CS6301 Human-Computer Interactions

Spring 2022

Assignments 6 – Write a “Pop the Balloons” Game

Your assignment is to write a relatively simple game. The game play screen shows balloons rising from the bottom and going off the top, and the object of the game is to pop as many as you can in a given time period, say 60 seconds. Here is how it works.

1. Bring up a screen that describes game play to the user. For example, it may say, “Pop only the RED Circles,” or “Pop only PURPLE squares.” Your game will have only circles and squares as shapes, and use only the colors Red, Orange, Yellow, Green, Blue, Purple, and White. These will be shown on a dark background. They will be different randomly-chosen sizes, ranging from 32dp to 64dp, but you can experiment with this. For circles, this is the diameter. For squares, this is the length of a side. The shape and color are also chosen randomly when the program starts.
2. When the user touches the OK button, bring up a screen that shows a random number of circles and squares in different sizes and colors, all at the bottom of the screen. The number should be between 6 and 12, although you can experiment with this. The program can display a mix of squares and circles, but they must not overlap. Again, choose the mix randomly.
3. The playing field is from the bottom of the screen to 2/3 of the screen height. You can have other things above that, such as a running score, timer, etc. Outline the playing field with colored lines.
4. The balloons start at the bottom of the screen and move up at different speeds. When the top of a balloon reaches the top of the playing field, the balloon disappears.
5. If you touch a correct shape (pop the balloon), you receive a point. If you pop the wrong shape or color, you lose a point. If a balloon you were supposed to pop goes off the top of the screen, you lose a point. Any other press is ignored. A new balloon starts at the bottom when one is popped by the user or disappears off the top.
6. The game ends after 1 minute. However, for every 10 balloons popped, the player gets 10 additional seconds.
7. Score is the total number of balloons popped minus the number of wrong pops. At the end of the game, show the total score and the number of balloons that should have been popped but that went off the screen before the player got them. If this is a high score, the user can press a button to enter it into the high scores list.

You will need to use a timer to drive this. On every timer tick, recompute the positions of the balloons and redisplay the score.

Use the High Scores screen you wrote for assignment 5 to track high scores for this game. You’ll have to modify your main Activity a little. There will be four Activities in the game:

1. The starting activity with the Start button and instructions.
2. The game play screen
3. The high scores display screen from the previous assignment
4. The high scores entry screen from the previous assignment, but you cannot enter the score manually; it will come from the game.

**Do not** use a surface holder; just create a custom View and override onDraw.

Remember that there is timer code you can use on my Web site, under Programming Tips, and possibly other Android code. Code from the slides posted on eLearning is also free use. **You may not use any other code. This is an individual assignment.**

**To hand in:**

Your complete Android Studio project as a Zip file. The file must be named <NetID>Asg6.zip. You will be required to show your working program running on your device to the TA, as well.

|  |  |
| --- | --- |
| Project Grading | |
| Meets the above requirements | 70 |
| Clean, object-oriented code. There must be at least one base class you define and two subclasses of it for the shapes. | 20 |
| **Program comments and naming conventions** | 10 |

**Additional grading criteria:**

Using a surface holder. -30

Game screen does not update correctly. This could be various problems from balloons not floating up to the top, popping balloons that have not been touched, balloons overlapping other balloons all balloons moving at the same speed, etc. -10 to -20

Scoring not integrated with the game. -10

Using images instead of drawn shapes for the balloons. -30

Not following good object-oriented design. -10 to -20

**No header comments in even one module: -5**