»
по дисциплине «Программирование для мобильных устройств»

Выполнил:

студентка
ИВТ,
группы ИП-813

Захарова К.Ю.

Проверила:

Ассистент кафедры ПМиК

Павлова У.В.

Новосибирск 2021

| | |
|---------------------|---|
| Оглавление | |
| Текст задания | 3 |
| Теория | 3 |
| Скриншоты | 4 |
| Листинг кода | 4 |

Текст задания

Создайте приложение "Компас". На экране отображается циферблат компаса, вращение циферблата осуществляется в зависимости от работы датчика местоположения.

Теория

Как заставить компас реагировать и находить стороны света?

Для этого необходимо воспользоваться датчиками, расположенными в смартфоне.

Данные датчики так же используются во многих приложениях, для которых необходимо точное положение телефона в пространстве (например, 2ГИС, где помимо системы GPS используются датчики-сенсоры телефона).

Для этого было реализовано 2 метода, которые помогают программе вычислить положение смартфона в пространстве и относительно его изголовья (будем считать, что изголовье смартфона находится там, где расположена фронтальная камера).

Методы:

TYPE_MAGNETIC_FIELD-Датчик магнитного поля, определяющий текущие показатели магнитного поля в микротеслах по трём осям.

TYPE_ACCELEROMETER-Измеряет ускорение в пространстве по осям X, Y, Z

К сожалению, в моём телефоне нет датчиков компаса. Но свою работу я проверила на другом смартфоне.

Скриншот



Листинг кода

MainActivity.java

```
package com.example.compass4;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.animation.Animation;
import android.view.animation.RotateAnimation;
import android.widget.ImageView;
import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.view.View;

public class MainActivity extends AppCompatActivity implements SensorEventListener {
    //объявляем картинку для компаса
    private ImageView imageView;
    private final float[] mGravity=new float[3];
    private final float[] mGeomagnetic=new float[3];
    //азимут поворота
    private float azimuth=0f;
    private float correctAzimuth=0f;
    //объявляем работу с датчиком устройства
    private SensorManager mSensorManager;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //Связываем объект ImageView с нашим изображением:
        imageView=(ImageView) findViewById(R.id.imageView);
        //Инициализируем возможность работать с датчиком устройства:
        mSensorManager = (SensorManager) getSystemService(SENSOR_SERVICE);
```

```

    }
    @Override
    protected void onResume(){
        super.onResume();
        //Устанавливаем слушатели ориентации сенсера

        mSensorManager.registerListener(this,mSensorManager.getDefaultSensor(Sensor.TYPE_MAGNETIC_FIELD
        ),SensorManager.SENSOR_DELAY_GAME);

        mSensorManager.registerListener(this,mSensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER)
        ,SensorManager.SENSOR_DELAY_GAME);
    }

    @Override
    protected void onPause(){
        super.onPause();
        //Останавливаем при надобности слушателя ориентации
        //сенсора с целью сбережения заряда батареи:
        mSensorManager.unregisterListener(this);
    }
    @Override
    //гравитация преобразование магнитного поля относительно x y z
    public void onSensorChanged(SensorEvent sensorEvent){
        final float alpha=0.97f;
        synchronized (this){
            if(sensorEvent.sensor.getType()==Sensor.TYPE_ACCELEROMETER){
                mGravity[0]=alpha*mGravity[0]+(1-alpha)*sensorEvent.values[0];
                mGravity[1]=alpha*mGravity[1]+(1-alpha)*sensorEvent.values[1];
                mGravity[2]=alpha*mGravity[2]+(1-alpha)*sensorEvent.values[2];
            }
            if(sensorEvent.sensor.getType()==Sensor.TYPE_MAGNETIC_FIELD){
                mGeomagnetic[0]=alpha*mGeomagnetic[0]+(1-alpha)*sensorEvent.values[0];
                mGeomagnetic[1]=alpha*mGeomagnetic[1]+(1-alpha)*sensorEvent.values[1];
                mGeomagnetic[2]=alpha*mGeomagnetic[2]+(1-alpha)*sensorEvent.values[2];
            }

            //Создаем анимацию вращения:
            float[] R =new float[9];
            float[] I =new float[9];
            boolean success = SensorManager.getRotationMatrix(R,I,mGravity,mGeomagnetic);
            if(success){
                float[] orientation = new float[3];
                SensorManager.getOrientation(R,orientation);
                azimuth = (float)Math.toDegrees(orientation[0]);
                azimuth = (azimuth+360)%360;
                Animation anim =new RotateAnimation(-currentAzimuth,-
                azimuth,Animation.RELATIVE_TO_SELF,0.5f,Animation.RELATIVE_TO_SELF,0.5f);
                currentAzimuth=azimuth;
                //Продолжительность анимации в миллисекундах:
                anim.setDuration(500);
                anim.setRepeatCount(0);
                //Настраиваем анимацию после завершения подсчетных действий датчика:
                anim.setFillAfter(true);
                //Запускаем анимацию:
                imageView.startAnimation(anim);

            }

        }
    }
    @Override
    public void onAccuracyChanged(Sensor sensor,int i){

```

```
}  
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:id="@+id/activity_main"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:paddingBottom="16dp"  
    android:paddingLeft="16dp"  
    android:paddingRight="16dp"  
    android:paddingTop="16dp"  
    tools:context=".MainActivity">  
  
    <ImageView  
        android:id="@+id/imageView"  
        android:layout_width="300dp"  
        android:layout_height="300dp"  
        android:layout_centerInParent="true"  
        android:layout_marginStart="70dp"  
        android:layout_marginTop="116dp"  
        android:layout_marginEnd="70dp"  
        android:layout_marginBottom="38dp"  
        android:contentDescription="@string/todo"  
        app:layout_constraintEnd_toEndOf="parent"  
        app:layout_constraintStart_toStartOf="parent"  
        app:layout_constraintTop_toTopOf="parent"  
        app:srcCompat="@drawable/ig" />  
  
</RelativeLayout>
```