PLEASE HAND IN

PLEASE HAND IN

UNIVERSITY OF TORONTO
Faculty of Arts and Science

St. George Campus

DECEMBER 2006 EXAMINATIONS

CSC 207H1F
Duration — 3 hours

Examination Aids: Any handwritten or printed materials. No electronic aids.

Student Number: |_|_|_|_|_|_|_|_|_|

Last (Family) Name(s): _____

First (Given) Name(s): _____

---

*Do **not** turn this page until you have received the signal to start.*
(In the meantime, please fill out the identification section above, and read the instructions below *carefully*.)

This final examination consists of 8 questions on 13 pages (including this one). *When you receive the signal to start, please make sure that your copy of the examination is complete.*
There are two mostly-blank pages at the end of the exam that you may use if you run out of space.
Comments are not necessary unless specifically indicated in the question, you may abbreviate `System.out.println` as `S.o.p`, and you may assume that any necessary imports have been done.

Good Luck!

MARKING GUIDE

1: _____/10
2: _____/ 7
3: _____/12
4: _____/ 5
5: _____/ 6
6: _____/10
7: _____/ 7
8: _____/ 8

TOTAL: _____/65

Question 1. [10 MARKS]

All product codes at Jim's Jellybean Company consist of 3 uppercase letters followed by 13 to 15 digits.

Part (a) [3 MARKS]

Write a regular expression that will only match lines starting with a valid product code. Your expression should be written so that group(0) contains the entire line, not just the product code at the beginning.

Part (b) [3 MARKS]

Write a regular expression that will match lines consisting of 2 or more product codes separated by commas and optional white space. The whitespace may occur before or after any of the tokens on the line but may not occur within the product codes themselves. The product codes on the line must all be valid but they don't have to be the same as each other.

Part (c) [3 MARKS]

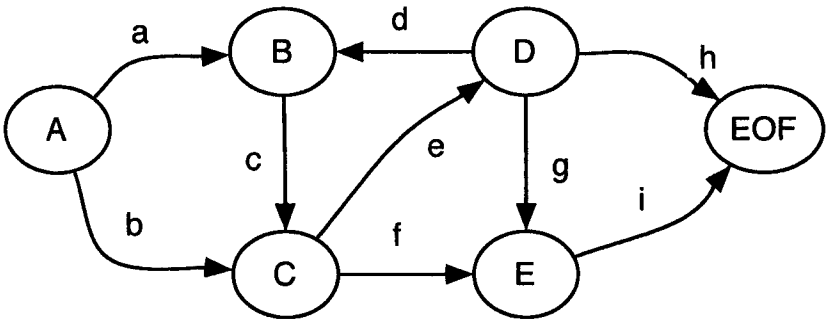
Write a regular expression that will match lines consisting of 2 or more occurrences of the same valid product code separated by whitespace with no commas. Whitespace can optionally occur before the first token and after the last.

Part (d) [1 MARK]

Consider the following regular expression `^[:;][-^]\)$`. List all the strings that exactly match this expression or, if there are more than ten such strings, describe them.

Question 2. [7 MARKS]

This question is about regular finite-state machines. In particular, it is about the finite-state machine shown in this diagram:



In this machine, the input is a sequence of characters. The start state is the one labelled A, and the "success" state is the one labelled EOF. Any transactions not shown on the diagram go to error.

Part (a) [2 MARKS]

There is more than one input that successfully reaches the success state. Give the two shortest inputs that succeed.

Part (b) [2 MARKS]

How long is the longest successful input? Why?

Part (c) [3 MARKS]

Below is the beginning of a handler method for state D. Complete the handler, assuming that every state in the diagram has a handler function, and that a function `error(message)` can be called in case of erroneous input to display a message and terminate execution.

```
def handleD(ch):
    '''Return handler function for next state when starting from state D
    with input character ch.
    '''
```

Question 3. [12 MARKS]

Part (a) [10 MARKS]

For the purposes of this question, a Java class `A` *depends directly on* another Java class `B` if `A` contains a field (static variable or instance variable) of type `B`, a method parameter of type `B` or a method returning type `B`. Class `A` *depends on* class `C` if `A` depends directly on `C` or if `A` depends directly on `B` and `B` depends on `C`.

Write a Java method `getDep(...)` that given a Java class `A` will return a set of all the other Java classes that `A` depends on. The signature of the method is up to you. In part (b) you will demonstrate how to use your method, and that will relate to the signature you design.

Note that `getDep()` must work on classes, not on source files.

Part (b) [2 MARKS]

Complete the main method to use getDep to print out the classes that `java.util.HashMap` depends on.

```
//all necessary imports present but not shown
public class Reflect {
    public static void main(String[] args) throws Exception {
        String name = "java.util.HashMap";
        // do any set-up you need here (if any at all)



        // call your method here


        // print out the Contents of your set here




    }
    // the rest of this class would include your method from part (a)
```

Question 4. [5 MARKS]

Imagine that you are a 207 TA and you are writing a python program `checkout_one.py` that you will use to do a CVS checkout on a repository or partial repository for a single student. By default your script should check out the entire student repository, but you also want to be able to specify the subdirectory for a single exercise. The script should default to checking out the current version but you also need a way to specify a time and date for the checkout so that you can use the script to checkout files as of a past assignment deadline. The repository root is determined based on the student login and the module name is the same for every student so these don't need to be specified each time.

Below are five possible interface designs. Each is demonstrated by an example showing a check out of the `e1` directory for the login `c4studly` as of `"2006-10-31 10:05:00"`, followed by an example of a checkout of the complete current module for `g6keen`.

Circle the best design. For each of the other designs, comment on the problems or disadvantages that make you think it is worse than your choice.

1.
```
checkout_one.py -e 1 -d "2006-10-31 10:05:00" -s c4studly
checkout_one.py -s g6keen
```

2.
```
checkout_one.py c4studly e1 "2006-10-31 10:05:00"
checkout_one.py g6keen
```

3.
```
checkout_one.py
 then program prompts user for login
 then prompts user for directory
     or provides pull down menu with choices including full project
 then prompts for date using a calendar and clock set initially to the default
```

4.
```
checkout_one.py -e 1 -D "2006-10-31 10:05:00" c4studly
checkout_one.py g6keen
```

5.
```
checkout_one.py -d "2006-10-31 10:05:00" -e 1 c4studly
checkout_one.py g6keen
```

Question 5. [6 MARKS]

Part (a) [1 MARK]

CVS has the ability to access remote repositories; however to do this the user must specify the type of remote shell to use by setting an environment variable called `CVS_RSH`. Write a Unix command that would set this variable to the value `ssh`, the name of a commonly used remote shell that uses encryped communication. Specify the shell you assume is in use.

Part (b) [1 MARK]

Write a command that would print the value of the variable `CVSROOT` to the terminal.

Part (c) [2 MARKS]

You have written a Java program `MyGame.java` in the directory `/u/project1/code`. Your program needs access to a jar file called `myjar.jar` in the directory `/u/library/jars`. Describe one way in which you could make the jar file accessible to `javac`, thus allowing the program to be compiled.

Part (d) [2 MARKS]

Consider the following newsgroup posting:

```
I had no problems compiling my problem but when I try to run it I
get an error. I'm in the same subdirectory with my source code and the
class files. I've pasted the first few lines of the message below.
Can anyone tell me the problem?

$ java Exercise6
Exception in thread "main" java.lang.NoClassDefFoundError: Exercise6 (wrong name: exercise6/Exercise6
)
        at java.lang.ClassLoader.defineClass1(Native Method)
        at java.lang.ClassLoader.defineClass(Unknown Source)
```

What do you suspect is this student's problem and what advice can you give to fix the problem?

Question 6. [10 MARKS]

Write a python program that reads two commandline arguments. The first is a string providing a regular expression pattern and the second is a directory that may contain jar files. Your program must look inside all the jar files in the directory and list any class file whose full name contains a match for the pattern.

For example, to find classes used in CSC207 that are possibly implementing the factory design pattern, you might call your program like this:

```
python find_classes.py "[fF]actory" /u/csc207h/fall/pub/jar
```

Your answer goes here:

Question 7. [7 MARKS]

Part (a) [2 MARKS]

Write a Python program `get_uniq.py` that reads lines from the standard input and prints to the standard output a list of all the unique words in the input, in any order. Assume that the input file contains no punctuation.

Part (b) [2 MARKS]

Write a Python program `convert.py` that reads a list of words from the standard input, converts each word to lowercase, and then prints out the list in alphabetical order. Assume that each input word is on a separate line.

Part (c) [2 MARKS]

Write a one-line Unix command that uses the programs above to create a file called `sortedTokens.txt` containing the sorted, lowercase unique words from a file in the current directory called `text.txt`. Be careful here. Consider the desired output if the input is this text: `The cow jumped over the moon`

Part (d) [1 MARK]

For each of the python or plain text files used in the UNIX command in Part C, what UNIX permissions are necessary for the command to work properly?

Question 8. [8 MARKS]

Here is an XML file:

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="item_list.xsl" alternate="no" ?>
<part_list>
 <part>
  <number>a2048</number>
  <desc>bit</desc>
  <price>0.23</price>
 </part>
 <part>
  <number>s1024</number>
  <desc>bulb</desc>
  <price>1.44</price>
 </part>
</part_list>
```

Part (a) [1 MARK]

The file is read into a document using either Java's or Python's DOM (Document Object Model) facility. Draw a diagram showing the structure of the document, including node labels but not text contents.

Part (b) [1 MARK]

The XML file above might be read with Java's JDom or Python's minidom library. For some XML files you might prefer to use a SAX ("Streaming API for XML") library instead. How would such files differ from the example here?

Part (c) [6 MARKS]

Complete the Java method `averagePrice(Element root, String desc)` so that the following program prints the average price for all parts with the description "bit". The input XML files for your program will have a similar format to the file shown above, but they will include many more parts and many parts will share the same description.

```
// all necessary imports present but not shown
public class AveragePrice {
    public static void main(String[] args) {
        try {
          SAXBuilder builder = new SAXBuilder();
          Document doc = builder.build(args[0]);
          Element root = doc.getRootElement();
          System.out.println("Average price of bits:"+averagePrice(root,"bit"));
        } catch (Exception e) {
            System.err.println(e);
        }
    }
    private static double averagePrice(Element root, String desc) {
```

[This is a "blank" page. Use the space below for rough work. This page will ***not*** *be marked, unless you clearly indicate the part of your work that you want us to mark.]*

[This is a "blank" page. Use the space below for rough work. This page will **not** *be marked, unless you clearly indicate the part of your work that you want us to mark.]*

Total Marks = 65