Name _____ Period_____

1. The `Assignment` class below creates and manages assignments for a gradebook,

Assignment.java

```
public class Assignment {
    private String name;
    private int totalPoints;
    private int dueDate;
    public Assignment(String n, int tp, int dd){
        name = n;
        totalPoints = tp;
        dueDate = dd;
    }

    public String getFormattedDueDate(){
        int temp = dueDate;
        int year = temp%100;
        temp /= 100;
        int month = temp%100;
        temp /= 100;
        int day = temp;
        return month + "/" + day + "/" + year;
    }

    public int getTotalPoints(){
        return totalPoints;
    }

    public String getAssignment(){
        return name;
    }

    public String toString(){
        String result = name + "is worth " + totalPoints + " points ";
        result += " and is due " + " on " + getFormattedDueDate();
        return result;
    }
}
```

A portion of the Gradebook class is shown below. You will write code to complete the remainder of this class.

Gradebook.java

```
import java.util.Scanner;

public class Gradebook{

    public static void main(String args[]){
        //Creates a gradebook with 10 assignments
        Assignment gradebook[] = new Assignment[5];

        //Prompts the user for the assignment information
        Scanner input = new Scanner(System.in);
        System.out.println("What is the name of the assignment?");
        String name = input.nextLine();
        System.out.println("How many points is the assignment out of?");
        int points = input.nextInt();
        System.out.println("What is the due date (mm/dd/yy)?");
        String dueDate = input.next();
    }

}
```

(a) A scanner is used to get the input required for each assignment. In the space below, write code that could be used to create an Assignment using the input provided.

```
String dueDateArr[] = dueDate.split("/");
int dueDateFormatted = Integer.parseInt(dueDateArr[0] + dueDateArr[1] +
dueDateArr[2]);
Assignment a = new Assignment(name, points, dueDateFormatted);
```

/4

Score _____/17

(b) Below is a list of assignments that have been stored in the array gradebook,

Index	Name	Total Points	Due Date
0	Exam 1	18	90123
1	Exam 2	12	90823
2	Exam 3	17	91523
3	Lab 2	20	91523
4	Ticket Out the Door 4	5	90223

In the space below, indicate how you could find the assignment worth the most points. Once you have located the assignment, print its corresponding information. For example,

Lab 2 is worth 20 points and is due on 15/9/23

```
int high = gradebook[0].getTotalPoints();
int highIndex = 0;
for(int i = 1; i < gradebook.length;i++){
    if(gradebook[i].getTotalPoints() > high){
        high = gradebook[i].getTotalPoints();
        highIndex = i;
    }
}
System.out.println(gradebook[highIndex]);
```

/4

Score _____/17

(c) A gradebook can be visualized as a series of parallel arrays as follows. Where the values in each array represent the total points earned on the corresponding assignment.

	Exam 1	**Exam 2**	**Exam 3**	**Lab 2**	**Ticket Out the Door 4**
Bart	15	9	14	20	2
Homer	14	11	12	18	4
Wilma	12	12	9	17	5
Averages	4.33	3.33	3.0	3.33	3.0

```
int Bart[] = {15, 9, 14, 20, 2};
int Homer[] = {14, 11, 12, 28, 4};
int Wilma[] = {12, 12, 9, 17, 5};
```

The avgGrades array stores the average grade for each assignment as a percentage. For example,

```
int avgGrades[] = {75, 88, 68, 108, 73}
```

In space below, write code that could be used to calculate the class average for each assignment and store the corresponding average in the avgGrades array.

```
        double totalEarned = 0;
        double totalPoints = 0;

        for(int i = 0; i < avgGrades.length;i++){
            avgGrades[i] = (int)(((Bart[i] + Homer[i] +
Wilma[i])/3.0)/gradebook[i].getTotalPoints()*100);
        }
```

/4

Score _____/17

A report card needs to be generated for each student. Consider the report card for Bart,

Exam 1: 15/18
Exam 2: 9/12
Exam 3: 14/17
Lab 2: 20/20
Ticket Out the Door 4: 2/5
Final Grade: 83

In the space below, write code that could be used to create the output shown for Bart using the `Assignment` class.

```
        double totalEarned = 0;
        double totalPoints = 0;

        for(int i = 0; i < Bart.length;i++){

            System.out.println(gradebook[i].getAssignment() + ":\t" + Bart[i]
+ "/" + gradebook[i].getTotalPoints());

            totalEarned += Bart[i];
            totalPoints += gradebook[i].getTotalPoints();
        }
         System.out.println("Final Grade: " +
(int)((double)totalEarned/(double)totalPoints * 100));
```

/5

Score _____/17

Score _____/17