

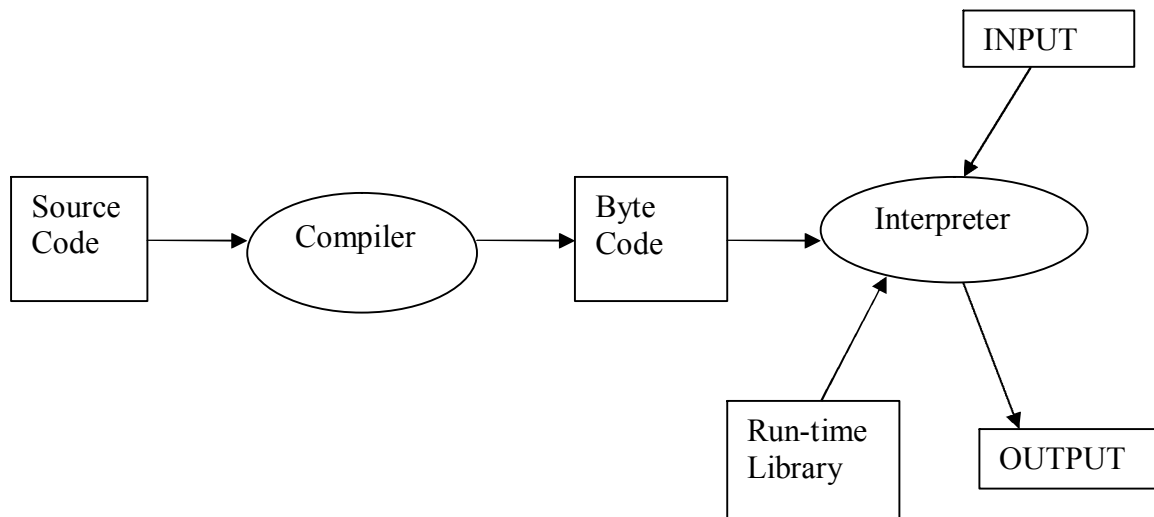
# CS471 Programming Language Structure I Fall, 1998

## Examination 1

Closed Book. Answer all the questions. The total points for each question or part of a question follows it in parentheses, thus: (12)

1

Java has both a compiler (which produces a "byte-code" version of the program) and an interpreter that interprets the byte codes. Draw a diagram that shows how a Java program produces output results. You should include the following features: compiler, interpreter, run-time library, source code, byte code. (16)



## 2

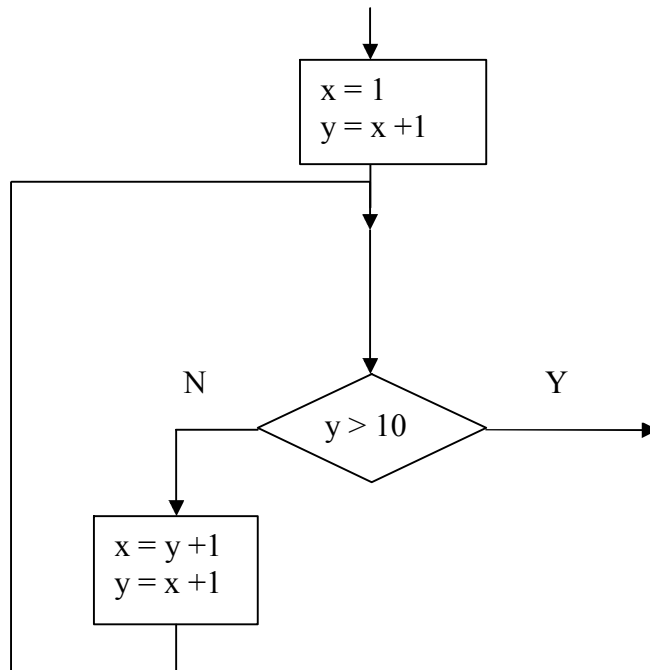
Below is an Ada loop construct. Executing the exit command causes the loop to terminate immediately. Transform the loop into an equivalent form that uses the standard structured programming while loop. You may use C or Pascal syntax, or you may draw the flow chart as your answer. (18)

```
x := 1;
loop
y := x + 1;
if y > 10 then exit;
x := y + 2;
end loop;
```

**Transformed version:**

```
x = 1;
y = x + 1;
while (y <= 10) {
  x = y + 2;
  y = x + 1;
}
```

**Flow chart:**



### 3

Look at the statement sequence below, written in a strongly-typed language. Explain how each statement is handled at compile and/or run time if:

the language is statically typed (i.e. all variables must have a type declaration). (12)

the language is dynamically typed (i.e. the declarations can be omitted) (12)

```

x, y : integer;
x = 1;
y = x + 1;
x = "abcd";
y = x + 1;
  
```

**Statically typed:**

**x = 1:** compile time: the type of the literal 1 is checked against the declared type of x. Run-time: the literal expression 1 is evaluated and the value bound to x's location

**y = x + 1:** compile time: the type of x is checked against that of 1 with integer addition. Run-time: the expression x + 1 is evaluated using integer addition to produce 2 which is bound to y

**x = "abcd":** this would be a compilation error

**y = x + 1:** just as the previous similar statement

**Dynamically typed:**

**x = 1:** the value 1 is bound directly to x

**y = x + 1:** the value of x is retrieved and added to 1 using integer addition; the value 2 is bound directly to y

**x = "abcd":** the value of the string "abcd" is bound directly to x

**y = x + 1:** the value of x is retrieved and found to be incompatible with the literal 1, so an error is produced

## 4

In the program below, written in C syntax, what is the value of g in the body of f1 with:

static scoping (8), and

dynamic scoping? (8)

By changing the order of function calls, show how g in the body of f1 could have the value 3 with dynamic scoping. (4)

```
int g = 1;
void f1();
void f2() {
    f1();
}
```

```
void f3() {
    int g = 2;
    f2();
}
```

```
void f1() {  
  /* what is the value of g here? */  
}
```

```
int main() {  
  int g = 3;  
  f3();  
}
```

### **Static scoping:**

**g=1**

### **Dynamic scoping:**

**g=2**

**To produce g=3: main calls f2 calls f1 or just main calls f1**

## **5**

Below is a grammar for a portion of Pascal. Show, by drawing a parse tree, that the string:

begin S1; begin S2; S3 end; begin S1; S3 end end

can be generated by the grammar. (16) Then alter the grammar, by making a simple change, so that the semi-colon becomes a statement terminator (as in C), not a separator, as it is in Pascal. (6)

```
<compound-stmt> ::= begin <stmt-seq> end  
<stmt-seq> ::= <stmt> | <stmt> ; <stmt-seq>  
<stmt> ::= <simple-stmt> | <compound-stmt>  
<simple-stmt> ::= S1 | S2 | S3
```

