

1. What Does This Program Do? (Pascal)

If input to the following program are the numbers 12 and 10, what is printed when the program is run?

```

program sr3 (input, output);
  var a,b,c: integer;
  procedure two(var a,b: integer);
    begin a := a-2; b := b-3; end;
  function one (a,b: integer): integer;
    begin
      if a>5 then
        one := one(a-1,b)+b
      else if (a<=5) and (a>0) then
        begin two(a,b); one := one(a-1, b-1)+5; end
      else if (a<=0) then
        one := 4;
      end;
    begin {mainline}
      read(a,b);
      writeln(one(a,b));
    end.
  
```

2. LISP

Evaluate the following expression:

```
(ADD 1 (MULT 2 (SUB 3 (DIV 3 2)) 1))
```

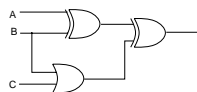
3. LISP

Evaluate the following expression:

```
(CDR (CDR (CAR (CDR '(2 ((3 (4 5) 6) 7))))))
```

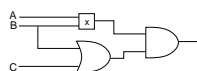
4. Digital Electronics

Find all ordered triples that make the following circuit true.



5. Digital Electronics

Find all gates (AND, OR, NAND, NOR, XOR, and XNOR) that can replace the “x” in the circuit below so that there is only one tuple that makes the circuit true.



1. The evaluation of this recursive program is as follows:

```

one(12, 10) = one(11, 10) + 10
one(11, 10) = one(10, 10) + 10
one(10, 10) = one(9, 10) + 10
one(9, 10) = one(8, 10) + 10
one(8, 10) = one(7, 10) + 10
one(7, 10) = one(6, 10) + 10
one(6, 10) = one(5, 10) + 10
one(5, 10) = two(5, 10) => a = 3, b = 7 => one(2, 6) + 5
one(2, 6) = two(2, 6) => a = 0, b = 4 => one(-1, 3) + 5
one(-1, 3) = 4
    
```

84

2. Evaluate the expressions by working from the inside to the outside:

```

(ADD 1 (MULT 2 (SUB 3 (DIV 3 2)) 1))
(ADD 1 (MULT 2 (SUB 3 1.5) 1))
(ADD 1 (MULT 2 1.5 1))
(ADD 1 3)
4
    
```

4

3. Work from the inside to the outside:

```

(CDR '(2 ((3 (4 5) 6) 7))) = (((3 (4 5) 6) 7))
(CAR '(((3 (4 5) 6) 7))) = ((3 (4 5) 6) 7)
(CDR '((3 (4 5) 6) 7)) = (7)
(CDR '(7)) = ()
    
```

() or nil

The parentheses are needed; the result of CDR is a list. In this case, a list with one element.

4. The following truth table shows the triples that make the expression true.

A	B	C	$A \oplus B$	$B + C$	$(A \oplus B) \oplus (B + C)$
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	1	1	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	0	1	1

(0,0,1), (1,0,0), (1,1,0), and (1,1,1)

5. The circuit can be represented as $(A?B)(B + C)$. For this to be true, both terms of the AND must be true. When $B + C$ is true, we have 6 possible solutions: $(* , 0, 1)$, $(* , 1, 0)$, and $(* , 1, 1)$. The problem claims that there's only one solution. Thus, the term $A?B = 1$ must cause exactly one of these 6 solutions to be true. The following truth table shows the relevant values of $A?B$:

A	B	C	AND	OR	NOR	NAND	XOR	XNOR
0	0	1	0	0	1	1	0	1
1	0	1	0	1	0	1	1	0
0	1	0	0	1	0	1	1	0
1	1	0	1	1	0	0	0	1
0	1	1	0	1	0	1	1	0
1	1	1	1	1	0	0	0	1

NOR