Fall 2001 CS2340
Midterm Exam
Name: ___________________________
StudentNumber: ________________

Different instructors and TAs are going to grade different problems. **PUT ALL OF YOUR WORK FOR A PROBLEM ON THE SAME SHEET!** You will rip your sheets apart and put each problem in a separate folder when handing them in.

1. (25) String to Factorial

Write a method `stringToFactorial`: that takes as input a string of positive integers separated by spaces (e.g. ‘12 13 14’) and returns a collection of the factorials of each of the numbers in the string (e.g., an `OrderedCollection(479001600 6227020800 87178291200)`). Hints: Strings understand `findTokens` which inputs a string of delimiters (e.g., `' ' `to specify a space) and returns a collection of strings that were separated by any of these delimiters (e.g., ‘12’ ‘13’ ‘13’). Hint2: Integers understand `factorial`. Hint3: Strings understand `asNumber`.

Take off only 1-2 pts. for syntax errors
2. (10) Changed/Update mechanism

In the third Clock user interface, a Clock instance forms the model and a ClockText instance forms a view. Describe the sequence of messages that occur when a second of time passes by, such that the Clock is updated and the updated time is displayed by the ClockText. Hints: Think about self changed:, nextSecond, update:, and dependents.

For a - (ClockText is dependent on Clock)

1 - Clock gets nextSecond

Clock updates time.

2 - Clock announces self changed:

3 - ClockText gets updated:

4 - ClockText asks clock (model) for time.

5 - ClockText updates the display.
3. (15) OO Theory

a. What is a class? What is an instance?

   Class - blueprint for instances. Specifies behavior & data. (Similar)

   Instance - an object formed from a class.

b. What is the difference between a class and instance method?

   Class methods are only understood by class object - S
   
   Instance methods understood by instances - Z

c. What is the difference between a class and instance variable?

   Class variable shared among all instances.

   Instance variable unique to instance.
4. (25) What's wrong with this picture?

a. The below class diagram for a course registration system exemplifies a typical object-oriented design flaw. What is it?

```
[Diagram of class diagram]
```

b. You're reviewing a set of CRC Cards for a fellow object-oriented designer who's working on an inventory control system. You see the classes: Item, WarehouseLocation, LinkedListOfShelves, Supplier, Shipper. What do you immediately point out as a flaw in his cards?

```
No data structures specified at CCA time. Spit.
```

c. You're working with a new designer with whom you're brainstorming candidate class names. He says, "Let's include Date and Time cards, because they're already in Squeak, so we'll already have those implemented." What mistake is being made?

```
Either okay:
- Don't worry about language at CCA time.
- Don't worry about implementation at CCA time. Don't worry about domain only.
```
d. There once was a city where the payroll system was designed in-house as an object-oriented system. At first, they just had hourly employees. Later, they had crossing guards, who sometimes were volunteers. Later they added salaried employees who received a weekly pay. They never redesigned their object system, and just kept adding on to it. What flaws (there are at least two) do you see in the below object system?

![Object diagram]

- Volunteer crossing guards have an hourly wage.
- Salaried employee has both a weekly salary and an hourly wage.
5. (25) UML Class Diagram

You are designing a strategy game around the marketing and signing of sports figures.

- There are athletes, who know their agents and their home team. They have a name, a salary, and a salary that they really want. They can negotiate with a team.
- There are agents, who know their athletes and the league. They have a name and a set of contracts with each athlete which describes the percentage that they get from the athlete's contract. They also know how to negotiate and how to make secret alliances with a team or a coach.
- The league knows all the teams, and can add or remove teams. It can also impose or remove rules on how teams behave.
- Teams know their athletes, and they can negotiate and make secret deals with other teams.
- Coaches know their teams, and they can also have agents. Coaches have a name and a salary, too.

Draw the UML class diagram for this set of classes. Hint: You are not limited to the classes identified above! You may find that you need a couple more classes, particularly a superclass or two, that helps to organize this set.