```
    /** Performs animation on a Button. */
    private void buttonAnimation(Button button) {
        Animation animation = AnimationUtils.loadAnimation(this, R.anim.animation);
        button.startAnimation(animation);
    }
}
```

Hint:

Elements:

android:startOffset
The amount of milliseconds the animation delays after start() is called.

[10]

Question 4 [11 marks]

(a) The preference framework in Android provides a flexible and convenient mechanism for manipulating shared preferences. The following listings define a preference screen associated with the game "Breakout". "Breakout" is a game in which the player controls the movement of a paddle horizontally to bounce a ball and hit some bricks. The player wins when all bricks have been hit and gone, and loses when the ball reaches the bottom of the play area.
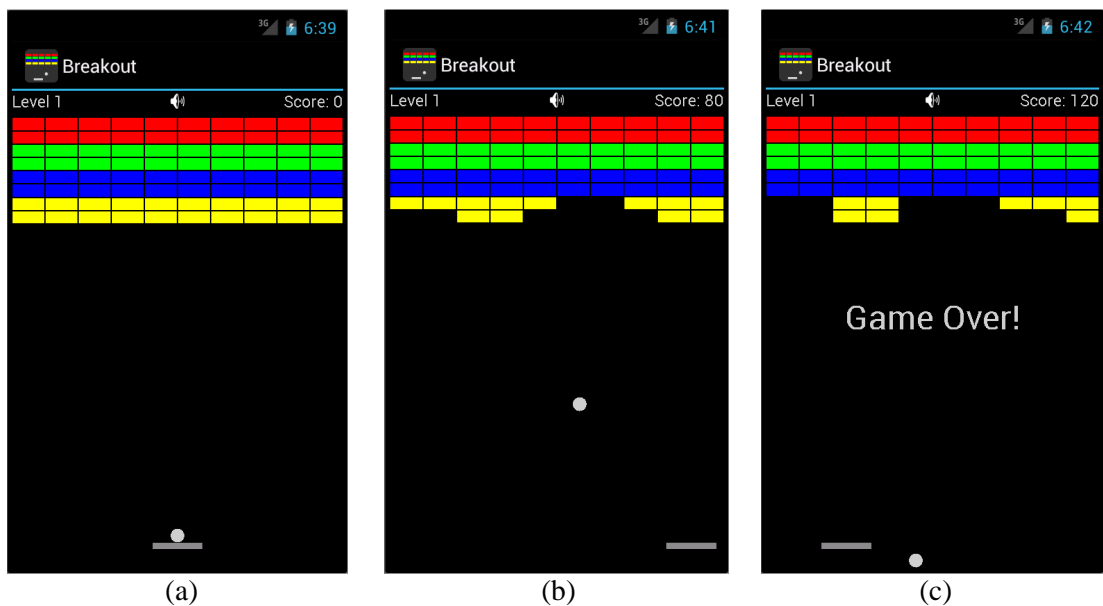

Figure 4. Game "Breakout".

Listing below depicts part of the string resource file "string.xml".

```
…
    <!-- Game Breakout -->
    <!-- Key preference items and the corresponding default value -->
    <string name="breakout_game_mode_key">breakout_game_mode</string>
    <string name="breakout_brick_pattern_key">breakout_brick_pattern_key</string>


    <!-- String array format -->
    <string-array name="breakout_game_mode">
```

```
        <item>Normal</item>
        <item>Ball x3</item>
    </string-array>


    <!--  String array format -->
    <string-array name="breakout_brick_pattern">
        <item>Colored bricks</item>
        <item>Picture bricks</item>
        <item>Metallic bricks</item>
    </string-array>
…
```

Listing below depicts the preference screen layout resource file "game_prefs.xml".

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >
    <PreferenceCategory
        android:title="Game setting">

        <ListPreference
            android:key="@string/breakout_game_mode_key"
            android:title="Game Mode"
            android:summary="Playing mode of the game"
            android:entries="@array/breakout_game_mode"
            android:entryValues="@array/breakout_game_mode"
            android:defaultValue="Normal" />

        <ListPreference
            android:key="@string/breakout_brick_pattern_key"
            android:title="Brick Pattern"
            android:summary="Brick pattern selection"
            android:entries="@array/breakout_brick_pattern"
            android:entryValues="@array/breakout_brick_pattern"
            android:defaultValue="Colored bricks" />

    </PreferenceCategory>
</PreferenceScreen>
```

Illustrate with diagram about what you would see when the preference screen defined in the layout resource file "game_prefs.xml" is shown. It is also required to **sketch** with diagram if preference component such as CheckBoxPreference, EditTextPreference, ListPreference, or MultiSelectListPreference is selected.

[8]

(b) Providing sound effect or background music in games is one of the ways that can increase the game playability and arouse interest. **Suggest** and **describe** one game option which can be applied in the "Breakout" game for that purpose.
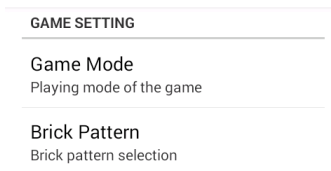
[3]
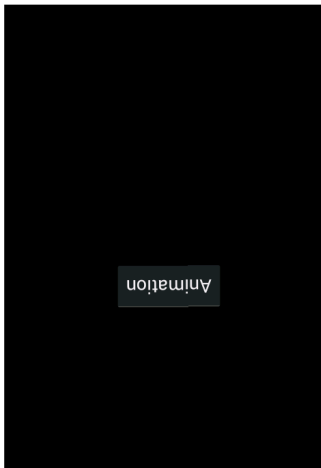
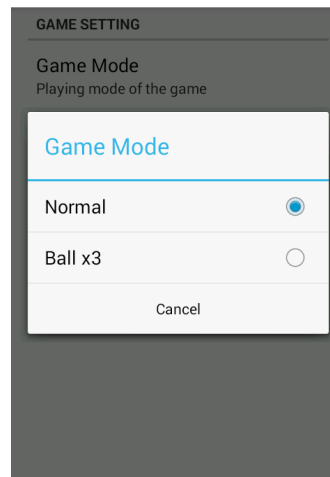**Illustration of animation with diagram.**

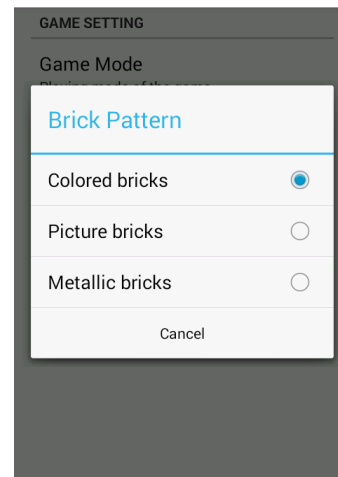**Question 4** [11 marks]

(a)

(a) Preference screen.   (b) Game Mode.   (c) Brick Pattern.

Figure 4. Game preferences.

(b)   **Progressive movement of bricks**
- **It also has the single bat and ball, but as the ball hits the paddle, the entire wall gradually advances downwards step by step.**

Question 4 [11 marks]

(a) "Obstacle Racing" is a sport game in which the player needs to overcome different forms of obstacles. The player can tap on the screen to jump over the obstacles as many as he/she can. The game ends when the player collided with the obstacle.

(a)   (b)

Figure 4. Game "Obstacle racing".

Listing below depicts part of the string resource file "string.xml".

```
…
  <!--  Game Obstacle racing -->
  <!--  Key preference items and the corresponding default value -->
  <string name="obstacle_racing_game_mode_key">obstacle_racing_game_mode</string>
  <string name="obstacle_racing_background_key">obstacle_racing_theme_key</string>

  <!--  String array format -->
  <string-array name="obstacle_racing_game_mode">
    <item>Normal</item>
    <item>Time attack</item>
  </string-array>

  <string-array name="obstacle_racing_theme">
    <item>Sunshine</item>
    <item>Sunset</item>
    <item>Snow</item>
  </string-array>
…
```
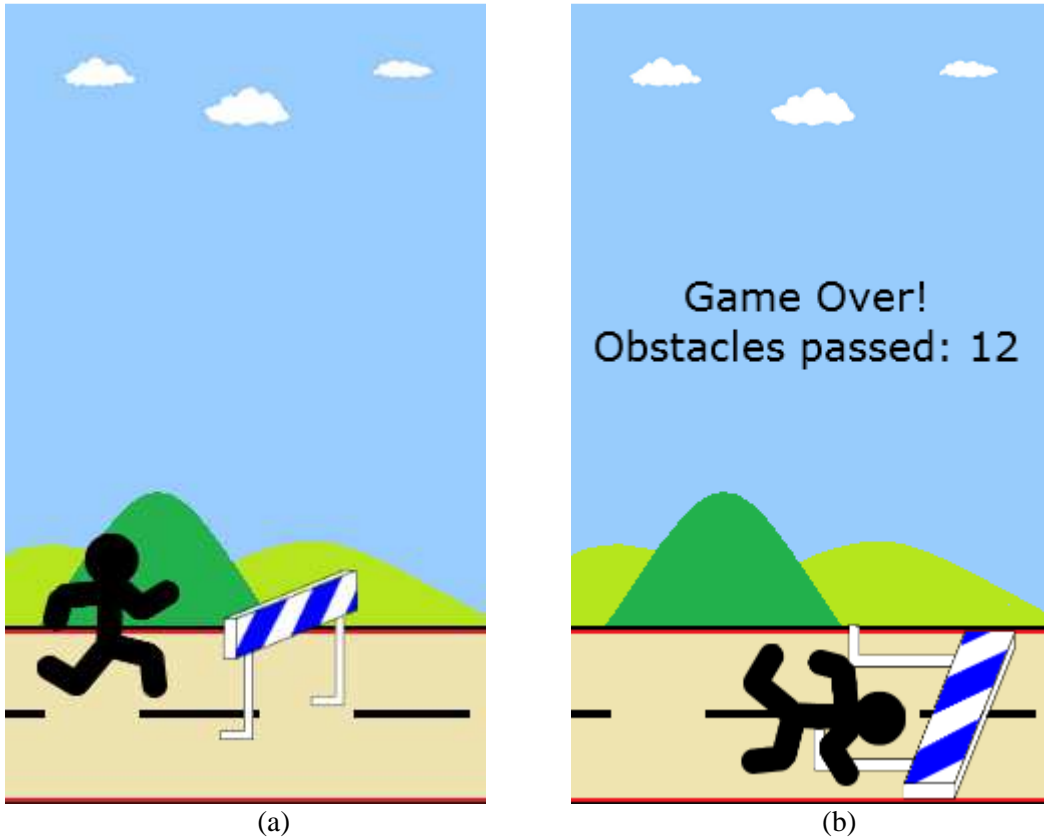
Listing below depicts the preference screen layout resource file "game_prefs.xml".

```xml
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >

    <PreferenceCategory
        android:title="Game Mode setting">
        <ListPreference
            android:key="@string/obstacle_racing_game_mode_key"
            android:title="Game Mode"
            android:summary="Playing mode of the game"
            android:entries="@array/obstacle_racing_game_mode"
            android:entryValues="@array/obstacle_racing_game_mode"
            android:defaultValue="Normal" />
    </PreferenceCategory>

    <PreferenceCategory
        android:title="Game Theme setting">
        <ListPreference
            android:key="@string/obstacle_racing_background_key"
            android:title="Game Theme"
            android:summary="Background of the game"
            android:entries="@array/obstacle_racing_theme"
            android:entryValues="@array/obstacle_racing_theme"
            android:defaultValue="Sunshine" />
    </PreferenceCategory>

</PreferenceScreen>
```

Illustrate with diagrams about what you would see when the preference screen defined in the layout resource file "game_prefs.xml" is shown. **Sketches** with diagram if preference component such as CheckBoxPreference, EditTextPreference, ListPreference, or MultiSelectListPreference is selected.
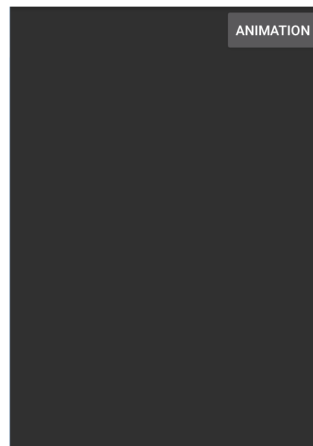
[8]

(b) Providing sound effect and background music or different game modes and themes are different ways that can arouse interest. **Suggest** and **describe** one game option which can be applied in the "Obstacle racing" game for increasing the game playability.

[3]
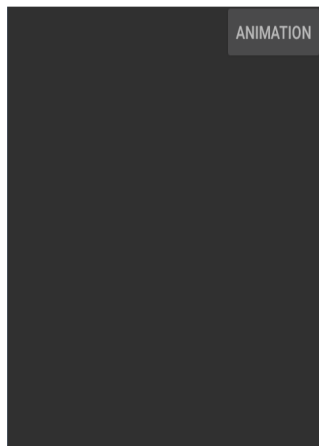
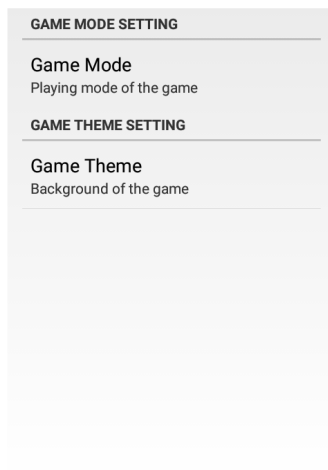(d)

(e) End of animation.
After 12s

Illustration of animation with diagrams.
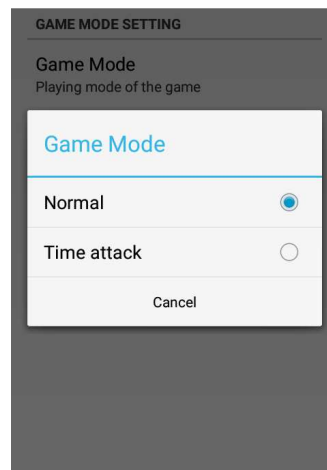
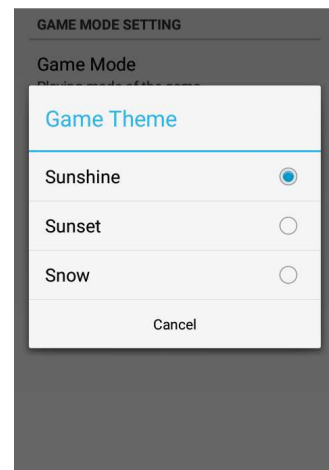## Question 4 [11 marks]

(a)

(a) Preference screen.   (b) Game Mode.   (c) Game Theme.

Figure 4. Game preferences.

(b)   **Progressive speed up of the player**
-    **The running speed of the player will be increased for certain time period.**

Question 4 [9 marks]

(a) "Balloon Piercing" is a game in which player needs to blow the balloon as big as it can. After the game started, the balloon is blown automatically. The player can stop it by tapping on the screen. Once the balloon is stopped from blowing, the game ends and the size of the balloon will be sketched on the screen. However, there is a thumb-pin inside the game. The game ends when the balloon is pierced by the thumb-pin and "Game Over" message will be shown to the player.

(a) (b)
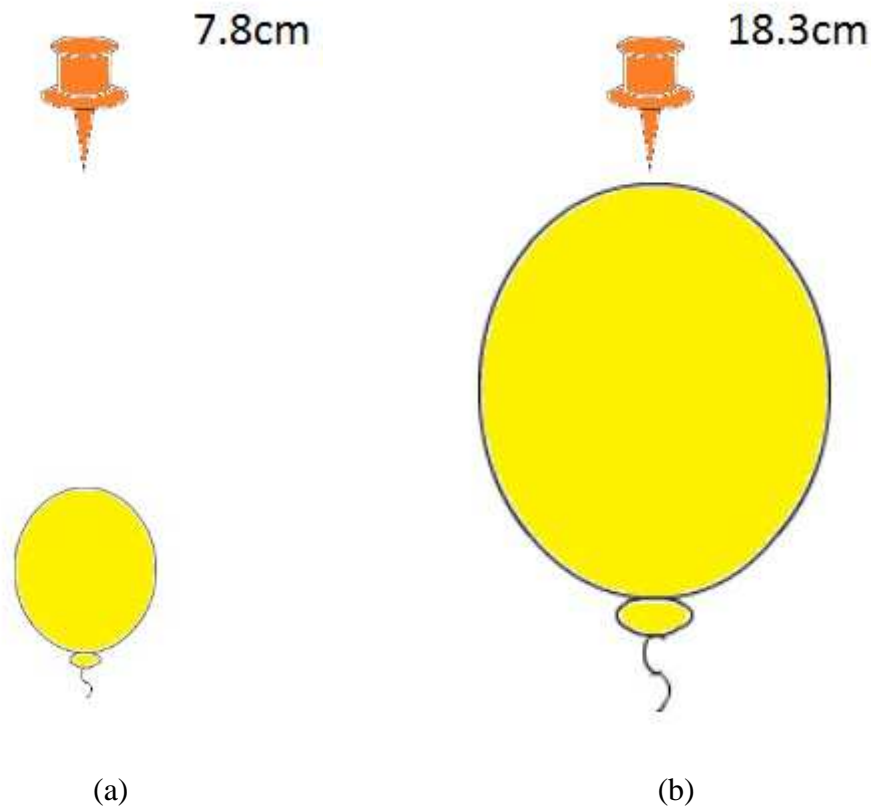
Figure 4. Game "Balloon Piercing".

Listing below depicts part of the string resource file "string.xml".

```
…
    <!-- Game Balloon piercing-->
    <!-- Key preference items and the corresponding default value -->
    <string
 name="balloon_piercing_color_key">balloon_piercing_color_key</string>
    <string
 name="balloon_piercing_speed_key">balloon_piercing_speed_key</string>

    <!-- String array format -->
    <string-array name="balloon_piercing_color">
        <item>Red</item>
        <item>Green</item>
        <item>Blue</item>
        <item>Yellow</item>
```

```
    </string-array>

    <string-array name="balloon_piercing_speed">
      <item>Slow</item>
      <item>Normal</item>
      <item>Fast</item>
    </string-array>
…
```

Listing below depicts the preference screen layout resource file "game_prefs.xml".

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >

    <PreferenceCategory
      android:title="Color setting">
      <ListPreference
        android:key="@string/balloon_piercing_color_key"
        android:title="Balloon Color"
        android:summary="Color of the balloon"
        android:entries="@array/balloon_piercing_color"
        android:entryValues="@array/balloon_piercing_color"
        android:defaultValue="Yellow" />
    </PreferenceCategory>

    <PreferenceCategory
      android:title="Game setting">
      <ListPreference
        android:key="@string/balloon_piercing_speed_key"
        android:title="Game Speed"
        android:summary="Growing speed of the balloon"
        android:entries="@array/balloon_piercing_speed"
        android:entryValues="@array/balloon_piercing_speed"
        android:defaultValue="Slow" />
    </PreferenceCategory>

</PreferenceScreen>
```

Illustrate with diagrams about what you would see when the preference screen defined in the layout resource file "game_prefs.xml" is shown. **Sketches** with diagram if preference component such as CheckBoxPreference, EditTextPreference, ListPreference, or MultiSelectListPreference is selected.

[6]

(b) Providing sound effect and background music or different game modes and themes are different ways that can arouse interest. **Suggest** and **describe** one game option which can be applied in the "Balloon Piercing" game for increasing the game playability.
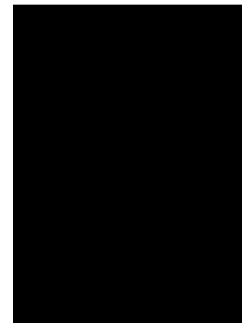
[3]

(a) Start of animation.	(b) After 2.5s.	(c) After 5s.

Illustration of animation with diagrams.

Question 4 [9 marks]

(a)
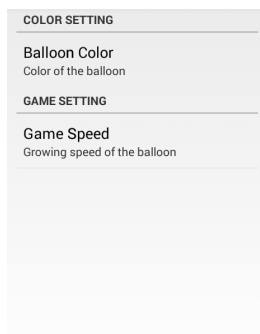
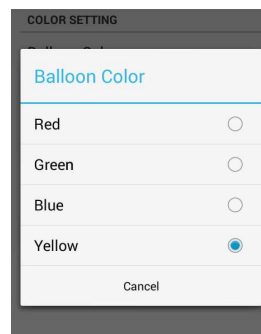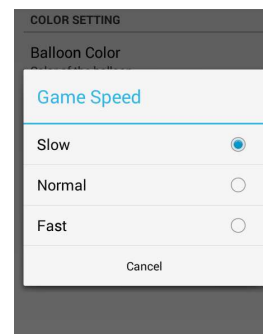|		
(a) Preference screen.	(b) Balloon Color.	(c) Game Speed.

Figure. Game preferences.

(b) **Progressive speed up of the balloon**
  - **The growing speed of the balloon will be increased for certain time period.**

Question 5 [15 marks]

> **Handle user input**
>   - **Evaluate the moving direction of the ball (left/right)**
> **Update game state**
>   - **Update the position of the ball**
>   - **Update the position of the paddle**
>   - **Update the time elapsed**
> **If the ball collided with the paddle OR**
> **If move out of the paddle**
>   - **Change the moving direction of the ball in y-axis**
> **If the ball collided with boundary of the screen**
>   - **Draw "Game Over!" on the screen**
>   - **Exit game loop (or End game)**
> **Generate paddle randomly**
> **Refresh the screen (or Update display)**
> **Sleep**

[15 marks, 1 mark for each operation]

Question 3 [6 marks]

The preference framework in Android provides a flexible and convenient mechanism for manipulating shared preferences.

Listing below depicts the string resource file "string.xml".

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

    <string name="app_name">PreferenceProject</string>
    <string name="action_settings">Settings</string>

    <string name="display_setting">Display Setting</string>
    <string name="display_title">Display</string>
    <string name="display_summary">Type of information to be displayed in the
game</string>

    <string name="sound_setting">Sound Setting</string>
    <string name="sound_title">Sound</string>
    <string name="sound_summary">Whether to play sound in the game</string>

    <!-- Key preference items and the corresponding default value -->
    <string name="display_key">display</string>
    <string-array name="default_array">
        <item>Nick name</item>
    </string-array>
    <string name="sound_key">sound</string>
    <bool name="sound_default">true</bool>

    <!-- display in string array format -->
    <string-array name="display_selection">
        <item>Nick name</item>
        <item>Equipment</item>
        <item>Level</item>
    </string-array>

</resources>
```

Listing below depicts the preference screen layout resource file "game_prefs.xml".

```xml
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >

  <PreferenceCategory
     android:title="@string/display_setting">

    <MultiSelectListPreference
       android:key="@string/display_key"
       android:title="@string/display_title"
       android:summary="@string/display_summary"
       android:entries="@array/display_selection"
       android:entryValues="@array/display_selection"
       android:defaultValue="@array/default_array" />

  </PreferenceCategory>

  <PreferenceCategory
     android:title="@string/sound_setting">

    <CheckBoxPreference
       android:key="@string/sound_key"
       android:title="@string/sound_title"
       android:summary="@string/sound_summary"
       android:defaultValue="@bool/sound_default" />

  </PreferenceCategory>

</PreferenceScreen>
```
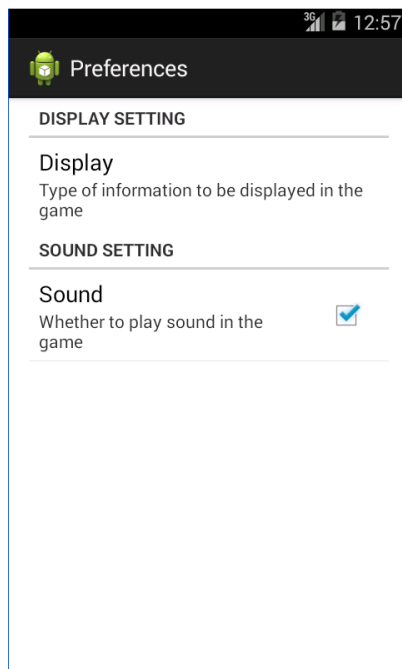
Illustrates with diagram about what you would see when the preference screen defined in the layout resource file "game_prefs.xml" is shown. It is also required to **sketch** with diagram if preference component such as CheckBoxPreference, EditTextPreference, ListPreference, or MultiSelectListPreference is selected.
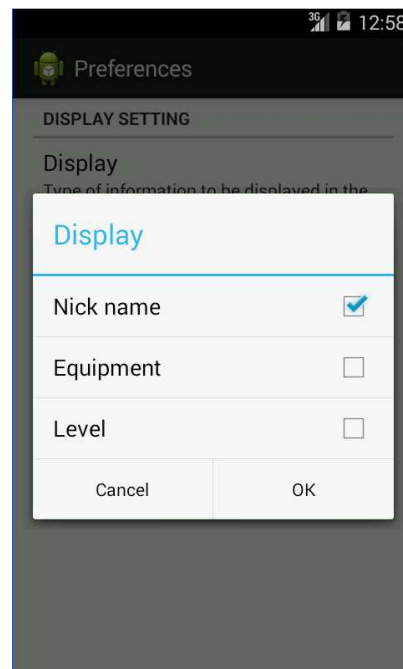
[6]

Question 3 [6 marks]

(a) Preference screen.   (b) Font color selection.

Figure. Game preferences.

Question 4 [12 marks]

- **The Button is aligned to the left side and placed in the vertical center of its parent container.**
- **The animation duration is 10000ms (or 10s).**
- **No repeated rounds are animated.**
- **The "AnimationSet" specifies a composite of four translation animation in sequential order.**
- **The first translation animation slides from its initial position to center of the top boundary of the container in 4s.**
- **It then moves to the center of the right boundary in 3s.**
- **The third movement is completed in 2s from the right side to the center position of the bottom edge.**
- **Finally, the button moves back to its initial position in 1s.**

(a)   "Cross the danger" is a game in which player needs to control the pedestrian to cross the road as fast as he can. After the game started, motorcycles are driving on the road. The player can change the moving direction of the pedestrian in order to cross the road by tapping on the screen. Once the pedestrian crosses the road successfully, the game ends and the total time elapsed will be sketched on the screen. When the pedestrian is crashed by motorcycle, the game ends and "Game Over!" message will be shown to the player.

Figure 4. Game "Cross the danger".

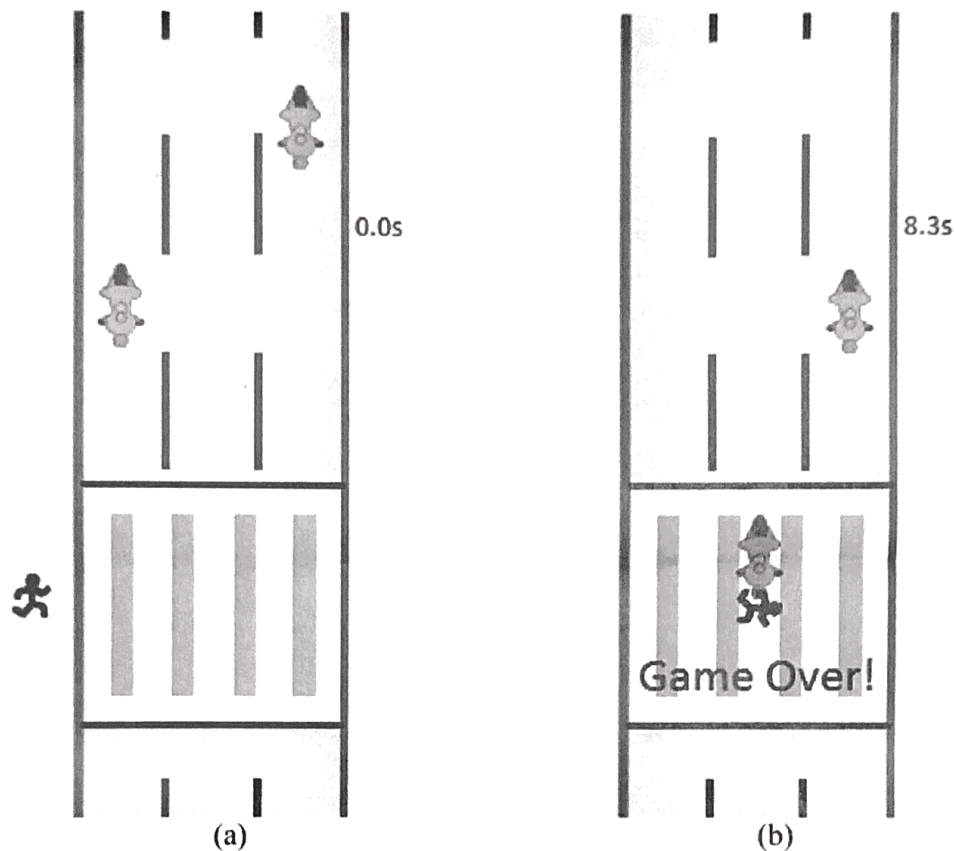Listing below depicts part of the string resource file "string.xml".

```
...
    <!-- Game Cross the danger-->
    <!-- Key preference items and the corresponding default value -->
    <string name="speed_of_obstacles_key">speed_of_obstacles_key</string>
    <string name="lines_of_road_key">lines_of_road_key</string>

    <!-- String array format -->
    <string-array name="speed_of_obstacles">
       <item>Slow</item>
       <item>Normal</item>
       <item>Fast</item>
    </string-array>
```

```xml
<string-array name="lines_of_road">
  <item>3</item>
  <item>4</item>
  <item>5</item>
</string-array>
...
```

Listing below depicts the preference screen layout resource file "game_prefs.xml".

```xml
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >

    <PreferenceCategory
        android:title="Game setting">
        <ListPreference
            android:key="@string/speed_of_obstacles_key"
            android:title="Speed of Obstacles"
            android:summary="Configure the speed of moving obstacles"
            android:entries="@array/speed_of_obstacles"
            android:entryValues="@array/speed_of_obstacles"
            android:defaultValue="Slow" />

        <ListPreference
            android:key="@string/lines_of_road_key"
            android:title="Lines of Road"
            android:summary="Determine the number of lines of road"
            android:entries="@array/lines_of_road"
            android:entryValues="@array/lines_of_road"
            android:defaultValue="3" />
    </PreferenceCategory>

</PreferenceScreen>
```

Illustrate with diagrams about what you would see when the preference screen defined in the layout resource file "game_prefs.xml" is shown. Sketches with diagram if preference component such as CheckBoxPreference, EditTextPreference, ListPreference, or MultiSelectListPreference is selected.

[6]

(b) Providing sound effect and background music or different game modes and themes are different ways that can arouse interest. Suggest and describe one game option which can be applied in the "Cross the danger" game for increasing the game playability.

[4]

Question 6 [15 marks]

The preference framework in Android provides a flexible and convenient mechanism for manipulating shared preferences.

Listing below depicts part of the string resource file "string.xml"

```
…
    <!--  Key preference items and the corresponding default value -->
    <string name="text_color_key">text_color</string>
    <string name="text_color_default">White</string>

    <string name="text_size_key">text_size</string>
    <string name="text_size_default">20</string>

    <string-array name="color_names">
       <item>White</item>
       <item>Red</item>
       <item>Yellow</item>
       <item>Green</item>
    </string-array>
…
```
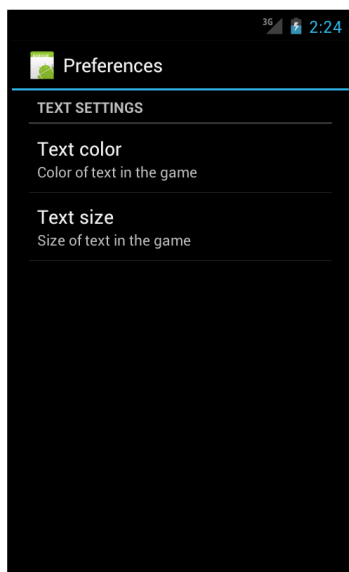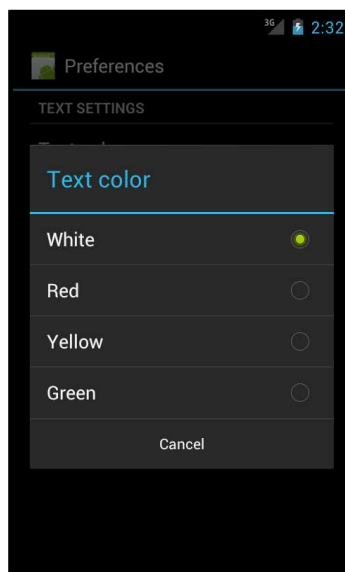
Listing below depicts part of the preference screen layout resource file "prefs.xml"

```
<?xml version="1.0" encoding="utf-8"?>
<PreferenceScreen xmlns:android="http://schemas.android.com/apk/res/android" >
…
</PreferenceScreen>
```
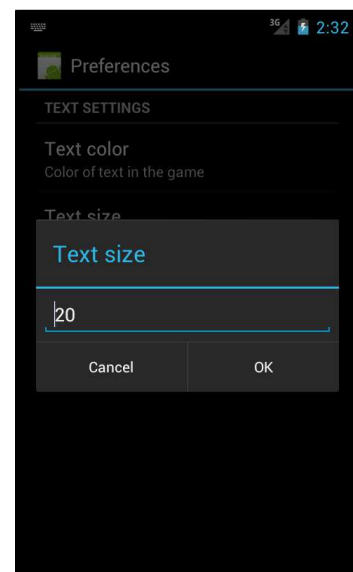
(a)   **Complete** the preference screen layout resource file "prefs.xml" with respect to the following figures.

(a) Preference screen.   (b) Font color selection.   (c) Font size configuration.
Figure 6. Game preferences.

In Figure 6 (a), two preference items are grouped into category "TEXT SETTINGS". Under this category, "ListPreference" and "EditTextPreference" are employed to change the font color and font size respectively.

[8]

(b) **Specify** the interface class that used in the preference framework for representing the shared preferences.

[2]

(c) "A To Z" is a game of touching from 'A' to 'Z' as fast as you can. **Suggest** and **describe** two game options for the "A To Z" game to increase the game playability and arouse interest.

[5]

[End of Question 6]

Question 6   **[15 marks]**

(a)
```xml
<PreferenceCategory
   android:title="Text settings">

   <ListPreference
      android:key="@string/text_color_key"
      android:title="Text color"
      android:summary="Color of text in the game"
      android:entries="@array/color_names"
      android:entryValues="@array/color_names"
      android:defaultValue="@string/text_color_default" />

   <EditTextPreference
      android:key="@string/text_size_key"
      android:title="Text size"
      android:summary="Size of text in the game"
      android:inputType="number"
      android:defaultValue="@string/text_size_default" />

</PreferenceCategory>
```

(b)   **Shared preferences are represented by and manipulated using the "android.content.SharedPreferences" interface.**

(c)   **Difficulty level**
   -   **Different difficulty level can selected by the player, e.g., "A To I", "A To P", or "A To Z".**
   Sound effect
   -   **Sound effect of correct/incorrect answer can be enabled/disabled.**

[End of Question 6]