

Assignment 1 Solution

1.

```
public class Person {
    private String _name;
    private Phone _phone = new Phone();

    public String getName() {
        return _name;
    }

    public Person(String aName, Phone aPhone) {
        _name = aName;
        aPhone = _phone;
    } // Included only for completeness

    public Phone getPhone() {
        return _phone;
    }
}

public class Phone {
    private String _officeAreaCode;
    private String _officeNumber;

    public String getOfficeAreaCode() {
        return _officeAreaCode;
    }

    public void setOfficeAreaCode(String newAreaCode) {
        _officeAreaCode = newAreaCode;
    }

    public String getOfficeNumber() {
        return _officeNumber;
    }

    public void setOfficeNumber(String newNumber) {
        _officeNumber = newNumber;
    }
}
```

2. *GameTest* class is not included as it does not change, per the problem directions.

```
public class Game {
    private int FIRST_SPACE = 0;
    private int BOARD_SIZE = 9;
    private int ROW_SIZE = 3;

    private char EMPTY_SPACE = '-';
    private char NO_WINNER = '-';
    private int NO_MOVE = -1;

    public StringBuffer board;

    public Game(String s) {board = new StringBuffer(s);}

    public Game(StringBuffer s, int position, char player) {
        board = new StringBuffer();
        board.append(s);
        board.setCharAt(position, player);
    }

    public int move(char player) {
        for (int i = FIRST_SPACE; i < BOARD_SIZE; i++) {
            if (board.charAt(i) == EMPTY_SPACE) {
                //Create a temporary game to determine
                //if a given move is a winning move
                Game game = play(i, player);
                if (game.winner() == player)
                    return i;
            }
        }

        for (int i = FIRST_SPACE; i < BOARD_SIZE; i++) {
            if (board.charAt(i) == EMPTY_SPACE)
                return i;
        }

        return NO_MOVE;
    }

    public Game play(int i, char player) {
        return new Game(this.board, i, player);
    }

    public char winner() {
```

```

//DEBT: Technically still magic numbers below (1 & 2). It will be taken
// care of when we remove the duplication here
if (board.charAt(FIRST_SPACE) != EMPTY_SPACE
    && board.charAt(FIRST_SPACE) == board.charAt(FIRST_SPACE + 1)
    && board.charAt(FIRST_SPACE + 1) == board.charAt(FIRST_SPACE + 2))
    return board.charAt(FIRST_SPACE);
if (board.charAt(FIRST_SPACE + ROW_SIZE) != EMPTY_SPACE
    && board.charAt(FIRST_SPACE + ROW_SIZE)
        == board.charAt(FIRST_SPACE + ROW_SIZE + 1)
    && board.charAt(FIRST_SPACE + ROW_SIZE + 1)
        == board.charAt(FIRST_SPACE + ROW_SIZE + 2))
    return board.charAt(FIRST_SPACE + ROW_SIZE);
if (board.charAt(FIRST_SPACE + (2 * ROW_SIZE)) != EMPTY_SPACE
    && board.charAt(FIRST_SPACE + (2 * ROW_SIZE))
        == board.charAt(FIRST_SPACE + (2 * ROW_SIZE) + 1)
    && board.charAt(FIRST_SPACE + (2 * ROW_SIZE) + 1)
        == board.charAt(FIRST_SPACE + (2 * ROW_SIZE) + 2))
    return board.charAt(FIRST_SPACE + (2 * ROW_SIZE));
return NO_WINNER;
}
}

```

3.

a.

```

public boolean isLeapMonth() {
    if (!isFebruary()) return false;
    if (!yearDivisibleBy(4)) return false;
    if (yearDivisibleBy(100) && !yearDivisibleBy(400)) return false;
    return true;
}

private boolean isFebruary() {
    return (_currentMonth == 2);
}

private boolean yearDivisibleBy(int divisor) {
    return (_currentYear.mod(divisor) == 0);
}

```

b.

```

public void handleEnds() {
    if (elementIsNegative() || elementIsAboveMax() ||
        elementIsLastElement()) {
        // do something
    }
}

```

```
private boolean elementIsNegative() {  
    return (elementNo < 0);  
}
```

```
private boolean elementIsAboveMax() {  
    return (elementNo > maxElement);  
}
```

```
private boolean elementIsLastElement() {  
    return (elementNo == lastElement);  
}
```

c.

```
public isPassing() {  
    if (_currentlyEnrolled && _hasProfessorsRespect) return true;  
    if (_graduateStudent) return true;  
    if (_childOfRichAlumnus) return true;  
    return false;  
}
```