Práctica 01 Programación para dispositivos II

Primera parte (30 puntos)

Diseño de la interfaz, tomando en cuenta dos idiomas y otro estilo de aplicación.

Segunda parte (30 puntos)

Hacer funcionar la parte lógica del programa.

Tercera parte (40 puntos)

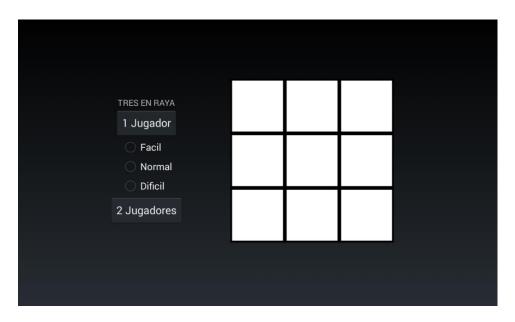
Corregir la inteligencia artificial, de tal forma que no se puede ganar en el modo "difícil"

Estilo de la aplicación.

<style name="AppTheme" parent="android:Theme.Holo.NoActionBar.Fullscreen">

Orientación de la pantalla

```
<activity
    android:screenOrientation="landscape"
</activity>
```



^{*} La práctica se avaluara en clases.

activityMain.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
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"
    tools:context=".MainActivity">
```

```
<LinearLayout
   android:layout marginBottom="8dp"
   android:layout marginEnd="8dp"
   app:layout constraintTop toTopOf="parent"
   <TextView
            android:layout height="wrap content"
            android:layout width="wrap content"
```

```
android:layout width="wrap content"
</LinearLayout>
    android:layout marginStart="8dp"
    android:layout marginTop="8dp"
    app:layout constraintTop toTopOf="parent">
    <ImageView</pre>
    <ImageView</pre>
    <ImageView</pre>
         android:onClick="toque"/>
    <ImageView</pre>
    <ImageView</pre>
```

```
android:id="@+id/cuadro7"
    android:src="@drawable/casilla"
    android:onClick="toque"/>

<ImageView
    android:id="@+id/cuadro8"
    android:src="@drawable/casilla"
    android:onClick="toque"/>

<ImageView
    android:id="@+id/cuadro9"
    android:src="@drawable/casilla"
    android:onClick="toque"/>
    </GridLayout>
</android.support.constraint.ConstraintLayout>
```

MainActivity.java

```
for(int cadaCasilla:casillas)
```

```
imagen=findViewById((cadaCasilla));
        imagen.setImageResource(R.drawable.casilla);
    RadioGroup configDificultad=findViewById(R.id.configDificultad);
    int id=configDificultad.getCheckedRadioButtonId();
    int dificultad=0;
        dificultad=1;
    partida=new Partida(dificultad); /*instanseamos la clase partida*/
    (findViewById(R.id.boton1Jugador)).setEnabled(false);/*desabilita el
public void toque(View vista)
    int casilla=0;/*variable para almacenar el numero de casilla que se
            casilla=i;
```

```
termina(resultado);
    if (jugadores==1)
        marcar(casilla);
private void termina(int resultado)
```

```
toast.show();

/*Comenzar otra ves*/
partida = null;

findViewById(R.id.boton1Jugador).setEnabled(true);
findViewById(R.id.configDificultad).setAlpha(1);
findViewById(R.id.boton2Jugadores).setEnabled(true);

/*Cambia la imagen a circulo o aspa dependiendo el turno del jugador
* que selecciono una casilla*/
private void marcar(int casilla)

{
    ImageView imagen=(ImageView)findViewById(casillas[casilla]);
    if(partida.jugador==1)
    {
        imagen.setImageResource(R.drawable.circulo);
    }
    else
    {
        imagen.setImageResource(R.drawable.aspa);
    }
}
```

Clase Partida

```
return casilla;
    casilla=-1;
if (casilla!=-1)
    return casilla;
        return casilla;
```

```
return casilla;
```

```
* false si la casilla esta marcada con circulo o aspa*/
public boolean comprueba_casilla(int casilla)
{
    if(casillas[casilla]!=0)
    {
        return false;//no se puede marcar
    }
    else
    {
        casillas[casilla]=jugador;
    }
    return true; //se puede marcar
}
```