

# 2023 AP Daily: Practice Sessions



## AP Computer Science A

### Session 2 – FRQ (Question 1: Methods & Control Structures)

1. This question involves simulation of the play and scoring of a single-player video game. In the game, a player attempts to complete three levels. A level in the game is represented by the `Level` class.

```
public class Level
{
    // Returns true if the player reached the goal on this level and returns
    false otherwise
    public boolean goalReached()
    { /* implementation not shown */ }

    // Returns the number of points (a positive integer) recorded for this level

    public int getPoints()

    { /* implementation not shown */ }

    // There may be instance variables, constructors, and methods that are
    not shown.
}
```

Play of the game is represented by the `Game` class. You will write two methods of the `Game` class.

```
public class Game
{
    private Level levelOne;
    private Level levelTwo;
    private Level levelThree;

    // Postcondition: All instance variables have been initialized.
    public Game()
    { /* implementation not shown */ }

    // Returns true if this game is a bonus game and returns false
otherwise
    public boolean isBonus()
    { /* implementation not shown */ }

    // Simulates the play of this Game (consisting of three levels) and
updates all relevant data
    public void play()
    { /* implementation not shown */ }

    //Returns the score earned in the most recently played game, as described in
part (a) public int getScore()
    { /* to be implemented in part (a) */ }

    // Simulates the play of num games and returns the highest score earned, as
described // in part (b)
    // Precondition: num > 0
    public int playManyTimes(int num)
    { /* to be implemented in part (b) */ }

    // There may be instance variables, constructors, and methods that are not
shown.
}
```

- a. Write the `getScore` method, which returns the score for the most recently played game. Each game consists of three levels. The score for the game is computed using the following helper methods.
- The `isBonus` method of the `Game` class returns `true` if this is a bonus game and returns `false` otherwise.
  - The `goalReached` method of the `Level` class returns `true` if the goal has been reached on a particular level and returns `false` otherwise.
  - The `getPoints` method of the `Level` class returns the number of points recorded on a particular level. Whether or not recorded points are earned (included in the game score) depends on the rules of the game, which follow.

The score for the game is computed according to the following rules.

- Level one points are earned only if the level one goal is reached. Level two points are earned only if both the level one and level two goals are reached. Level three points are earned only if the goals of all three levels are reached.
- The score for the game is the sum of the points earned for levels one, two, and three.
- If the game is a bonus game, the score for the game is tripled.

The following table shows some examples of game score calculations.

	Level One Results	Level One Results	Level One Results	isBonus Return Value	Score Calculation
goalReached <b>Return Value:</b>  getPoints <b>Return Value:</b>	true	true	true	true	$(200 + 100 + 500) \times 3 = 2,400$ The recorded points for levels one, two, and three are earned because the goals were reached in all three levels. The earned points are multiplied by 3 because <code>isBonus</code> returns true.
goalReached <b>Return Value:</b>  getPoints <b>Return Value:</b>	true	true	false	false	$200 + 100 = 300$ The recorded points for level one and level two are earned because the goal was reached in levels one and two. The recorded points for level three were not earned because the goal was not reached in level 3.
goalReached <b>Return Value:</b>  getPoints <b>Return Value:</b>	true	false	true	true	$200 \times 3 = 600$ The recorded points for only level one are earned because the goal was not reached in level two. The earned points are multiplied by 3 because <code>isBonus</code> returns true.
goalReached <b>Return Value:</b>  getPoints <b>Return Value:</b>	false	true	true	false	0 Because the goal in level one was not reached, no points are earned for any level.

Complete the `getScore` method.

```
// Returns the score earned in the most recently played game, as described in part (a)
public int getScore()
{
    b. Write the playManyTimes method, which simulates the play of num games
       and returns the highest game score earned. For example, if the four plays of the
       game that are simulated as a result of the method call playManyTimes(4)
       earn scores of 75, 50, 90, and 20, then the method should return 90.
```

Play of the game is simulated by calling the helper method `play`. Note that if `play` is called only one time followed by multiple consecutive calls to `getScore`, each call to `getScore` will return the score earned in the single simulated play of the game.

Complete the `playManyTimes` method. Assume that `getScore` works as intended, regardless of what you wrote in part (a). You must call `play` and `getScore` appropriately in order to receive full credit.

```
// Simulates the play of num games and returns the highest score earned, as
// described in part (b)
// Precondition: num > 0
public int playManyTimes(int num)
{
```

Class information for this question

```
public class Level
public Boolean goalReached()
public int getPoints()

public class Game
private Level levelOne
private Level levelTwo
private Level levelThree

public Game()
public Boolean isBonus()
public void play()
public int getScore()
public int playManyTimes (int num)
```