“Evaluation of a Prefix Expression”.

1. How many stacks are required for evaluation of prefix expression?
a) one
b) two
c) three
d) four
View Answer
Answer: b
Explanation: 2 stacks are required for evaluation of prefix expression, one for integers and one for characters.

2. While evaluating a prefix expression, the string is read from?
a) left to right
b) right to left
c) center to right
d) center to left to right
View Answer
Answer: b
Explanation: The string is read from right to left because a prefix string has operands to its right side.

3. The associativity of an exponentiation operator ^ is right side.
a) True
b) False
View Answer
Answer: a
Explanation: The associativity of ^ is right side while the rest of the operators like +,-,*,/ has its associativity to its left.

4. How many types of input characters are accepted by this algorithm?
a) one
b) two
c) three
d) four
View Answer

Answer: c
Explanation: Three kinds of input are accepted by this algorithm- numbers, operators and new line characters.

5. What determines the order of evaluation of a prefix expression?
a) precedence and associativity
b) precedence only
c) associativity only
d) depends on the parser
View Answer
Answer: a
Explanation: Precedence is a very important factor in determining the order of evaluation. If two operators have the same precedence, associativity comes into action.

Check this: Computer Science Books | Data Structure Books

6. Find the output of the following prefix expression.

```
*+2-2 1/-4 2+-5 3 1
```

a) 2
b) 12
c) 10
d) 4
View Answer
Answer: a
Explanation: The given prefix expression is evaluated using two stacks and the value is given by (2+2-1)*(4-2)/(5-3+1)= 2.

7. An error is thrown if the character '\n' is pushed in to the character stack.
a) true
b) false
View Answer
Answer: b
Explanation: The input character '\n' is accepted as a character by the evaluation of prefix expression algorithm.

8. Using the evaluation of prefix algorithm, evaluate +-9 2 7.
a) 10
b) 4
c) 17
d) 14
View Answer

Answer: d
Explanation: Using the evaluation of prefix algorithm, +-9 2 7 is evaluated as 9-2+7=14.

9. If -*+abcd = 11, find a, b, c, d using evaluation of prefix algorithm.
a) a=2, b=3, c=5, d=4
b) a=1, b=2, c=5, d=4
c) a=5, b=4, c=7,d=5
d) a=1, b=2, c=3, d=4

View Answer

Answer: b
Explanation: The given prefix expression is evaluated as ((1+2)*5)-4 = 11 while a=1, b=2, c=5, d=4.

10. In the given C snippet, find the statement number that has error.

```
//C code to push an element into a stack
1. void push( struct stack *s, int x)
2. {
3.     if(s->top==MAX-1)
4.     {
5.         printf(“stack overflow”);
6.     }
7.     else
8.     {
9.         s->items[++s->top]=x;
10.        s++;
11.    }
12. }
```

a) 1
b) 9
c) 10
d) 11

View Answer

Answer: c
Explanation: If the stack is not full then we are correctly incrementing the top of the stack by doing “++s->top” and storing the value of x in it. However, in the next statement “s++”, we are un-necessarily incrementing the stack base pointer which will lead to memory corruption during the next push() operation.