1. Assume the following rules of associativity and precedence for expression:

Precedence Highest \*, /, not

+, -, &, mod

-(unary)

=, /=, <. <=, >=, >

and

Lowest or, xor

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 \* c)1)2 + d)3

1. a \* b – 1 + c

(((a \* b)1 – 1)2 + c)3

1. a \* (b – 1) / c **mod** d

(((a \* (b – 1)1 )2/ c)3 **mod** d)4

1. (a – b) / c & (d \* e / a – 3)

(((a – b)1 / c)6 & ((((d \* e)3 / a)4 – 3)5)2)7

1. –a **or** c = d **and** e

((–a)1 **or** ((c = d)2 **and** e)3)4

1. a > b **xor** c **or** d <= 17

(((a > b)1 **xor** c)3 **or** (d <= 17)2)4

1. –a + b

(–(a + b)1)2

2. Show the order of evaluation of the expressions of the above expressions, assuming that

there are **no precedence** rules and all operators **associate right to left**.

1. a \* b – 1 + c

(a \* (b – (1 + c)1)2)3

1. a \* **(**b – 1**)** / c **mod** d

(a \* (**(**b – 1**)**2 / (c **mod** d)1)3)4

1. (a – b) / c & **(**d \* e / a – 3**)**

(**(**a – b**)**6 / (c & **(**(d \* (e / (a – 3)2)3)4**)**1)5)7

1. –a **or** c = d **and** e

(–(a **or** (c = (d **and** e)1)2)3)4

1. a > b **xor** c **or** d <= 17

(a > (b **xor** (c **or** (d <= 17)1)2)3)4

1. –a + b

(–(a + b)1)2