1. What Does This Program Do? (BASIC)

What is printed when the following program is run?

```
c = 0
a$ = "LAKEPLACIDNEWYORK"
for j = 1 to len(a$)-1
  if mid$(a$,j,1)<mid$(a$,j+1,1) then c=c+1
next j
print c
end
```

2. What Does This Program Do? (Pascal)

How different numbers are in the array `x` after the following program is run?

```
program intermediates(input,output);
var
  r, c: integer;
  x: array [1..11][1..11] of integer;
begin
  for r:=1 to 11 do
    for c:= 1 to 11 do
      x[r][c] := 1;
  for r:= 2 to 5 do begin
    for c:= 3 to 6 do begin
      x[r,c] := x[r+1,c+5]+2;
      x[r+5,c+5] := x[r-1,c+1]+3;
    end;
  end;
end.
```

3. What Does This Program Do? (Pascal)

When the following program is run, what is the final value of variable `b`?

```
program SR(input,output);
  var a,b,c,x:integer;
  begin
  b:=0;
  for a:=1 to 5 do begin
    x:=0;
    while x <= 5 do begin
      c:=5;
      repeat
        b:=b+c;
        c:=c-1;
      until c=0;
      x:=x+1;
      end;
    end
  end;
```

4. Prefix/Infix/Postfix Notation

Evaluate the following postfix expression, given that `a`=10, `b`=4, and `c`=2:

```
a b * c / a c + b c + / +
```

5. Prefix/Infix/Postfix Notation

Convert the following expression into prefix:

$$\frac{a * b^2}{c + 1} - \frac{a^2 + b}{a * b}$$

6. Prefix/Infix/Postfix Notation

Translate the following from prefix to postfix:

```
- + * 2 x * 3 y z
```

7. Prefix/Infix/Postfix Notation

If the binary operator `@` is the average of its two operands, evaluate the following postfix expression:

```
3 2 4 @ * 4 6 @ 3 * @
```

8. Data Structures

Consider the following sequence of operations on an empty stack:

```
push(a)
push(c)
push(s)
push(l)
pop
push(u)
pop
pop
push(s)
push(a)
pop
pop
```

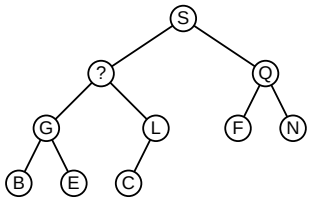
If the next operation were a `pop`, what would be popped off the stack?

9. Data Structures

Consider the binary search tree that is formed from the letters S N O W F L A K E, in that order. Now consider the binary search tree built from the letter in reverse order (that is, the letters E K A L F W O N S, in that order). Both trees have 9 nodes. What is the sum of the internal path lengths of the two trees?

10. Data Structures

Consider the following heap of 10 distinct letters:



List all letters that could replace the blackened node.

1. This program compares adjacent letters in `a$`, keeping count in variable `c` of the number of pairs that are in alphabetical order. The program prints the final value of `c`.

8

2. The final array is:

```
1 1 1 1 1 1 1 1 1 1 1
1 1 3 3 3 3 1 1 1 1 1
1 1 3 3 3 3 1 1 1 1 1
1 1 3 3 3 3 1 1 1 1 1
1 1 3 3 3 3 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 4 4 4 4
1 1 1 1 1 1 1 6 6 6 4
1 1 1 1 1 1 1 6 6 6 4
1 1 1 1 1 1 1 6 6 6 4
1 1 1 1 1 1 1 1 1 1 1
```

4

3. The body of the `for` loop is repeated 5 times. Each time, the body of the `while` loop is repeated 6 times (for `x`=0, 1, 2, 3, 4, and 5). Each time in the while loop, the value of `b` is incremented by 5+4+3+2+1 = 15. Thus, `b` is incremented by 15 a total of 30 times.

450

4. The expression converts to infix as follows (an expression is boxed after it has been converted):

`[a * b] c / [a + c] [b + c] / +`

$$\left[\frac{a * b}{c}\right] \left[\frac{a + c}{b + c}\right] +$$

$$\left[\frac{a * b}{c} + \frac{a + c}{b + c}\right]$$

Substitute the values of the variables:

$$\frac{a * b}{c} + \frac{a + c}{b + c} = \frac{10 * 4}{2} + \frac{10 + 2}{4 + 2} = 20 + 2 = 22$$

22

5. The expression converts to − [x] [y],
where [x] and [y] are the operands of the minus.
[x] converts to / [p] [q],
where [p] is * a ↑ b 2 and [q] is + c 1.
[y] converts to / [s] [t],
where [s] is + ↑ a 2 b and [t] is * a b.

- / * a ↑ b 2 + c 1 / + ↑ a 2 b * a b

6. A straightforward approach to this problem is to convert the prefix expression to infix, and then the infix into postfix. The conversion to infix is as follows:

```
-+*2x*3yz  =  -+[2*x] [3*y]z
           =  -[2*x+3*y]z
           =  [2*x+3*y-z]
```

The conversion from infix to postfix is as follow:

```
2*x+3*y-z  =  [2x*]+[3y*]-z
           =  [2x*3y*+]-z
           =  [2x*3y*+z-]
```

2 x * 3 y * + z -

7. The evaluation is as follows:

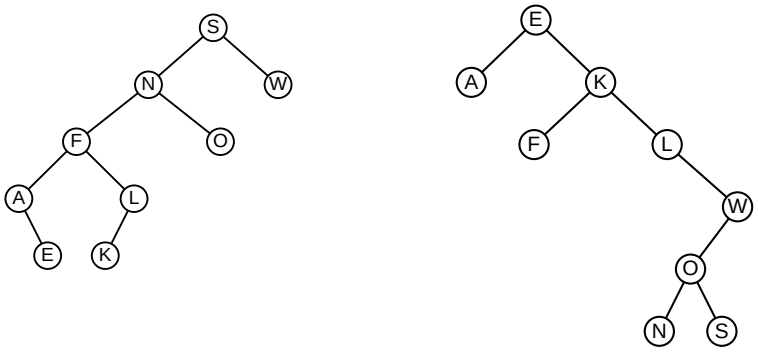
```
3 2 4 @ * 4 6 @ 3 * @
3 [3] * [5] 3 * @
[9] [15] @
[12]
```

12

8. Remember that a stack is "last-in, first-out." That is, the last item pushed will be the first removed. The items popped are `L`, `U`, `S`, `A`, and `S` in this order. The next item to be popped will be `C`.

`C`

9. The two trees are shown below. The tree at the left (SNOWFLAKE) has an internal path length of 20; at the right (EKALFWONS) 23. There is no simple relationship between the pair of trees formed in this way.



43

10. In a heap, every node must be larger than its children. So, the blackened node must be smaller than S (its parent) and larger than G and L (its children). This gives 6 possibilities: M, N, O, P, Q, and R. However, two of those (Q and N) are already in the heap, so they cannot be in the blackened node.

M, O, P, and R (in any order)