

59.5

9

Mobile Computing - Android Exam 1 - 60 points

Name Harika Sahini (5546653)

Time: 75 minutes

This exam is closed book, closed notes, closed for electronic devices. The only resource you are allowed to use is your brain!

Question 1) [5] For each of the following states of an activity fill in the table:

State	Is the activity visible?	Is the activity able to interact with a user?	What method will be called as the activity enters the state?
Stopped	No	No	onStop()
Paused	Partially visible	No	onPause()
Resumed	Yes	Yes	onResume()

Question 2) [4] Explain the difference between an explicit and an implicit intent. Give an example of when you would use each one.

Explicit Intent: Explicit intent is the one which you would call by name or subclass name. These are used when you wish to call an activity from another activity in your own app. Ex:- You can use this to move to another activity which is specific in your app.

Implicit Intent: These intents are called by Action. We can specify Action, data, category to know which action the intent is going to perform.

Ex:- To open a web browser of some URL from your app you will use implicit intent.

Question 3) [10] Place the views in this grid layout. Clearly indicate the edges of each text view.

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<GridLayout
```

```
    android:id="@+id/gridLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="4"
    android:rowCount="4" >
```

```
<TextView
```

```
    android:layout_width="100dp"
    android:layout_height="50dp"
    android:layout_column="1"
    android:layout_columnSpan="2"
    android:layout_row="0"
    android:text="She" />
```

```
<TextView
```

```
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:layout_column="2"
    android:layout_columnSpan="2"
    android:layout_row="2"
    android:layout_rowSpan="2"
    android:text="loves" />
```

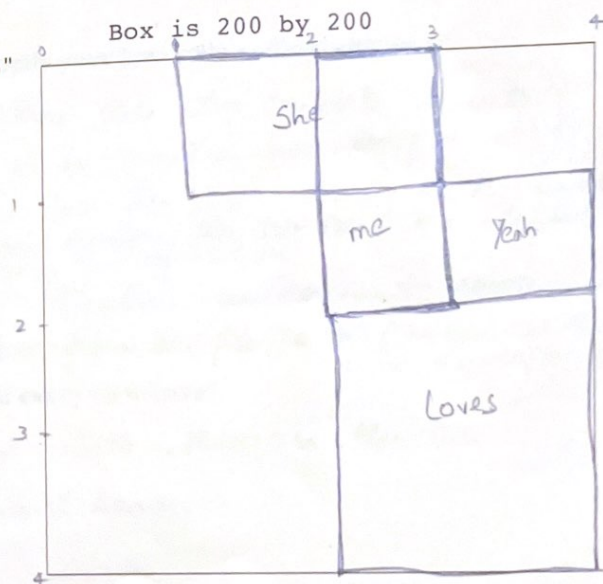
```
<TextView
```

```
    android:layout_width="50dp"
    android:layout_height="100dp"
    android:layout_column="2"
    android:layout_row="0"
    android:layout_rowSpan="2"
    android:text="me" />
```

```
<TextView
```

```
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_column="3"
    android:layout_row="1"
    android:text="Yeah" />
```

```
</GridLayout>
```



Question 4) [9]

Part A: Create a little code fragment that will

- 1) Create an intent for activity Booking.
- 2) Add the following key value pairs to the intent
- 3) "location" => "Aruba"
- 4) "days" => 5
- 5) Start the Booking activity for a result.

Intent intent = new Intent(this, Booking.class);

intent.putExtra("location", "Aruba");

intent.putExtra("days", 5);

startActivity(intent);

startActivityForResult(intent, requestCode)

C
O
D
E

Part B: Create a little code fragment in the Booking activity that will

- 1) Create an intent that will be returned.
- 2) Add the key-value pair "cost" => 2523.90
- 3) Set the result with an error code of 0
- 4) End the activity

Intent in = getIntent();

Double cost = ~~getDoubleExtra~~ ^{putExtra}("cost", 2523.90);

setResult(0, in);

finish();

C
O
D
E

Question 5) [4]

A) Explain the steps when you compile your Java code ending with an APK.

When we compile the Java code, JVM converts it into byte code, from there it is converted into .dex files using Dalvik virtual machine. These dex file contains all the meta data of app components and therefore the JVM executes the code & helps in launching the application.

B) Explain the role of ART.

ART stands for Android Runtime environment responsible for running the android application from dex file. The ART features are still under development.

Question 6) [2] What two attributes must every view have?

layout-height and layout-width, these are the two attributes every view should have.

Question 7) [10] Place the views in this linear layout on the screen. Clearly indicate the edges of each text view and the text content. (One character is about 10dp in height and width.)

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

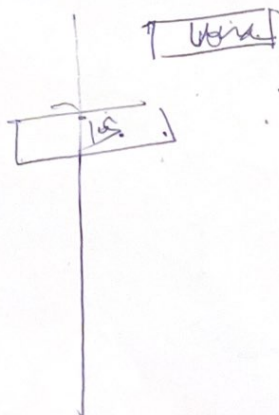
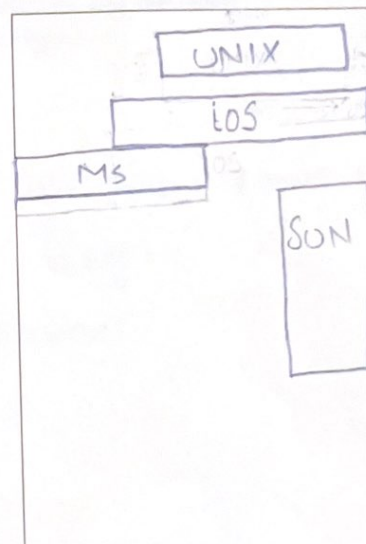
    <TextView
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:text="UNIX"
        android:layout_gravity="right" />

    <TextView
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:gravity="center_horizontal"
        android:text="iOS" />

    <TextView
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_gravity="left"
        android:text="MS" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="100dp"
        android:text="Sun" />
</LinearLayout>
```

Screen is 200 by 500



Question 8) [10] Suppose that you have a layout for an activity that contains the following edit text and button

```
<EditText
    android:id="@+id/cookiesET"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="5"
/>

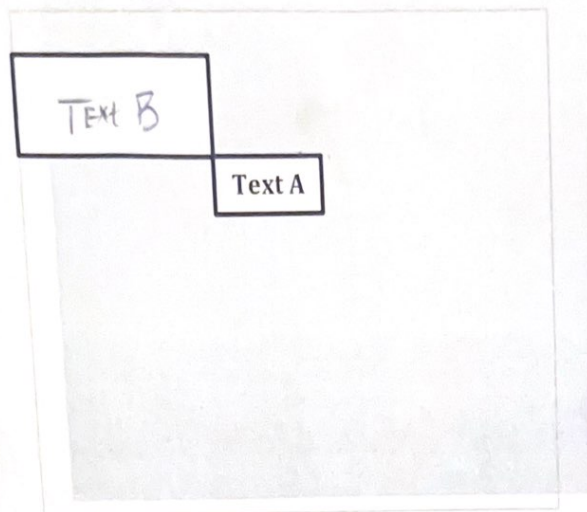
<Button
    android:id="@+id/eatBTN"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:onClick="eatAction"
    android:text="Eat a cookie"
/>
```

Create a method that when the button is pressed will take the text from the EditText, convert it to an integer, subtract 1 from the value and put the result back into the edit text. You must prevent a crash on bad input.

C
O
D
E

```
public void eatAction(View v) {
    EditText et = (EditText) findViewById(R.id.cookiesET);
    try {
        String textVal = et.getText().toString();
        int i = Integer.parseInt(textVal);
        i = i - 1;
        et.setText(i);
    }
    catch (Exception e) {
        Log.i(e);
        throw e;
    }
}
```

Question 9) [6] Suppose that we have the following components in a constraint layout with the boundary shown. Consider each of the following relations and write YES/NO to indicate if the relation holds. Assume that the margin and bias are both 0.



A) In TextA: layout_constraintTop_toBottomOf="TextB"

Yes

B) In TextA: layout_constraintBottom_toTopOf="TextB"

No

C) In TextA: layout_constraintStart_toEndOf="TextB"

Yes

D) In TextB: layout_constraintTop_toTopOf="parent"

No

E) In TextB: layout_constraintStart_toStartOf="parent"

Yes

F) In TextB: layout_constraintStart_toStartOf="TextA"

No