Introduction to Computer Science 2012 Final Exam

Name: _____   ID: _____

1. Compare the following two pseudo codes of the same algorithm:

```
procedure PS1(n)
s ← 0;
i ← 1;
while (i is less than or equals to n)
do
(
    s ← s + i;
    i ← i + 1;
)
Return the value of s;
```

```
procedure PS2(n)
if (n equals 1) then
(
    Return the value 1;
)
Else
(
    Return the value PS2(n-1) + n;
)
```

   (a) If the number of addition and subtraction operations is used to estimate the complexity of these algorithms, what is the complexity of PS1 and PS2 in big-theta notation? **[5 points]**

   (b) If both algorithms are implemented on the same computer, which one will take longer to execute? Why do you think so? **[5 points]**

[Solution]

   (a) Both algorithms have complexity $\Theta(n)$.

   (b) PS2 will take longer because recursive structure (which uses function calls) has much higher overhead than iterative structure (which uses loop control).

2. Based on the preconditions that X and Y are assigned nonnegative integers, identify a loop invariant for the following while-loop that, when combined with the termination condition, implies that the value associated with Z upon loop termination must be X – Y. **[10 points]**

```
procedure PS3()
Z ← X;
J ← 0;
while (J is less than Y) do
(
    Z ← Z – 1;
    J ← J + 1;
)
```

[Solution]

   The loop invariant is "Z equals X – J."

3. Given the following recursive procedure PS4, answer the following questions if the procedure is started with N equals 1:

```
procedure PS4 (N)
print the value of N;
if (N < 3) then
{
    Call the procedure PS4(N+1)
}
print the value of N.
```

   (a) What sequence of numbers would be printed? **[5 points]**
   (b) What is the termination condition? **[5 points]**

[Solution]

    (a) 1 2 3 3 2 1.

    (b) N ≥ 3.

4. Please fill in the blanks in the following sentences: **[10 points]**

    (a) Most machine languages are based on the _____ programming paradigm.

    (b) The programming language Prolog solves problems based on facts and a reasoning engine. It belongs to the _____ programming paradigm.

    (c) Programming languages are _____ languages, which are precisely defined by grammars, unlike natural languages.

    (d) In order to solve a problem, the _____ programming paradigm requires the programmer to describe the problem to the computer, instead of asking the programmer to describe an algorithm for solving the problem to the computer.

    (e) The _____ programming paradigm focus on connecting smaller pre-defined program units so that the overall input-to-output relationship is obtained.
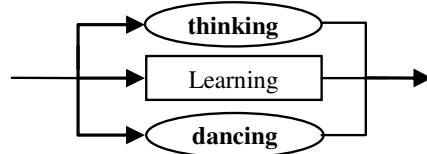
[Solution]

    (a) Imperative.

    (b) Declarative.

    (c) Formal.

    (d) Declarative.

    (e) Functional.

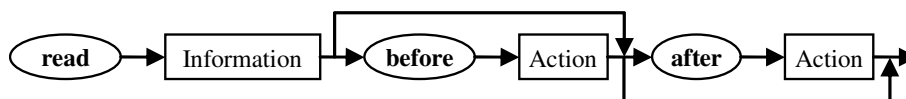5. Assuming that we have a programming language with the following syntax diagrams:

Information:

Action:

Learning:

Please answer the following questions:

    (a) Draw the parse tree of the following program statement: "read book before dancing after thinking" **[3 points]**

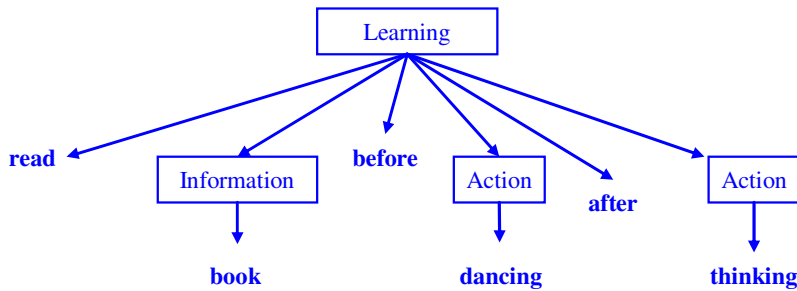   (b) The syntax of the language is ambiguous. Give an example statement that has more than

one parse trees. **[3 points]**

(c) How can you change the syntax diagram of "Learning" so that you can remove the ambiguity and parse the statement you gave in b) in only one way. **[4 points]**
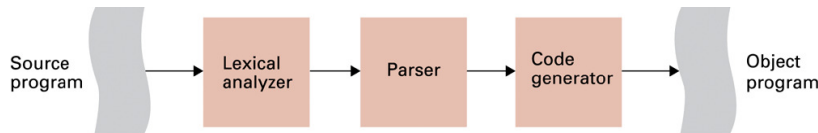
[Solution]

(a)

(b) Yes. For example, "read magazine before read book after dancing" is an ambiguous statement that has two possible parse trees.

(c)

Learning:

6. The translation process from a source program to machine codes is shown in the following diagram. Please describe what the three main activities: lexical analyzer, parser, and code generator do in the translation process. **[10 points]**

[Solution]

(a) Lexical analyzer – Converts the source program into tokens. Each token represents a reserved word, constant, or variable, etc.

(b) Parser – Use the syntax diagrams of the language to analyze the program (represented as a sequence of tokens) and generating parse trees.

(c) Code generator – Generates the intermediate representations and/or the target machine codes from the parse trees of the program.

7. In software engineering, we learn the concept of coupling and cohesion. However, the concept can be easily applied to domains beyond computer science. For example, a novel is composed of many chapters while an encyclopedia is composed of many sections.

(a) Which one has higher level of coupling: chapters in a novel or sections in an encyclopedia? Please explain why. **[5 points]**

(b) In principle, an interesting crime novel should have high coupling among its chapters and low cohesion within each chapter. Please explain why. Note that a crime novel is a novel of crime story, for example, the novels about Sherlock Holmes. **[5 points]**
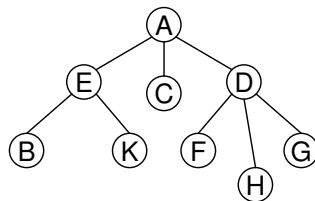
[Solution]

(a) Chapters in a novel, because chapters in a novel must tells a consistent story. On the other hand, sections in an Encyclopedia can cover totally unrelated topics.

(b) Crime novels often combine seemingly unrelated events in a chapter so that the readers cannot see through the plots easily. Thus, the events in a chapter may not be very cohesive.

8. In each of the following cases, identify whether the activity relates to a sequence diagram, a use case diagram, or a class diagram. **[10 points]**
(a) Represents the way in which users will interact with the system
(b) Represents the relationship between classes in the system
(c) Represents the manner in which objects will interact to accomplish a task

[Solution]

(a) Use case diagram.

(b) Class diagram.

(c) Sequence diagram.

9. Please show how we can store the following ternary tree (each node has at most three children) in a contiguous one-dimensional array without using pointers. Note that there are more than one way to do it, and you can mark an empty memory cell with an 'X'. **[10 points]**


[Solution]

A ECD BXK XXX FHG.

10. A student tries to use the following algorithm to search for a node in a binary tree:

```
procedure Search(tree, value)
if (root pointer of tree is NIL)
then
   (return a failure for the search;)
else
   (
      if (value equals value of root node) then
      (
         return a success for the search;
      )
      else if (value < value of root node) then
      (
         apply Search() to the left subtree, and
         return the result of the search;
      )
      else if (value > value of root node) then
      (
         apply Search() to the right subtree, and
         return the result of the search;
      )
   )
```

Assume that the tree, called **my_tree**, is as follows:
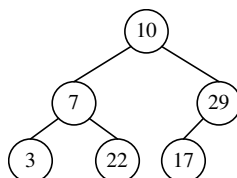
When the student calls Search(**my_tree**, 22), the search declares a failure.

(a) Why does the search fail? **[5 points]**

(b) Please make simple modifications to the algorithm so that Search() can return correct result for searching any binary tree. **[5 points]**

[Solution]

(a) The algorithm only works for an ordered binary tree. The input binary tree is not an ordered tree.

(b)

```
procedure Search(tree, value)
if (root pointer of tree is NIL)
then
   (return a failure for the search)
else
   (
      if (value equals value of root node)
      then
      (
         return a success for the search;
      )
      apply Search() to the left subtree;
      if (the result is a failure) then
      (
         apply Search() to the right subtree;
      )
      return the result of the previous search;
   )
```