Exams:

This test is timed. Your test will not be graded if you try to take more than the time allowed. Therefore, before you begin, please take a moment to look over the entire test so that you can budget your time.

Clarity is important; if your writing or code is sloppy or hard to read, you will lose points. Correct syntax also makes some difference.

There are 110 points all, including 10 bonus points.

1. (total 20 points) Short answers: multiple choice and fill-in-the-blank.

(a) (4 points) There are many different criteria for evaluating programming languages, including readability, writability, reliability, cost, and so on. Among these, the most important criterion for judging a programming languages is _____________. Justify your answer by stating the reason.

(b) (2 points) A language is said to be _____________ if it has a relatively small set of primitive constructs that can be combined in a relatively small number of ways to build the control and data structures of the language. Such a language is easy to learn and read because every possible combination is legal without an exception and the meaning is context independent.

(c) (2 points) The major methods of implementing programming languages are compilation, interpretation, and hybrid implementation. Java uses _____________.

Reminders

This test is closed-notes and closed-book. However, you are allowed to bring 1 page (8.5” × 11”) of notes (both sides). Your notes must be of your own, they must be hand written, and they must be turned in with your test. This test is to be done individually, and you are not to exchange or share materials with other students during the test.

If you need more space, use the back of a page. Note when you do that on the front.

This test is timed. Your test will not be graded if you try to take more than the time allowed. Therefore, before you begin, please take a moment to look over the entire test so that you can budget your time.

Clarity is important; if your writing or code is sloppy or hard to read, you will lose points. Correct syntax also makes some difference.

There are 110 points all, including 10 bonus points.
(d) (2 points) An attribute grammar is a BNF with three additions: attributes, attribute computation functions, and predicate functions. Among the three additions, the role of ________________ is to state the semantic rules or constraints of a grammar rule.

(e) (2 points) The address of a variable (l-value) is the machine memory address with which it is associated. Which of the following statements is incorrect about the address of a variable?
   i. A variable may have different addresses at different times during execution.
   ii. The same name may have different addresses at different places in a program.
   iii. If two variable names can be used to access the same memory location, they are called aliases.
   iv. Aliases are harmful to program efficiency.

(f) (2 points) A variable is characterized by attributes such as name, address, value, type, lifetime, and scope. The ________________ of a variable is the range of statements over which it is visible. It can be determined statically or dynamically.

(g) (2 points) Which storage binding scheme is described below? Storage bindings are created for variables when their declaration statements are elaborated. It allows recursion and conserves storage, however, it incurs overhead of allocation and deallocation and doesn’t support history sensitive variables for subprograms.
   i. Static
   ii. Stack-dynamic
   iii. Explicit heap-dynamic
   iv. Implicit heap-dynamic

(h) (2 points) PHP supports several different types of values including Booleans, integers, floating-point numbers, strings, objects, and resources. Every PHP value regardless its type is either true or false value.
   i. True
   ii. False

(i) (2 points) There can be different PHP variables mainly depending on their scopes and lifetimes, including all of the following EXCEPT for:
   i. Macro variables
   ii. Local variables
   iii. Global variables
   iv. Static variables
   v. Superglobal variables
2. (5 points) Pick the most suitable language for each of the following programming domains by considering only Fortran, COBOL, Prolog, C/C++, and PHP.

- Scientific applications
- Business applications
- Artificial intelligence
- Systems programming
- Web applications

3. (5 points) There are five different subscript and storage bindings possible for arrays: static, fixed stack-dynamic, stack-dynamic, fixed heap-dynamic, heap-dynamic. Name and describe briefly the approach used by Java.

4. (5 points) PHP provides two different equality operators, == and ===. State the difference between the two operators and show an example to highlight the difference.

5. (8 points) State the difference between static binding and dynamic binding, and give an example of each in terms of bindings of variable attributes occurring in the following Java local variable declaration.

```java
if (...) {
    int x = 10;
    ...
}
```
6. (10 points) This question is about the ambiguity of a grammar.

- When a grammar is said to be ambiguous? I.e., give the definition of the ambiguity of a grammar.
- Prove or disprove the ambiguity of the following grammar.

\[
\begin{align*}
  <E> & \rightarrow <E> \ast <E> \mid <id> \\
  <id> & \rightarrow X \mid Y \mid Z
\end{align*}
\]

7. (7 points) Write a BNF grammar for the language consisting of strings that have \( n \) copies of the letter \( a \) followed by \( 2 \ast n \) copies of the letter \( b \), where \( n > 0 \). For example, the strings \( abb, aabbbb, \) and \( aaabbbbb \) are in the language but \( a, b, ab, ba, \) and \( aab \) are not.
8. (10 points) Consider the following PHP code snippet.

(a) What is printed? Justify your answer by stating the reason.
(b) What would happen with dynamic scoping?

```php
$name = "Jack";
$closure = function ()
{
    use ($name)
    {
        return sprintf(“Hello,%s”, $name);
    };

    function callClosure()
    {
        global $closure;
        $name = "Jay";
        return $closure();
    }

    $name = "Jane";

    echo callClosure();
};
```

9. (15 points) Suppose a programming language PL$_m$ that supports multi-dimensional arrays. Your program written in PL$_m$ contains a two-dimensional array, say $A[100, 50]$ (100 rows $\times$ 50 columns). You learned that (i) each element of the array requires 2 bytes of memory, and (ii) $A[0, 0]$ and $A[10, 1]$ are stored at memory addresses 1000 and 2002, respectively, on a byte-addressable machine.

(a) Does PL$_m$ use row major order or column major order? Justify your answer by stating the reason.

(b) Write a formula that, given valid indices $i$ and $j$, gives the memory address where $A[i, j]$ is stored.

(c) Calculate the memory address of $A[10, 10]$. 
10. (total 25 points) Consider the Connect Four web service that you wrote for your PHP assignment. A Connect Four board consists of $m \times n$ places, and each place of a board is uniquely identified by a pair of 0-based column and row indices. Below you are to write partial code for two classes: Place and Board.

(a) (5 points) As shown above, each place of a board is represented by an instance of the Place class. Write the missing constructor and the `isEmpty()` function of the Place class shown below. The constructor takes a pair of 0-based column/row indices, say $x$ and $y$, and creates an empty place. The `isEmpty()` function tests if a place is unoccupied or not.

```php
class Place {
    private $x; // 0-based column, or slot, index of this place
    private $y; // 0-based row index of this place
    private $player; // occupant of this place

    function isEmpty() { ... } // Function to test if a place is empty
    function setPlayer($player) { ... } // Function to set the player
    function getX() { ... } // Function to get the column index
    function getY() { ... } // Function to get the row index
    ... // Other methods
}
```
(b) (5 points) Write the missing constructor of the Board class that takes two parameters, `$width` and `$height`. It should create an appropriate number of places and store them in an *one-dimensional array* `$places`; see the next two questions for the use of the array.

```php
class Board {
    private $width; // number of columns
    private $height; // number of rows
    private $places; // 1-d array containing all the places of this board

    function getPlace($x, $y) { ... } // to be written in (c) below
    function dropIn($x, $player) { ... } // to be written in (d) below
    ... } 
```

(c) (7 points) Write the `getPlace()` function of the Board class shown in Question 10b above. The `getPlace()` function returns the specified place; if the given column or row index is invalid, it returns null. Hint: use the `foreach` statement to iterate over all places of a board and find the specified place.
(d) (8 points) Write the `dropIn()` function of the Board class shown in Question 10b above. The function takes a column index ($x$) and a player ($\text{player}$) and lets the player occupy the lowest available place within the column. If the given index is invalid or the column is full, it has no effect.