

Test 2

Programming Language Concepts

April 20, 2020

Name: _____

1. (15 points) Write a program in C++ or C that includes two different enumeration types and has a significant number of operations using the enumeration types. Also write the same program using only integer variables.

Compare the readability and predict the reliability differences between the two programs.

2. (10 points) Perl allows both static and a kind of dynamic scoping. Write a Perl program that uses both and clearly shows the difference in effect of the two. Explain clearly the difference between the dynamic scoping described in this chapter and that implemented in Perl.

Online Perl Compiler:

https://www.tutorialspoint.com/execute_perl_online.php

3. (10 points) Write three functions in C/C++: one that declares a large array statically, one that declares the same large array on the stack, and one that creates the same large array from the heap. Call each of the subprograms a large number of times (at least 100,000) and output the time required by each. Explain the results.

Explain why or why not you can't do this in Java, what are the implications of this? If you can't specify which type of array you can't declare.

4. (10 points) Convert front.c and rda.c given in the content area of chapter 4; specifically the functions for `EXPR`, `TERM`, and `FACTOR` given in to Java; Add a working function for error and the Modulo operation in its proper order of operations.

Describe how this represents its precedence and associativity rules.

5. (10 points) Let the function `fun` be defined as

```
int fun(int *k) {
    *k += 4;
    return 3 * (*k) - 1;
}

void main() {
    int i = 10, j = 10, sum1, sum2;
    sum1 = (i / 2) + fun(&i);
    sum2 = fun(&j) + (j / 2);
}
```

Run the code in on some system that supports C and edit it to determine the values of `sum1` and `sum2`. Explain the results. Also, explain the results if there were no precedence rules.

6. (12 points) Consider the following program, written in **JAVASCRIPT-LIKE** syntax::

```
// main program
var x, y, z;
function sub1() {
    var a, y, z;
    . . .
}
function sub2() {
    var a, b, z;
    . . .
}
function sub3() {
    var a, x, w;
    . . .
}
```

Given the following calling sequences, what reference environments of the last subprogram activated at the line that contains the ellipses (. . .)? Include with each visible variable the name of the unit where it is declared. Provide one answer for dynamic scoping rules and one answer for static scoping rules.

- main calls sub1; sub1 calls sub2; sub2 calls sub3.
 - main calls sub1; sub1 calls sub3.
 - main calls sub2; sub2 calls sub3; sub3 calls sub1.
 - main calls sub3; sub3 calls sub1.
 - main calls sub1; sub1 calls sub3; sub3 calls sub2.
 - main calls sub3; sub3 calls sub2; sub2 calls sub1
7. (8 points) Evaluate $a > b > c$ in terms of mathematics (logical inequalities). Evaluate the same expression in terms of a C based language. Do the two mean the same thing? What does each expression say? If the two are the same show how they are the same or different.
8. (15 points) Assume the following rules of associativity and precedence for expressions:

Precedence

Highest

- (unary), (prefix) ++, (prefix) --
 (postfix) ++, (postfix) --
 +, *, &
 -, /, %, not
 < , <=, >=, >, !=
 =, +=, *=, /=
 and
 or, xor

Lowest

Associativity
 Left to right

Show the order of evaluation of the following expressions by parenthesizing all subexpressions and placing a superscript on the right parenthesis to indicate order. For example, for the expression $a * b + c / d$ the order of evaluation would be represented as

$((a * b)^1 + c)^2 / d)^3$

Also rewrite the expression where there are no precedence rules and the statement is given right to left associativity, For example for the same problem above

$d / c + b * a$

If you feel the statement can't be rewritten without parenthesis explain why?

- $a * b - 1 + c$
 - $++a * (b - 1) / c \% d$
 - $(a - b) / c \& (d * e / a - 3)$
 - $-a \text{ or } c = d \text{ and } e$
 - $a > b \text{ xor } c \text{ or } d \leq 17$
9. (10 points) Write a BNF description of the precedence and associativity rules defined for the expressions in problem above.