



The University of Fiji

School of Science and Technology

Department of Computer Science/ Information Technology

ITC391

Special Topics in ICT

Semester I, 2013

FINAL EXAMINATION QUESTION PAPER

Time Allowed: 3 Hours [10 Minutes Extra Allowed For Reading the Paper]
Weighting: 50%
Total Marks: 100 marks.

Part I:

Multiple Choice	-	20 marks
-----------------	---	----------

Part II:

Short Answer Question 1	-	16 marks
Short Answer Question 2	-	16 marks
Short Answer Question 3	-	18 marks
Short Answer Question 4	-	30 marks

Total	100 marks
--------------	------------------

Instructions

- Show your answers clearly in each question to gain maximum marks.
- You may ask for extra sheets from the supervisors if the need arises.
- Do not forget to write your name and ID number on all the extra sheet(s) attached.
- Programmable calculators are not allowed.

Section A: Multiple Choices

1. ____ is the first open-source technology platform for mobile devices.
a. Apple iOS
b. Blackberry OS
c. Google Android
d. Microsoft Phone 7
2. Android apps are written in ____.
a. C++
b. Fortran
c. Java
d. Python
3. A(n) ____ is a single element such as a TextView, Button, or CheckBox control.
a. app
b. layout
c. modality
d. widget
4. Designing a program is analogous to ____.
a. constructing a building
b. doing laundry
c. painting with watercolor
d. performing an experiment
5. A ____ organizes layout components in a vertical column or horizontal row.
a. Bag layout
b. Graphical
c. Linear layout
d. Relative layout
6. The most popular free Android app is ____.
a. Facebook
b. Google Maps
c. Pandora
d. YouTube
7. The ____ method creates an intent to start another Activity.
a. beginActivity
b. createActivity
c. prepareActivity
d. startActivity
8. The set of resources available to an app is referenced by ____.
a. res
b. R
c. S
d. V
9. To display text in a TextView control, use the method ____.
a. assignText()
b. displayText()
c. setText()
d. showText()
10. The recommended prefix for a launcher icon is ____.
a. ic_launcher
b. ic_start
c. launch_icon
d. start_icon
11. The intent action ____ opens a Web browser on the Android.
a. ACTION_VIEW
b. BROWSE_VIEW
c. HTTP_VIEW
d. WEB_VIEW
12. The TextView property for backgrounds is ____.
a. application
b. background
c. bkgrd
d. image

13. A(n) ____ in Java executes a one-time task.
- a. alarm
 - b. cron
 - c. ticker
 - d. timer
14. On an Android, 1000 milliseconds is roughly equivalent to ____ second(s).
- a. 0.1
 - b. 1
 - c. 10
 - d. 100
15. The method that hides an Activity is ____.
- a. onDestroy
 - b. onResume
 - c. onStart
 - d. onStop
16. Typically you add an ImageView control by dragging the control onto the ____.
- a. emulator
 - b. Gallery
 - c. Palette
 - d. XML code
17. The last argument to the onItemClick method determines the row id of the selected item, which is most useful for a ____.
- a. GridView
 - b. ListView
 - c. Spinner
 - d. TextView
18. A common aspect ratio is ____.
- a. 4:3
 - b. 3:4
 - c. 16:8
 - d. 10:6
19. AVD is short for ____.
- a. Aftermarket Value Depreciation
 - b. Alternative Viewing Display
 - c. Android Virtual Devices
 - d. Anonymous Virtual Design
20. The DatePickerDialog constructor requires ____ argument(s).
- a. one
 - b. two
 - c. three
 - d. five

Section B: Short Answer Questions

Question 01

(16 marks)

- a) Define what each of the following codes does in an android application: **(13 marks)**
- i. `<EditText android:id="@+id/title"`
`android:layout_width="fill_parent"`
`android:layout_height="wrap_content" />`
 - ii. `<TextView android:layout_width="wrap_content"` → 1
`android:layout_height="wrap_content"`
`android:text="@string/date" />`
`<Button` → 4
`android:id="@+id/reminder_date"`
`android:layout_height="wrap_content"`
`android:layout_width="wrap_content"`

```

/>
<TextView android:layout_width="wrap_content"           → 9
    android:layout_height="wrap_content"
    android:text="@string/time" />
<Button                                           → 12
    android:id="@+id/reminder_time"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content"
/>

```

```

iii.  mDateButton.setOnClickListener(new View.OnClickListener() {   → 1
        @Override
        public void onClick(View v) {                               → 4
            showDialog(DATE_PICKER_DIALOG);                          → 5
        }
    });
    updateDateButtonText();                                         → 8
    updateTimeButtonText();                                         → 9

```

- b) The Android system has a framework built around dialog boxes that allows you to provide any implementation that you may need. List down various types of dialog boxes that are available. (3 marks)

Question 2 (16 marks)

- a) Unfortunately, a built-in Android validation framework does not exist. Hopefully in future versions of the Android platform this will be introduced. However, you have ways to validate input with the current framework. The method in which you provide validation to the user is up to you. What are some common methods in which developers implement validation? (3 marks)
- b) Android provides a simple mechanism for you to add menus to your applications. Define what are the different types of menus? (3 marks)
- c) A bad menu is a menu that provides very little (if any) helpful text in the menu description and provides no icon. What are few common menu faux pas included in a bad menu. (2 words)
- d) Given the ReminderListActivity class file, explain which parameters that are passed. (3 marks)

```

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent intent)
{
    super.onActivityResult(requestCode, resultCode, intent);
    // Reload the list here
}

```

- e) Therefore, you have various ways in which you can grab the user's attention without stealing the main focus away from his current activity. List and explain these mechanisms. (3 marks)
- f) The layout for adding and editing is fairly simple because the form contains very few fields. List and explain what these fields are. (2 marks)

Question 3 (18 marks)

- a) Intent provides a facility for performing late runtime binding between the codes in different applications. Its most significant use is in the launching of activities, where it can be thought of as the glue between activities. Intent can be launched into the message bus system a couple of ways, list these different ways. (3 marks)
- b) The Android operating system is open platform. Describe what is open platform? (1 mark)
- c) For Android devices to be certified compatible (devices have to be compatible to have access to the Android Market), they must follow certain hardware guidelines. What are these guidelines? (2 marks)
- d) The Android ADT gives you two options for creating launch configurations. What are these two types of configurations? (1 mark)
- e) List the subfolders that appear after the Hello Android project is expanded in the package explorer. (1 mark)
- f) The application manifest file provides all the essential information to the Android system — information that it must have before it can run any of your application's code. List other information that the application manifest file provides. (2 marks)
- g) This XML file defines exactly what your view is to look like.

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

What will these lines of code produce once build and executed. (1 mark)

- h) Complete the table below: (2 marks)

Layout	Description
LinearLayout	

RelativeLayout	
FrameLayout	
TableLayout	

- i) Given the screen size density write the icon size for each of the different densities:
(1 mark)

- a. Ldpi - _____
- b. Mdpi - _____
- c. Hdpi - _____

- j) An activity is a single, focused thing a user can do. Explain what onCreate() and onPause() activities do. Also what are the four states of an activity? (2 marks)

- k) Explain what does these do: (2 marks)

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    Button toggleButton = (Button)findViewById(R.id.toggleButton);
    toggleButton.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
        }
    });
}
```

Question 4 (30 marks)

- i. The values resource can contain many different types of value type resources for your application. List and explain four of the values in android package. (2 marks)
- ii. Each of these classes plays a vital role in Android as well as in the app widget framework. Explain what these classes do. (2 marks)
 - a. The Intent
 - b. The BroadcastReceiver
 - c. The AppWidgetProvider
 - d. The IntentService
 - e. The AppWidgetProviderInfo
- iii. For the following programs given at the bottom explain what each of the lines are coding indicated by the line numbers: (26 marks)

```
i. public class AppWidget extends AppWidgetProvider {
    @Override
    public void onReceive(Context context, Intent intent) {
        if (intent.getAction() == null) {
            context.startService(new Intent(context,
                ToggleService.class));
        } else {
            super.onReceive(context, intent);
        }
    }
}
```

→ 6

```

    }
    }
    @Override
    public void onUpdate(Context context, AppWidgetManager
    appWidgetManager, int[] appWidgetIds) {
        context.startService(new Intent(context,
        ToggleService.class)); → 16
    }
    public static class ToggleService extends IntentService { → 19
    public ToggleService() {
        super("AppWidget$ToggleService"); → 22
    }
    @Override
    protected void onHandleIntent(Intent intent) { → 26
        ComponentName me=new ComponentName(this, AppWidget.class); → 27
        AppWidgetManager mgr=AppWidgetManager.getInstance(this); → 28
        mgr.updateAppWidget(me, buildUpdate(this)); → 29
    }
    private RemoteViews buildUpdate(Context context) { → 30
        RemoteViews updateViews=new
        RemoteViews(context.getPackageName(),R.layout.widget); → 32
        AudioManager audioManager =
        (AudioManager)context.getSystemService(Activity.AUDIO_SERVICE); → 34
        if(audioManager.getRingerMode() ==
        AudioManager.RINGER_MODE_SILENT) {
            updateViews.setImageDrawableResource(R.drawable.phone_state,
            R.drawable.phone_state_normal); → 40
        }
        audioManager.setRingerMode(AudioManager.RINGER_MODE_NORMAL);
        } else {
            updateViews.setImageDrawableResource(R.drawable.phone_state,
            R.drawable.phone_state_silent); → 45
            audioManager.setRingerMode(AudioManager.RINGER_MODE_SILENT);
        }
        Intent i=new Intent(this, AppWidget.class); → 49
        PendingIntent pi
        = PendingIntent.getBroadcast(context, 0, i,0); → 52
        updateViews.setOnClickListener(R.id.phone_state,pi); → 54
        return updateViews; → 56
    }
    }
    }

```

- ii. <?xml version="1.0" encoding="utf-8"?>
 <ScrollView
 xmlns:android="http://schemas.android.com/apk/res/android"
 android:layout_width="fill_parent"
 android:layout_height="fill_parent"> → 5

```

<LinearLayout                                     → 6
    android:orientation="vertical"                 → 7
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/title" />                → 12
    <EditText android:id="@+id/title"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content" />        → 15
    <TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/body" />                 → 18
    <EditText android:id="@+id/body"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:minLines="5"
    android:scrollbars="vertical"
    android:gravity="top" />                       → 24
    <TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/date" />                 → 27
    <Button
    android:id="@+id/reminder_date"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content" />         → 31
    <TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/time" />                 → 34
    <Button
    android:id="@+id/reminder_time"
    android:layout_height="wrap_content"
    android:layout_width="wrap_content" />         → 38
    <Button android:id="@+id/confirm"
    android:text="@string/confirm"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />        → 42
</LinearLayout>
</ScrollView>

```

The End!!