University of British Columbia WMST 201/CPSC 101 - Winter 2009

Final Examination

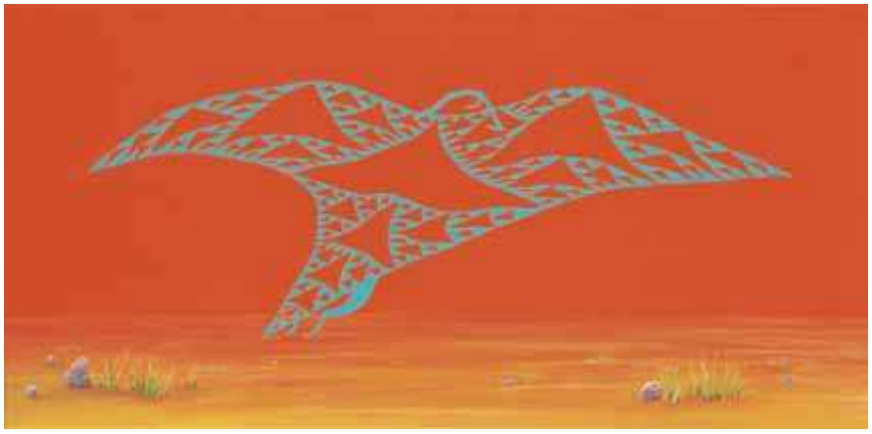
December 18, 2009

Name:

Student Number:

Keep the exam booklet closed until the beginning of the examination. Write your answers in the spaces provided. Write legibly. Take your time. Read each question carefully and think about the problem being asked. Check that you have 17 pages in total. Good luck!

Duration: 120 minutes



1	2	3	4	5	6	7	8	9	10	total
/12	/9	/10	/12	/9	/13	/10	/8	/8	/9	/100

1. (Multiple Choice/Short Questions, 12 marks)

   (a) (3 marks) Explain how six hexadecimal digits represent a colour in HTML.

   (b) (2 marks) Machine languages contain `jmp` instructions. Name an instruction in a high-level language which is translated into a sequence of instructions that include the `jmp`.

   (c) (2 marks) Which of the following statements are true for both the genetic code and the extended ASCII code? Check all that apply.

   ◯ Both are fixed-length codes.

   ◯ Both represent symbols in binary notation.

   ◯ Both codes represent information digitally.

   ◯ In both codes, there is exactly one "codeword" for each symbol (i.e., either an amino acid or an alphabet symbol) being represented.

   (d) (2 marks) Technology for automated weaving in the 1800's—the Jacquard Loom—used ideas that would later be adopted by computing technologies. Name one such idea.

   (e) (3 marks) The artwork on the preceding (cover) page of this exam, called "Wings", is by Ann Erpino. In what way is the pattern in the artwork self-similar?

2. (Minds & Machines, 9 marks)

   Alan Turing envisioned that it would be possible to build a computer that could pass the Turing test. That is, a human could interact (via an electronic interface) for five minutes with either another human or the computer, by asking questions and getting responses. The human would not be able to reliably tell whether the respondent was human or computer. Alan Turing then asked two questions about a computer that could pass the Turing Test:

   (i) Is the computer intelligent?
   (ii) Does the computer understand English?

   (a) (6 marks) For *one* of the questions above, state whether *you personally* think the answer is yes or no. Provide three clear justifications for your answer. For at least two of your justifications, refer to specific points made by Turing, Searle, or Hawkins & Blakelee. Indicate clearly which person made the points you refer to. Use full sentences throughout your answer.

   (b) (3 marks) Aamodt & Wang argue that building a machine with the information processing capabilities of the brain in the next two decades or so is infeasible. List two of their reasons.

3. (Fragment Assembly, 10 marks)

Below is a greedy algorithm for the Fragment Assembly Problem (FAP). Refer to this algorithm when answering the questions on the following page.

algorithm Greedy Algorithm for Fragment Assembly Problem
input: an instance of the Fragment Assembly Problem,
i.e., a list of DNA fragments
output: an assembly of the fragments which is
identified using a "greedy" approach

Choose any two fragments with the largest overlap (break ties randomly).
Place these first in the assembly.
Repeat until all fragments are in the assembly:
Choose the fragment that has the largest overlap with the right
end of the partially completed assembly (break ties randomly).

Add the chosen sequence to the right end of the assembly.

Stop once all sequences are added and output the resulting assembly.

(a) (2 marks) Write down the assembly that could result from the greedy algorithm when the input fragments are:

CGCAA, TTCGC, TTT

(b) (2 marks) Write down the two assemblies that could result from the greedy algorithm when the input fragments are:

GACAA, TTTGAC, ACTTT

(c) (3 marks) An alternative to the greedy algorithm is an enumeration approach. Briefly explain a major weakness of this approach.

(d) (3 marks) Why might an efficient algorithm for the Traveling Salesman Problem be useful for solving the Fragment Assembly Problem?

4. (Image Representation, 12 marks)

   (a) (2 marks) Give an example of a type of image that is well suited to vector image representation.

   (b) (4 marks) Describe two properties of image representation that distinguish bitmap representations from vector representations and explain the ways that the representations differ with respect to the properties you chose.

   (c) (2 marks) What was Sketchpad and how did it change the way that users create images?

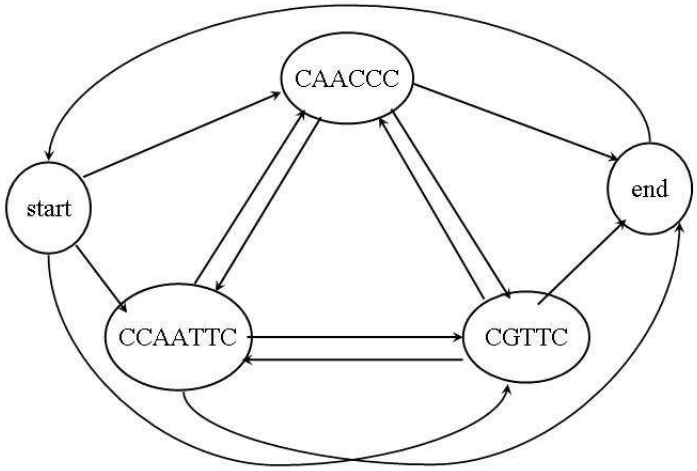
   (d) (1 mark) Did Sketchpad support vector or bitmap image representation?

   (e) (3 marks) In the 1970's, researchers at Xerox PARC developed a new type of computer screen technology. What was this technology called and how did it work?

5. (Traveling Salesman Problem, 9 marks)

   Each tour in the following network corresponds to an assembly of the fragments CGTTC, CAACCC, and CCAATTC. (For simplicity, edge costs are not shown.)

   (a) (4 marks) Using your pen or pencil, highlight ANY ONE tour in the network and write down the corresponding assembly of the fragments. Label the edges (i.e., arrows) between nodes (fragments) of your tour with the costs which are consistent with the reduction from the Fragment Assembly Problem to the Travelling Salesman Problem. (Note: Your tour does not have to be the lowest-cost tour.)



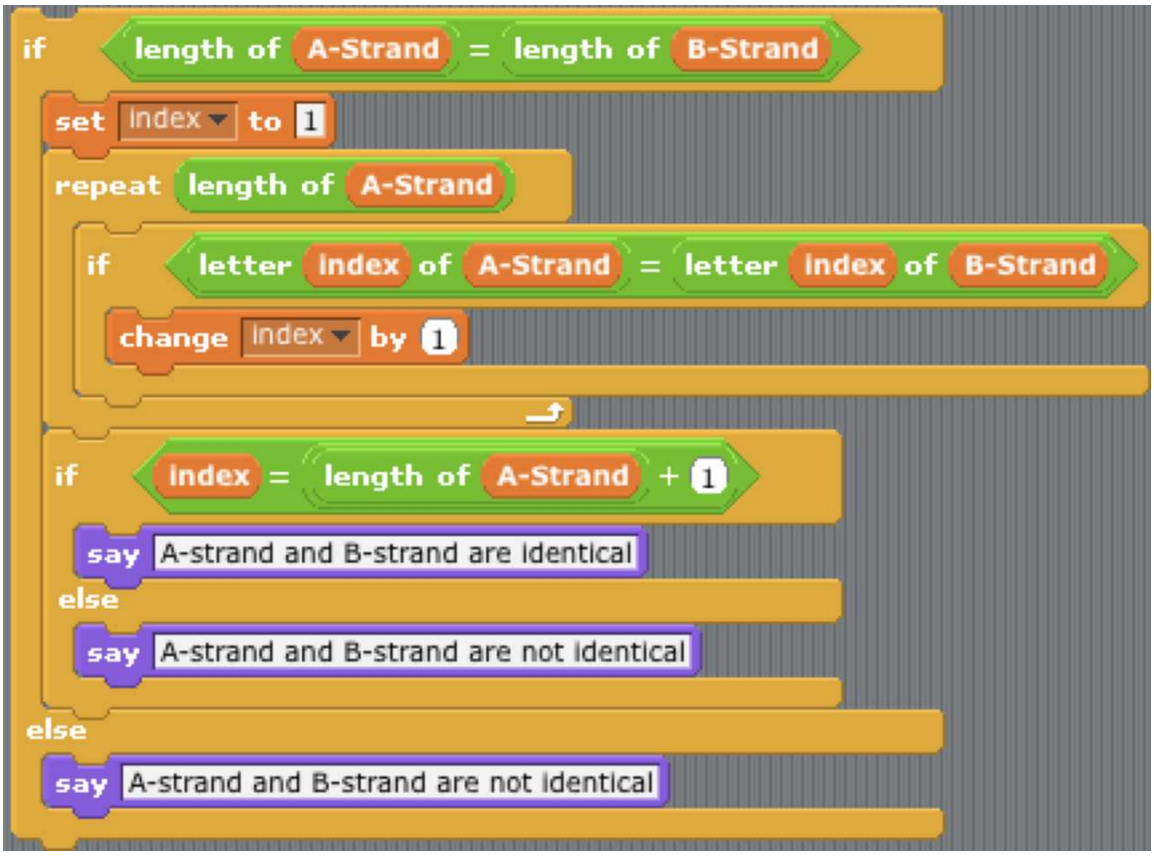
   (b) (2 marks) The following path through the above network is *not* a tour. Explain why.

   start → CAACCC → CCAATTC → CGTTC → CAACCC → end → start

   (c) (3 marks) Why is there a $1,000,000 prize for proving whether or not there is an efficient algorithm for solving the Traveling Salesman Problem?

6. (Scratch, 13 marks)

   The following Scratch script determines whether two strings, stored in variables `A-strand` and `B-strand`, are identical. (Such a script would be useful for fragment assembly, for example.)



(a) (2 marks) What are the variables in the script?

(b) (3 marks) Suppose that variable `A-strand` stores the strand AGG and variable `B-strand` stores the strand AAG. In this case, what is the value of `index` at the end of the script?

(c) (2 marks) Suppose that variable `A-strand` stores the strand ACGT and variable `B-strand` stores the strand ACGTT. In this case, how many times does the `repeat` loop get executed?

(d) (3 marks) Define `A-strand` to be a *proper prefix* of `B-strand` if the length of `A-strand` is less than the length of `B-strand` and the letters of `A-strand` appear in order *at the left end* of `B-strand`.

For example if `A-strand` is CACC and `B-strand` is CACCT then `A-strand` is a proper prefix of `B-strand`. But if `A-strand` is CACC and `B-strand` is AACACCT then `A-strand` is *not* a proper prefix of `B-strand` because, while the letters of `A-strand` appear in `B-strand`, they don't appear at the left end. Also, if `A-strand` is CACC and `B-strand` is CACC then `A-strand` is *not* a proper prefix of `B-strand` because both strands have the same length.

What changes would you make to the code so that the sprite says "A-strand is a proper prefix of B-strand" when the strand stored in variable `A-strand` is indeed a proper prefix of the strand stored in variable `B-strand`, and says "A-strand is not a proper prefix of B-strand" otherwise?

(e) (3 marks) What changes would you make to the code so that the script indicates whether or not `B-strand` is a proper prefix of `A-strand`?

(This page intentionally left blank, for your use to write notes or drafts of answers.)

7. (Javascript, 10 marks)

As a staff member in the communications department at UBC, you've been asked to build an on-line directory application that provides easy access to phone numbers for all instructors and TA's at UBC. Fortunately you recall that you learned about a javascript phone book application in your WMST201/CPSC101 class and figure you can expand on that. Review the javascript phone book code below and answer the questions which follow, so that you are all set to develop a UBC-wide phone book application.

```
 1. var phoneBook = new Array();
 2. phoneBook[0]  = new Array("Ada",     "604-123-4567");
 3. phoneBook[1]  = new Array("Anne",    "604-234-4567");
 4. phoneBook[2]  = new Array("Bruno",   "604-345-4567");
 5. phoneBook[3]  = new Array("Gillian", "604-456-4567");
 6. phoneBook[4]  = new Array("Maria",   "604-567-4567");
 7. phoneBook[5]  = new Array("Sapna",   "604-678-4567");
 8. phoneBook[6]  = new Array("Yong",    "604-789-4567");

 9.  function phoneNumber(request)
10. // returns the phone number of the person whose
11  // name is stored in request
12. {
13.    var index = 0; result = 0;
14.    while (index < phoneBook.length)
15.    {
16.       if (phoneBook[index][0] == request)
17.       { result = phoneBook[index][1]; }
18.       index = index + 1;
19.    }
20.    return result;
21. }
```

(This page intentionally left blank, for your use to write notes or drafts of answers.)

(a) (3 marks) For each of the following programming concepts, give a line number from the code with an example of that concept:

i. a comment:
ii. a variable declaration:
iii. a conditional:
iv. a loop:
v. an assignment statement:
vi. a parameter:

(b) (1 mark) What is phoneBook[3][1]?

(c) (1 mark) What is phoneBook.length?

(d) (1 mark) What value does the function call `phoneNumber("Bruno")` return?

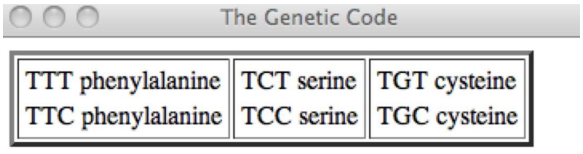
(e) (1 mark) What value does the function call `phoneNumber("Jason")` return?

(f) (3 marks) You'd like to add another person to the phone book. Her name is Giuliana and her phone number is 604-222-2222. What changes would you make to the code so that Giuliana's number is returned by the function call `phoneNumber("Giuliana")`?

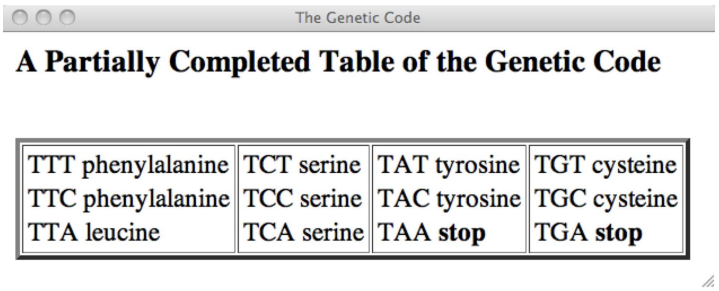
8. (HTML, 8 marks)

   Answer the questions on the next page about the following html specification.

```
<html>
<head> <title>  The Genetic Code    </title>  </head>

<body>
<table border = "3">
<tr>
  <td>
  <table>
    <tr> <td> TTT phenylalanine </tr> </td>
    <tr> <td> TTC phenylalanine </tr> </td>
  </table>
  </td>

  <td>
  <table>
    <tr> <td> TCT serine </td> </tr>
    <tr> <td> TCC serine </td> </tr>
  <table>
  </td>

  <tr>
  <table>
    <tr> <td> TGT cysteine </td> </tr>
    <tr> <td> TGC cysteine </td> </tr>
  </table>
  </tr>
</tr>
</table>
</body>
</html>
```



(a)



(b)

(a) (2 marks) The html code is intended to produce the web page shown in part (a) of the Figure at the bottom of page 14. However, there are four errors. Fix the code to correct these errors. You can mark your changes on the preceeding page.

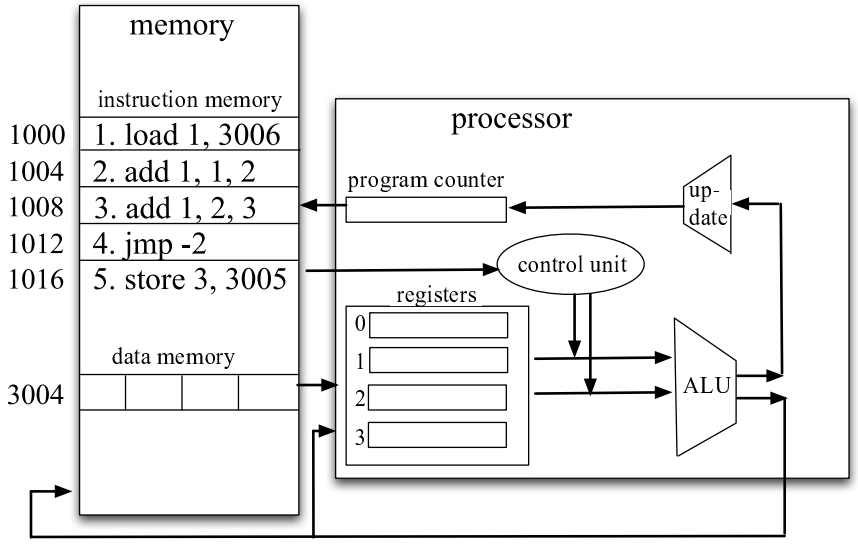
(b) (3 marks) Add additional code to produce the web page shown in part (b) of the Figure. Don't worry about getting spacing or heading sizes exactly right. You can add your code to the right of the given code, and use arrow to indicate where the new code should be inserted in the old code.

(c) (3 marks) Draw a rough sketch of the page that would be produced by the code below.

```
<html>
<head>
<title> The Genetic Code </title>
</head>

<body>
<table border = "3">
  <tr>
    <td> TTT phenylalanine </td>
    <td> TTC phenylalanine </td>
  </tr>

  <tr>
    <td> TCT serine </td>
    <td> TCC serine </td>
  </tr>

  <tr>
    <td> TGT cysteine </td>
    <td> TGC cysteine </td>
  </tr>

</table>
</body>

</html>
```

9. (Machine Language, 8 marks)

In the figure below, instructions of a program are placed in memory starting at location 1000. In the table below the figure, show what are the contents of the four registers, the program counter and the memory locations 3004, 3005, and 3006 after the program is executed for *six* steps. The initial values, before any instruction is executed, are given in the first row of the table.



instruction number	program counter	registers				memory locations		
		0	1	2	3	3004	3005	3006
	1000	4	3	5	3	6	3	4
1								
2								
3								
4								
5								
6								

10. (Interfaces, 9 marks)

    (a) (3 marks) Some cognitive scientists study the properties and limits of human perception. Describe a limitation of human perception that is important to consider when designing user interfaces.

    (b) (3 marks) List three principles of good graphical interface design.

    (c) (3 marks) For each of the principles that you list, describe a concrete example of how the GIMP user interface succeeds or fails in adhering to the principle.