Игра «Камень, ножницы, бумага»

Не закреплены scp, scc, clear

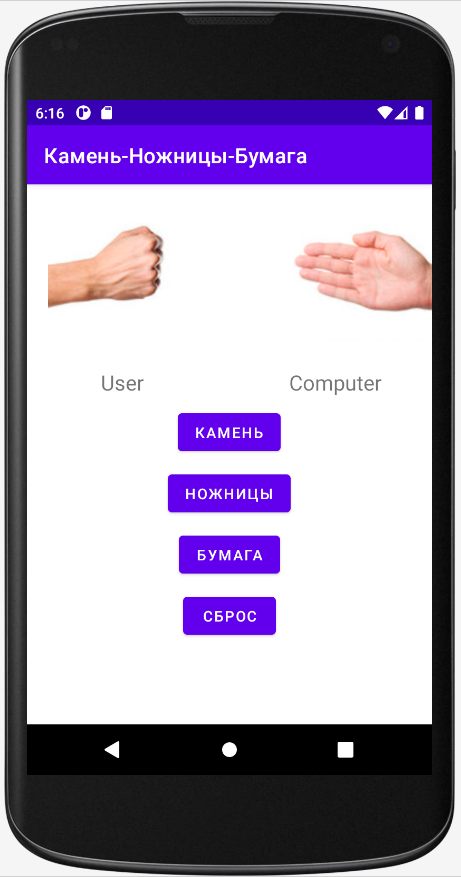
В данном задании реализована старинная игра «Камень, ножницы бумага». Два игрока показывают друг другу жесты, и выигрыш определяется по сочетанию жестов. Приняты следующие правила.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| User | Жест | Computer (жест) | Очки (User) | Очки(Comp) |
| Камень |  | Ножницы | 1 | 0 |
| Бумага | 0 | 1 |
| Ножницы |  | Камень | 0 | 1 |
| Бумага | 1 | 0 |
| Бумага |  | Камень | 1 | 0 |
| Ножницы | 0 | 1 |

Пользователь выбирает жест из списка, а компьютер выбирает жест с помощью числа 1, 2 ил 3, которое задает генератор случайных чисел.

Проделайте следующее.

1. Создайте проект.
2. Изображения, хранящиеся в папке, прилагаемой к заданию, скопируйте в папку drawable.
3. Откройте файл AndroidManifest и в строке label (обычно, строка 8) замените имеющееся в кавычках название проекта на «Камень, ножницы, бумага».
4. Откройте файл activity\_main и разместите показанные на скриншоте элементы:



Изображения: ImageView, тексты: TextView, кнопки Button. При добавлении на экран элементов ImageView выберите начальное значение Empty.

1. Откройте файл activity\_main и замените контейнер Constraints на LinearLayout с вертикальным размещением элементов.
2. Для размещения элементов ImageView вставьте вложенный контейнер LinearLayout с горизонтальным размещением элементов.
3. Для размещения элементов TextView вставьте вложенный контейнер LinearLayout с горизонтальным размещением элементов.
4. Кнопки разместите в оставшейся части исходного контейнера LinearLayout.
5. Назначьте имена:

для ImageView соответственно imgPlayer и imgComp.

Для кнопок сверху вниз: st, sc, pap и reset.

1. Разметка приведена ниже.

<LinearLayout

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:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

tools:context=".MainActivity">

<LinearLayout

android:orientation="horizontal"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content">

<ImageView

android:id="@+id/imgPlayer"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_marginStart="20dp"

app:srcCompat="@drawable/empty"

tools:ignore="ContentDescription" />

<ImageView

android:id="@+id/imgComp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="100dp"

android:layout\_weight="1"

app:srcCompat="@drawable/empty"

tools:ignore="ContentDescription" />

</LinearLayout>

<LinearLayout

android:orientation="horizontal"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_marginStart="70dp"

android:textSize="20sp"

android:text="@string/user" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:textSize="20sp"

android:layout\_marginStart="90dp"

android:text="@string/computer" />

</LinearLayout>

<Button

android:id="@+id/st"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_gravity="center"

android:text="Камень"

tools:ignore="HardcodedText" />

<Button

android:id="@+id/sc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_gravity="center"

android:text="Ножницы"

tools:ignore="HardcodedText" />

<Button

android:id="@+id/pap"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_gravity="center"

android:text="Бумага"

tools:ignore="HardcodedText" />

<Button

android:id="@+id/reset"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_gravity="center"

android:text="Сброс"

tools:ignore="HardcodedText" />

</LinearLayout>

1. Перейдите в файл MainActivity.
2. Объявите глобально:

*// Глобальные объявления*

Button st, sc, pap, reset;*// Кнопки*

ImageView imgPlayer;*// Изображение для Usera*

ImageView imgComp;*// Изображение для Computera*

*// Массив изображений для Computera*

int[] compImg={R.drawable.*stor*, R.drawable.*scissr*,R.drawable.*paper*,R.drawable.*empty*};

Integer scoreComp,scorePlay, res;*// Выборы игроков и результат*

1. Закрепите за элементами имена:

setContentView(R.layout.*activity\_main*);

*// Закрепление имен за элементами управления*

imgPlayer = findViewById(R.id.*imgPlayer*);

imgComp = findViewById(R.id.*imgComp*);

st = findViewById(R.id.*st*);

sc = findViewById(R.id.*sc*);

pap = findViewById(R.id.*pap*);

reset = findViewById(R.id.*reset*);

1. Начальное значение счета

scorePlay=3; *// Начальный счет ! ничья*

1. Добавьте обработчики щелчков по кнопкам.

sc.setOnClickListener(this);

st.setOnClickListener(this);

pap.setOnClickListener(this);

reset.setOnClickListener(this);

1. Оформите метод обработки кнопок:

@SuppressLint("NonConstantResourceId")

@Override

public void onClick(View view) {

int num = 0;*// Значение num - выигрыш для Usera*

switch (view.getId()) {*// Выбранная кнопка*

case R.id.*st*: *// Камень*

imgPlayer.setImageResource(R.drawable.*stol*);

num = playing(1);*// Вызов хода компьютера*

if (num == 1) res = 3; *// Ничья*

if (num == 2) res = 1; *// Comp выбрал ножницы*

else res = 2; *// Comp выбрал бумагу*

break;

case R.id.*sc*:*// Ножницы*

imgPlayer.setImageResource(R.drawable.*scissl*);

num = playing(2);

if (num == 2) res = 3;*// Ничья*

if (num == 1) res = 2;*// Comp выбрал камень*

else res = 1; *// Comp выбрал бумагу*

break;

case R.id.*pap*: *// Бумага*

imgPlayer.setImageResource(R.drawable.*papel*);

num = playing(3);

if (num == 3) res = 3; *// Ничья*

if (num == 2) res = 2; *// Comp выбрал ножницы*

else res = 1; *// Comp выбрал камень*

break;

case R.id.*reset*:*// Возврат в исходное положение*

imgPlayer.setImageResource(R.drawable.*empty*);

scorePlay = 3;

imgComp.setImageResource(compImg[3]);

break;

}

showVisual(res);*// Показать итог кона*

}

1. Оформите метод игры компьютера

public int playing(Integer scorePlay) {*// scorePlay выбор игрока*

Random ran = new Random();*// Числа отсчитываются от нуля*

scoreComp = ran.nextInt(3) + 1;*// Чтобы счет от 1*

*//* if (!scorePlay.equals(scoreComp)) {

switch (scoreComp) {

case 1: *// Вернуть камень*

imgComp.setImageResource(R.drawable.*stor*);

break;

case 2: *// Вернуть ножницы*

imgComp.setImageResource(R.drawable.*scissr*);

break;

case 3: *// Вернуть бумагу*

imgComp.setImageResource(R.drawable.*paper*);

break;

// }

// }

// else scoreComp =scorePlay;*// Ничья ‘эта проверка в другом месте*

return scoreComp;

}

1. Оформите метод графического показа результата. Показ результата задержан с помощью стандартного метода CountDownTimer. Первый параметр при вызове метода задержка времени в миллисекундах. Вторая цифра позволяет выполнить отсчет прошедших тактов (тиканье в часах). В данной задаче не используется.

private void showVisual(int num) {

*// Задержка времени 1-я цифра время задержки (ms), 2-я такт*

new CountDownTimer(3000, 1000) {

public void onFinish() {

switch (num) {

case 1: *// Выиграл игрок*

imgPlayer.setImageResource(R.drawable.*win*);

imgComp.setImageResource(R.drawable.*loss*);

break;

case 2: *// Выиграл компьютер*

imgPlayer.setImageResource(R.drawable.*loss*);

imgComp.setImageResource(R.drawable.*win*);

break;

case 3: *// Ничья*

imgPlayer.setImageResource(R.drawable.*confusion*);

imgComp.setImageResource(R.drawable.*confusion*);

break;

}

}

*// Метод для отсчета тактов (не используется)*

public void onTick(long millisUntilFinished) {

}

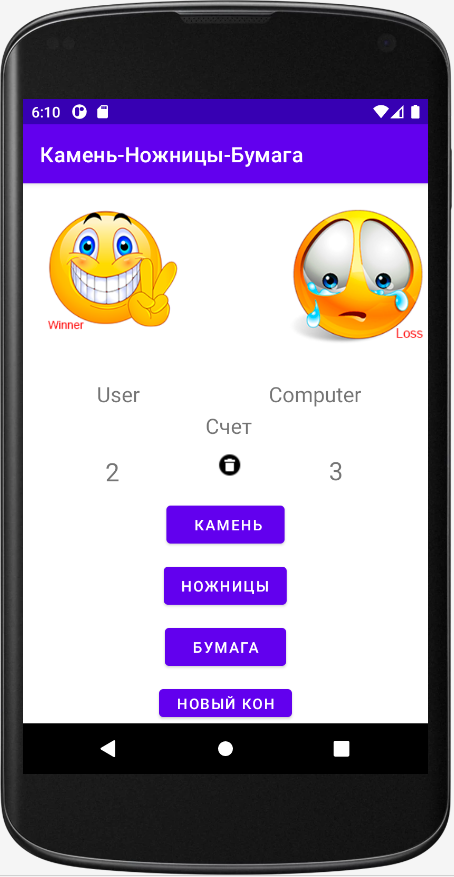
}.start();

}

1. Отладьте программу и предъявите преподавателю.

Для отчета по выполнению задания выполнение до пункта 19 достаточно.

1. Для усовершенствования программы желательно добавить ведение счета игры. Для этого:
   1. Измените контейнер LinearLayout, содержащий надписи: «User» и «Computer», на RelativeLayout.
   2. Добавьте элементы, показанные на экране (три элемента TextView и ImageView – это кнопка для сброса счетчиков)



Внимание. При добавлении элементов система задает значения для операторов вида: android:layout\_marginStart="184dp", android:layout\_marginTop="79dp" и им подобным значения, при которых элемент уходит за пределы экрана. Поэтому либо исключите их, либо задайте небольшие размеры. После этого перетащите надписи по экрану или подберите требуемые значения. Измените размер текста (для надписей textSize="20sp" для результата счета textSize="24sp").

* 1. Имена для вставленных элементов (объявите их глобально):

TextView: scp и scc, для ImageView: clear.

* 1. Разметка

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:id="@+id/textView2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginStart="70dp"

android:layout\_marginTop="10dp"

android:layout\_weight="1"

android:text="@string/user"

android:textSize="20sp" />

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_alignParentEnd="true"

android:layout\_marginTop="10dp"

android:layout\_marginEnd="63dp"

android:text="Computer"

android:textSize="20sp" />

<TextView

android:id="@+id/textView3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="173dp"

android:layout\_marginTop="40dp"

android:text="Счет"

android:textSize="20sp" />

<TextView

android:id="@+id/scp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="78dp"

android:layout\_marginTop="81dp"

android:textSize="24sp" />

<TextView

android:id="@+id/scc"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="290dp"

android:layout\_marginTop="80dp"

android:textSize="24sp" />

<ImageView

android:id="@+id/clear"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentStart="true"

android:layout\_alignParentTop="true"

android:layout\_marginStart="184dp"

android:layout\_marginTop="79dp"

android:clickable="true"

app:srcCompat="@drawable/clear\_circle" />

</RelativeLayout>

* 1. Объявите глобально переменные тип int:

scorep - счет для пользователя

scorec - счет для компьютера

* 1. Закрепите за элементами TextView и ImageView имена.
  2. Установите в результатах счета значения 0.

scp.setText("0");

scc.setText("0");

* 1. Добавьте обработчик щелчка по элементу clear:

clear.setOnClickListener(this);

* 1. В оператор switch добавьте обработку обнуления:

case R.id.*clear*:*// Обнуление счета*

scorep=0; scorec=0; *// Обнулить в памяти*

scp.setText("0"); *// Стереть на экране*

scc.setText("0");

break;

* 1. Добавьте в метод showVisual увеличения счета для результатов, отличных от ничьей (существующие там операторы не трогать).

case 1: *// Этот оператор в методе есть*

scorep++;

scp.setText(String.*valueOf*(scorep));

и

case 2: *// Этот оператор в методе есть*

scoreс++;

scс.setText(String.*valueOf*(scoreс));

* 1. Проверьте работу приложения.