IT 372 – Final Exam

May 29, 2019 **Name _____**

Part A: Multiple Choice Questions. Circle the letter of the correct response for each question. Give an optional reason for each question for partial credit. If you circle the correct response, the reason will not be considered. Answer all 10 questions. 5 points each.

1. What is the default name of the Java activity file in an Android Studio project?
   a. `activity_main.java` b. `ActivityMain.java`
   c. `main_activity.java` d. `MainActivity.java`

2. What is the file extension of an Android executable file?
   a. `.apk` b. `.class` c. `.dex` d. `.exe`

3. What was the first hand-held computer, introduced in 1993?
   a. Blackberry b. Newton Message Pad c. Palm Pilot d. Quark

4. In 2011 Apple sued which Android device manufacturer for patent/copyright infringement?
   a. Dell b. Motorola c. Qualcom d. Samsung

5. The Kotlin language is primarily used for developing apps for which platform?
   a. Android b. Apple iOS c. Ubuntu d. Windows

6. The Swift language is primarily used for developing apps for which platform?
   a. Android b. Apple iOS c. Ubuntu d. Windows

7. In which Android Studio project folder the resource file `colors.xml` located?
   a. `app/main/src/res/values` b. `app/src/main/res/values`
   c. `res/app/src/values` d. `src/main/res/app/values`

8. Which attribute of a `LinearLayout` causes the layout items to be arranged horizontally.
   a. `android:angle="90"` b. `android:horizontal="true"`
   c. `android:orientation="horizontal"` d. `android:vertical="false"`

9. In the layout file for an activity, which attribute sets the text color to teal for a widget?
   a. `android:foreColor="0x00f0f0"` b. `android:foreColor="0x008080"`
   c. `android:textColor="0x00f0f0"` d. `android:textColor="0x008080"`

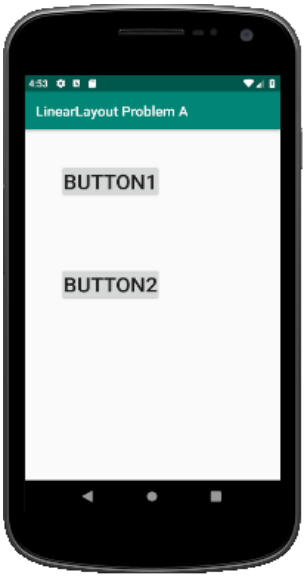
10. The layout file `activity_main.xml` is specified like this:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:andriod="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="5dp"
        android:layout_margin="5dp"
        android:textSize="30sp"
        android:text="Button1" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="50dp"
        android:layout_margin="50dp"
        android:textSize="30sp"
        android:text="Button2" />
</LinearLayout>
```

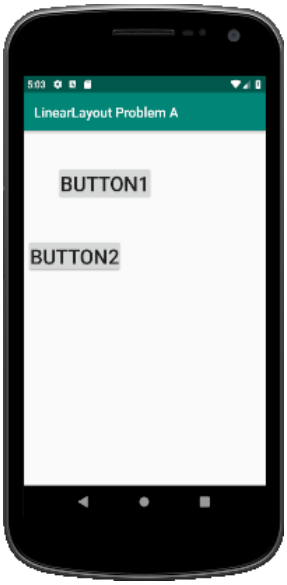
Which emulator display is produced?

a.



b.

c.



d.

11. The emulator displays this `LinearLayout` in landscape mode:

The XML resource files `colors.xml, strings.xml, styles.xml, and activity_main.xml` are shown on Page 4. Which choice of `TextView` code matches the emulator display?

a.
```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/darkGray"
    android:layout_margin="50dp"
    android:text="@string/t_1" />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/lightGray"
    android:layout_margin="20dp"
    android:text="@string/t_1" />
```

b.
```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/lightGray"
    android:layout_margin="20dp"
    android:text="@string/t_2" />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/darkGray"
    android:layout_margin="50dp"
    android:text="@string/t_2" />
```

c.
```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/lightGray"
    android:layout_margin="50dp"
    android:text="@string/t_1" />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/lightGray"
    android:layout_margin="20dp"
    android:text="@string/t_2" />
```

d.
```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
     android:background="@color/darkGray"
    android:layout_margin="20dp"
    android:text="@string/t_2" />
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/lightGray"
    android:layout_margin="20dp"
    android:text="@string/t_1" />
```

Resource files and layout file for Question 11 on Page 3.

>>>> colors.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008577</color>
    <color name="colorPrimaryDark">#00574B</color>
    <color name="colorAccent">#D81B60</color>
    <color name="darkGray">#808080</color>
    <color name="lightGray">#d0d0d0</color>
</resources>
```

>>>> strings.xml:

```
<resources>
    <string name="app_name">Linear Layout Problem B</string>
    <string name="t_1">Sample Text 1</string>
    <string name="t_2">Sample Text 2</string>

</resources>
```

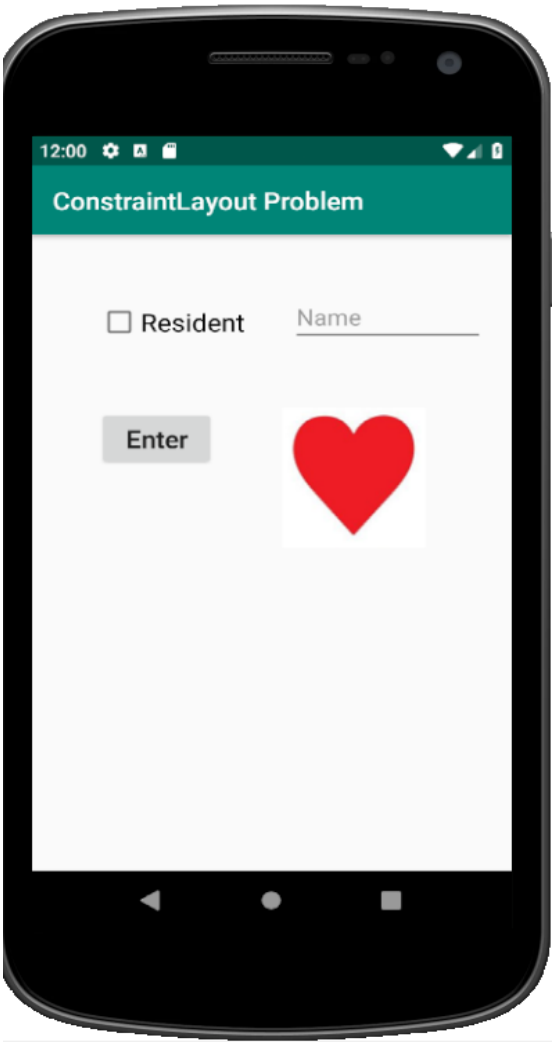
>>>> styles.xml

```
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
        <item name="android:textSize">30dp</item>
    </style>
</resources>
```

>>>> activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    tools:context=".MainActivity">

</LinearLayout>
```



12. The app on the left was set up with `ConstraintLayout`. Which constraints are needed to ensure to position the `ImageView` widget, which displays the heart.

a.
```
android:layout_marginLeft="500dp"
android:layout_marginTop="500dp"
app:layout_constraintLeft_toRightOf=
    "@+id/button"
app:layout_constraintTop_toBottomOf=
    "@+id/editText"
```

b.
```
android:layout_marginLeft="50dp"
android:layout_marginTop="50dp"
app:layout_constraintLeft_toRightOf=
    "@+id/editText"
app:layout_constraintTop_toTopOf=
    "@+id/button"
```

c.
```
android:layout_marginLeft="50dp"
android:layout_marginTop="50dp"
app:layout_constraintLeft_toRightOf=
    "@+id/button"
app:layout_constraintTop_toBottomOf=
    "@+id/editText"
```

d.
```
android:layout_marginLeft="50dp"
android:layout_marginTop="500dp"
app:layout_constraintLeft_toRightOf=
    "@+id/button"
app:layout_constraintTop_toRightOf=
    "parent"
```

13. To cause the emulator display to switch from portrait to landscape or vice versa, depending on the emulator orientation, in which file is this statement placed?

```
android:screenOrientation="fullSensor"
```

a. `AndroidManifest.xml` b. `ActivityMain.java`
c. `main_activity.xml` d. `styles.xml`

14. Which attribute of an `EditText` widget sets gray text, which disappears when the user starts typing in the widget?

a. `android:grayText` b. `android:hint`
c. `android:prompt` d. `android:tempText`

15. Which `Activity` method obtains the width of the current layout? (The current layout is set by the `setContentView` method.)
    a. `getWidth` b. `layoutWidth` c. `width` d. `viewWidth`

16. For what is a `Bundle` object usually used?
    a. to create Java objects from the widgets in an activity.
    b. to pass data in the layout to a new activity.
    c. to save the layout data from an activity immediately before that activity is destroyed.
    d. to save the layout data from an activity to a database immediately before that activity is destroyed.

17. Which attribute in the `Spinner` node of the layout file sets the id for the spinner to `spnr_month`?
    a. `android:id="@id/spnr_month"` b. `android:id="@+id/spnr_month"`
    c. `android:id="id/spnr_month"` d. `android:id="spnr_month"`

18. Which of these `ArrayList` methods appends an item to the end of an `ArrayList` collection object?
    a. `add` b. `append` c. `concatenate` d. `insert`

19. In which file is the string specified that appears in the title bar of an Android app?
    a. `AndroidManifest.xml` b. `ActivityMain.java`
    c. `main_activity.xml` d. `strings.xml`

20. Which of the following attributes sets `EditText` widget control to only allow numbers to be entered.
    a. `android:inputChars="number"` b. `android:inputChars="text"`
    c. `android:inputType="number"` d. `android:inputType="text"`

21. For an Android app, which method is used to create a new SQLite database?
    a. The constructor of a class derived from `SQLiteOpenHelper`.
    b. The constructor of a class derived from `SQLiteDatabase`.
    c. The `onCreate` method of a class derived from `SQLiteOpenHelper`.
    d. The `onCreate` method of a class derived from `SQLiteDatabase`.

22. Which statement deletes all rows from the database table `trans`, where the database is represented by the `SQLiteDatabase` object `db`?
    a. `db.execSQL("delete * from trans");`
    b. `db.execSQL("delete all from trans");`
    c. `db.execSQL("delete from trans");`
    d. `db.execSQL("drop trans");`

23. What does a `Cursor` object contain?

a. the column number of a database table
b. the result of a database query
c. the row number of a database table
d. the SQL statement for a query

24. If the `ContentValues` object `cv` is defined and populated like this:

```
ContentValues cv = new ContentValues( );
cv.put("id", 1234name);
cv.put("date", "5/29/19");
cv.put("gender", 35.98);
```

Which statement inserts the data row defined by `cv` into the table `trans`?

a. `db.insert("trans", cv);`
b. `db.insert("trans", cv, null);`
c. `db.insert("trans", null, cv);`
d. `insert(db, "trans", null, cv);`

25. If A is a class and `I` is an interface that requires the method `f` defined like this:

```
public interface I {
    public int f( );
}
```

Which of these statements is the syntax correct for instantiating an anonymous inner class that implements `I`?

a.
```
I i = new I( ) {
      @Override
      public int f(  ) {
          return 123;
      }
  }
```

b.
```
I i = new I( ) {
      @Override
      public int f(  ) {
          return 123;
      }
  };
```

c.
```
I i = new I( {
      @Overrides
      public int f(  ) {
          return 123; }
  })
```

d.
```
I i = new I( {
      @Override
      public int f(  ) {
          return 123; }
  });
```

26. An object of which class is used to set the stroke color for drawing a line with the method `c.drawLine`, where `c` is a `Canvas` object.

a. `Canvas` b. `Drawing` c. `Paint` d. `View`

27. Which statement in the current activity launches the new activity `SecondaryActivity`?

a. `(new Intent(this, new SecondaryActivity( )).startActivity( );`
b. `(new Intent(this, SecondaryActivity.class)).startActivity( );`
c. `startActivity(new Intent(this, new SecondaryActivity( ));`
d. `startActivity(new Intent(this, SecondaryActivity.class))`

28. The `EditText` and `TextView` widgets `edtxtAmount` and `txtTip` are obtained from their layout IDs like this:
```
EditText edtxtAmount = (EditText) findViewByID(R.id.amount);
TextView txtTip = (TextView) findViewByID(R.id.tip);
```
Which `onClick` event handler in the `MainActivity.java` file is correct for the `Compute Tip` button? Assume that a valid float is entered in the EditText widget for the amount. The tip is computed as 18% of the `amount`, which is the amount of the check in a restaurant.

a.
```
public void onClick(View view) {
      txtTip.setText(String.valueOf(0.18 * Double.parseDouble(
          edtxtAmount.getText( ).toString( ))));
  }
```

b.
```
public void onClick(View view) {
      txtTip.setText(0.18 * Double.parseDouble (
          edtxtAmount.getText( )));
  }
```

c.
```
public void onClick(View view) {
      txtTip.setText(18.0 * Integer.parseInt(
          edtxtAmount.getText( ).toString( )));
  }
```

d.
```
public void onClick(View view) {
      txtTip.setText(String.valueOf(18.0 * Integer.parseInt(
          edtxtAmount.getText( )));
  }
```

29. The layout in the `activity_main.xml` is defined like this:
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools=http://schemas.android.com/tools
    android:id="@+id/linear_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/txt_touchpoint"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="(0, 0)" />

</LinearLayout>
```

A widget `mv`, instantiated from the `MyView` class, derived from the `View` class, is dynamically added to the layout like this:

```
LinearLayout layout = (LinearLayout)
findViewById(R.id.linear_layout);
TextView tv = (TextView) findViewById(R.id.txt_touchpoint);
MyView mv = new MyView(this, tv);
layout.addView(mv);
```

We pass the `TextView` object `tv` to the `MyView` constructor so we can change the text of `tv` within the `mv` object.

A `MyView` object is supposed to detect `Touch` events and display them in the TextView with ID `txt_touchpoint`. For example, if the point `X=45, Y=132` is touched, `(45, 132)` is displayed in the textview. Which of the choices on pages 9, 10, and 11 defines the `MyView` class with its source code lines in the correct order? (Three of the choices have their lines in the wrong order.) The object `tv` must be declared as `final` to be able access it within the inner class.

a.

```
public class MyView extends View {
    private float x, y;
    public MyView(Context context, TextView tvParam) {
        super(context);

        final TextView tv = tvParam;
        float x = e.getX();
        float y = e.getY();
        this.setOnTouchListener(new View.OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent e) {
                if (e.getAction( ) == MotionEvent.ACTION_UP) {
                    String displayValue = String.format(
                        Locale.getDefault( ),"(%.2f,%.2f)", x, y);
                    tv.setText(displayValue);
                }
                return true;
            }
        });
    }
}
```

b.

```
public class MyView extends View {
    private float x, y;
    public MyView(Context context, TextView tvParam) {
        super(context);

        final TextView tv = tvParam;
        this.setOnTouchListener(new View.OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent e) {
                if (e.getAction( ) == MotionEvent.ACTION_UP) {
                    float x = e.getX();
                    float y = e.getY();
                    String displayValue = String.format(
                        Locale.getDefault( ),"(%.2f,%.2f)", x, y);
                    tv.setText(displayValue);
                }
                return true;
            }
        });
    }
}
```

c.

```
public class MyView extends View {
    private float x, y;
    public MyView(Context context, TextView tvParam) {
        super(context);

        final TextView tv = tvParam;
        this.setOnTouchListener(new View.OnTouchListener( ) {
        });

        @Override
        public boolean onTouch(View v, MotionEvent e) {
            if (e.getAction( ) == MotionEvent.ACTION_UP) {
                float x = e.getX();
                float y = e.getY();
                String displayValue = String.format(
                    Locale.getDefault( ),"(%.2f,%.2f)", x, y);
                tv.setText(displayValue);
            }
            return true;
        }
    }
}
```

d.

```
public class MyView extends View {
    private float x, y;
    public MyView(Context context, TextView tvParam) {
        final TextView tv = tvParam;
        super(context);

        this.setOnTouchListener(new View.OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent e) {
                if (e.getAction( ) == MotionEvent.ACTION_UP) {

                    String displayValue = String.format(
                        Locale.getDefault( ),"(%.2f,%.2f)", x, y);
                    tv.setText(displayValue);
                    float x = e.getX();
                    float y = e.getY();
                }
                return true;
            }
        });
    }
}
```

30. Which Android Studio window is used to view directories and files (for example databases) that reside on the emulator?

a. Device File Explorer b. Emulator Settings c. Favorites d. Project Explorer