

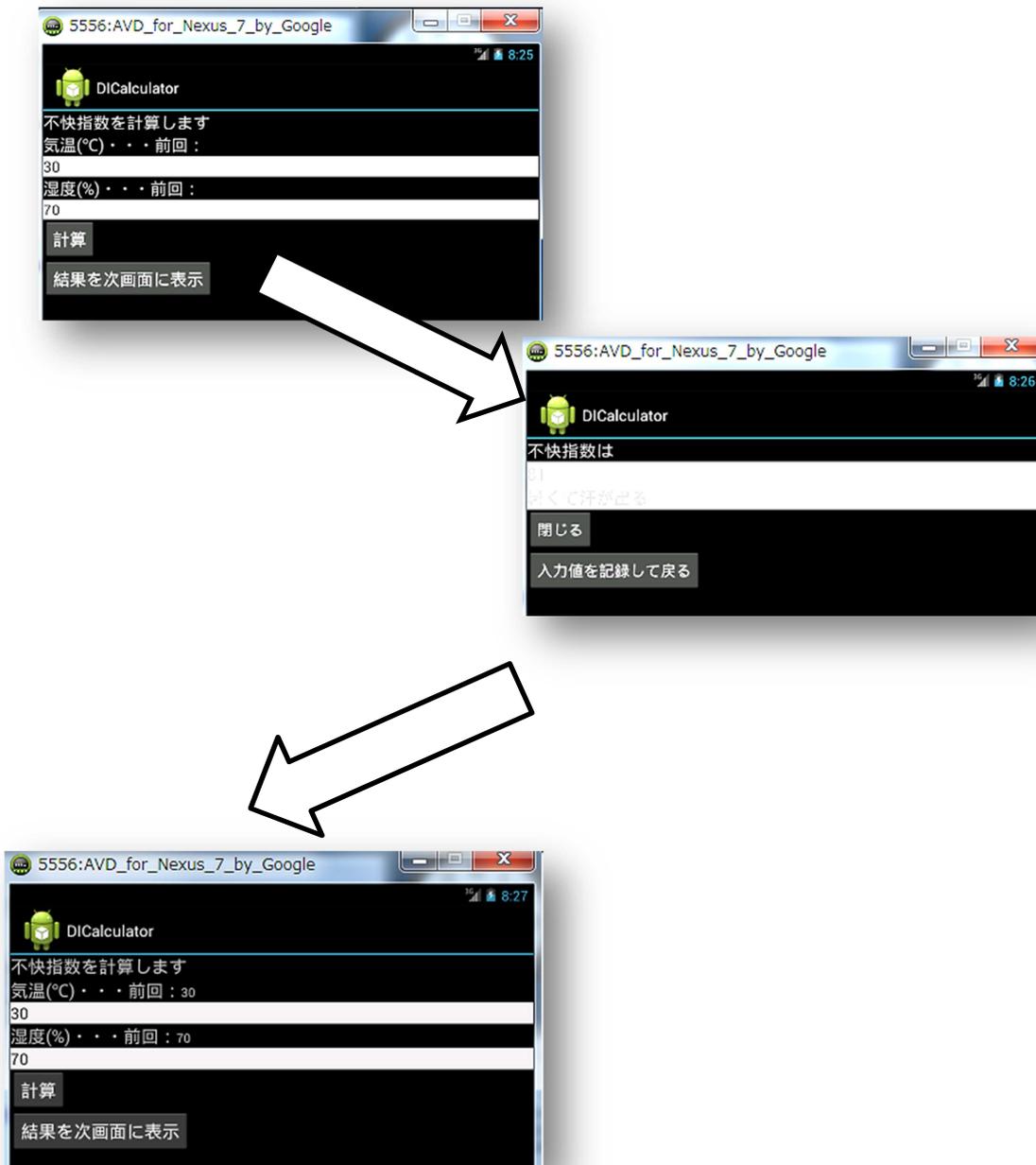
## 7 別画面からの応答を受け取る

startActivityResult メソッドによって起動されたアクティビティは、呼び出し元のアクティビティの子どもの位置づけになります。このような、他のアクティビティの子どもであるアクティビティのことを、「サブアクティビティ」と呼びます。

親のアクティビティは、サブアクティビティが閉じられて、再び親のアクティビティに戻ってきたときに、サブアクティビティからの応答を受け取って処理することができます。

どのアクティビティからの応答であるかを判断するには、startActivityResult メソッドの引数に指定したリクエストコードの内容を利用します。

ここでは、計算結果画面のサブアクティビティから、親のアクティビティに情報を受け渡すようにしてみます。



## 7.1 リソースの追記

網掛け部分を追加してください。

ファイル名 : res/values/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  ...<省略>...
  <string name="label_from_85">暑くてたまらない</string>
    <string name="label_previous_time">...前回</string>
    <string name="button_save_result">入力値を記録して戻る</string>
</resources>
```

## 7.2 レイアウトの追記・修正

網掛け部分を追加してください。

7.2.1 ファイル名 : res/layout/main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent"
  android:orientation="vertical" >
  <TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/label_discomfortIndex"
    android:textSize="20sp"
  />
  <LinearLayout //①
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
  >
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/label_temperature"
    android:textSize="20sp"
  />
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/label_previous_time"
    android:textSize="20sp"
  />
  <TextView
    android:id="@+id/label_previous_temperature"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textIsSelectable="true"
  />
</LinearLayout>
<EditText
  android:id="@+id/text_temperature"
```

```

        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:background="#F0F0F0"
        android:inputType="number"
        android:maxLength="2"
        android:textColor="#000000"
    />
<LinearLayout //②
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/label_humidity"
        android:textSize="20sp"
    />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/label_previous_time"
        android:textSize="20sp"
    />
    <TextView
        android:id="@+id/label_previous_humidity"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textIsSelectable="true"
    />
</LinearLayout>

...

</LinearLayout>

```

### ①②<LinearLayout

今までは、すべてのウィジェットを縦並び (vertical) にしていましたが、一部を横並び (horizontal) に表示させたいために新たに Lineralayout を定義しています。

#### 7.2.2 ファイル名 : res/layout/result.xml

Button の定義を追加します。

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
>

...

<Button
    android:id="@+id/button_close"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

```

```

        android:text="@string/button_close_dialog"
    />
    <Button
        android:id="@+id/button_save_result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/button_save_result"
    />
</LinearLayout>

```

### 7.3 入力用アクティビティ (DICalculatorActivity) の追記・修正

網掛けの箇所を追加してください。

ファイル名： src/jp.edu.mie /DICalculatorActivity.java

```

public class DICalculatorActivity extends Activity
{
    ...
    private TextToSpeech tts;//テキスト読み上げ
    private TextView labelPreviousTemperature;//前回の温度
    private TextView labelPreviousHumidity;//前回の湿度
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        ...
        buttonShowNextActivity
            = (Button)findViewById(R.id.button_show_next_activity);
        labelPreviousTemperature = (TextView)findViewById
            (R.id.label_previous_temperature);
        labelPreviousHumidity = (TextView)findViewById
            (R.id.label_previous_humidity);
        ...
    }
    class calculateClickListener implements OnClickListener
    {
        public void onClick(View view)
        {
            if(view == button)
            {
                ...
            }
            else if(view == buttonShowNextActivity)
            {
                calculate();//不快指数の計算
                TextToSpeech();//テキスト読み上げ
                Intent intent = new Intent(getApplicationContext(),
                    ResultActivity.class);
                intent.putExtra("DI", di);
                intent.putExtra("TE", temperature);
                intent.putExtra("HU", humidity);
                startActivityForResult(intent, 0);
            }
        }
        ...
    }
}
//不快指数の計算

```

```

void calculate()
{
    . . .
}
. . .

class SampleInitListener implements OnInitListener
{
    public void onInit(int status) {}
}
protected void onActivityResult(int requestCode, int resultCode, Intent intent)
{ //①
    if(resultCode == RESULT_OK)
    {
        labelPreviousTemperature.setText(String.valueOf
            (intent.getIntExtra("PREVIOUS_TE", 0)));
        labelPreviousHumidity.setText(String.valueOf
            (intent.getIntExtra("PREVIOUS_HU", 0)));
    }
}
}

```

①protected void onActivityResult(int requestCode, int resultCode, Intent intent)

遷移先の Activity からの応答は、onActivityResult メソッドで受け取ることができます。

パラメータの resultCode には、遷移先の Activity の終了時に setResultCode メソッドで設定した値が渡されます。ここでは、resultCode が「RESULT\_OK」ならば、intent 情報を取得し、widget に設定しています。

| 引数           | 内容   |
|--------------|--|
| requestCode  | 呼び出した Activity を識別するコード。<br>startActivityForResult メソッドで設定したものの。 |
| resultCode   | 処理を行った際のコード  |
| data(intent) | 遷移先で設定したデータ  |

## 【演習】

```

//不快指数の計算
void calculate()
{
    temperature = Integer.parseInt(texttemperature.getText().toString());
    humidity = Integer.parseInt(texthumidity.getText().toString());
    di = (int)(0.81 * temperature + 0.01 * humidity
        * (0.99 * temperature - 14.3) + 46.3);
}

```

の部分で

```

int temperature = Integer.parseInt(texttemperature.getText().toString());
int humidity = Integer.parseInt(texthumidity.getText().toString());

```

にしたら、前回の値が 0 で表示されます。その理由を考えなさい。

## 7.4 結果表示用のアクティビティ (ResultActivity) の追記・修正

網掛けの箇所を追加してください。

ファイル名: `src/jp.edu.mie /ResultActivity.java`

```
public class ResultActivity extends Activity
{
    private int temperature;//温度
    private int humidity;//湿度
    private int di;//不快指数
    private Intent intent;//インテント
    private Button button;//閉じる
    private Button buttonSaveResult;//入力値を記録して戻る
    @Override
    public void onCreate(Bundle savedInstanceState)
    {
        . . .
        button.setOnClickListener(new closeClickListener());
        buttonSaveResult = (Button)findViewById(R.id.button_save_result);
        buttonSaveResult.setOnClickListener(new closeClickListener());//①

        intent = getIntent();
        if (intent != null)
        {
            di = intent.getIntExtra("DI", 0);
            temperature = intent.getIntExtra("TE", 0);
            humidity = intent.getIntExtra("HU", 0);
            diValue.setText(String.valueOf(di));
            . . .
        }
    }
    class closeClickListener implements OnClickListener
    {
        public void onClick(View view)
        {
            if(view == button)
            {
                setResult(RESULT_CANCELED);
                finish(); //自分自身を終了する
            }
            else if(view == buttonSaveResult)
            {
                intent.putExtra("PREVIOUS_TE", temperature);//②
                intent.putExtra("PREVIOUS_HU", humidity);
                setResult(RESULT_OK, intent);//③処理が成功した
                finish(); //自分自身を終了する
            }
        }
    }
}
```

①buttonSaveResult.setOnClickListener(new closeClickListener());

「入力値を記録せずに戻る」ボタンに対して、リスナをセットします。

②intent.putExtra("PREVIOUS\_TE", temperature);

Intent の putExtra メソッドを使いインテントに情報を設定します。インテント情報は、内部的に Map のオブジェクトを保持しており、キーと値のペアで管理されています。