GeeksQuiz

Computer science mock tests for geeks

Search

- Home
- Latest Questions
- Articles
- C/C++ Programs
- Contribute
- Books

Subscribe

Stack | Set 4 (Evaluation of Postfix Expression)

June 30, 2014

The Postfix notation is used to represent algebraic expressions. The expressions written in postfix form are evaluated faster compared to infix notation as parenthesis are not required in postfix. We have discussed infix to postfix conversion. In this post, evaluation of postfix expressions is discussed.

Following is algorithm for evaluation postfix expressions.
1) Create a stack to store operands (or values).
2) Scan the given expression and do following for every scanned element.
…..a) If the element is a number, push it into the stack
…..b) If the element is a operator, pop operands for the operator from stack. Evaluate the operator and push the result back to the stack
3) When the expression is ended, the number in the stack is the final answer

Example:
Let the given expression be “2 3 1 * + 9 -“. We scan all elements one by one.
1) Scan ‘2’, it’s a number, so push it to stack. Stack contains ‘2’
2) Scan ‘3’, again a number, push it to stack, stack now contains ‘2 3′ (from bottom to top)
3) Scan ‘1’, again a number, push it to stack, stack now contains ‘2 3 1′
4) Scan ‘*’, it’s an operator, pop two operands from stack, apply the * operator on operands, we get 3*1 which results in 3. We push the result ‘3’ to stack. Stack now becomes ‘2 3′.
5) Scan ‘+’, it’s an operator, pop two operands from stack, apply the + operator on operands, we get 3 + 2 which results in 5. We push the result ‘5’ to stack. Stack now becomes ‘5’.
6) Scan ‘9’, it’s a number, we push it to the stack. Stack now becomes ‘5 9′.
7) Scan ‘-‘, it’s an operator, pop two operands from stack, apply the – operator on operands, we get 5 – 9 which results in -4. We push the result ‘-4′ to stack. Stack now becomes ‘-4′.
8) There are no more elements to scan, we return the top element from stack (which is the only element left in stack).

Following is C implementation of above algorithm.

```
// C program to evaluate value of a postfix expression
#include <stdio.h>
#include <string.h>
#include <ctype.h>
#include <stdlib.h>

// Stack type
struct Stack
{
    int top;
    unsigned capacity;
    int* array;
};

// Stack Operations
struct Stack* createStack( unsigned capacity )
{
    struct Stack* stack = (struct Stack*) malloc(sizeof(struct Stack));

    if (!stack) return NULL;

    stack->top = -1;
    stack->capacity = capacity;
    stack->array = (int*) malloc(stack->capacity * sizeof(int));

    if (!stack->array) return NULL;

    return stack;
}

int isEmpty(struct Stack* stack)
{
    return stack->top == -1 ;
}

char peek(struct Stack* stack)
{
    return stack->array[stack->top];
}

char pop(struct Stack* stack)
{
    if (!isEmpty(stack))
        return stack->array[stack->top--] ;
    return '$';
}

void push(struct Stack* stack, char op)
{
    stack->array[++stack->top] = op;
}
```

```
// The main function that returns value of a given postfix expression
int evaluatePostfix(char* exp)
{
    // Create a stack of capacity equal to expression size
    struct Stack* stack = createStack(strlen(exp));
    int i;

    // See if stack was created successfully
    if (!stack) return -1;

    // Scan all characters one by one
    for (i = 0; exp[i]; ++i)
    {
        // If the scanned character is an operand or number,
        // push it to the stack.
        if (isdigit(exp[i]))
            push(stack, exp[i] - '0');

        //  If the scanned character is an operator, pop two
        // elements from stack apply the operator
        else
        {
            int val1 = pop(stack);
            int val2 = pop(stack);
            switch (exp[i])
            {
             case '+': push(stack, val2 + val1); break;
             case '-': push(stack, val2 - val1); break;
             case '*': push(stack, val2 * val1); break;
             case '/': push(stack, val2/val1);   break;
            }
        }
    }
    return pop(stack);
}

// Driver program to test above functions
int main()
{
    char exp[] = "231*+9-";
    printf ("Value of %s is %d", exp, evaluatePostfix(exp));
    return 0;
}
```

Output:

```
Value of 231*+9- is -4
```

Time complexity of evaluation algorithm is O(n) where n is number of characters in input expression.

There are following limitations of above implementation.
1) It supports only 4 binary operators ‘+’, ‘*’, ‘-‘ and ‘/’. It can be extended for more operators by adding

more switch cases.
2) The allowed operands are only single digit operands. The program can be extended for multiple digits by adding a separator like space between all elements (operators and operands) of given expression.

References:
http://www.cs.nthu.edu.tw/~wkhon/ds/ds10/tutorial/tutorial2.pdf

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

Tags: Stack

Related Questions:

- Binary Heap
- Delete all occurrences of a given key in a linked list
- How to create mergable stack?
- Deque | Set 1 (Introduction and Applications)
- A data structure for n elements and O(1) operations
- Convert left-right representation of a bianry tree to down-right
- Print level order traversal line by line
- C Program for Red Black Tree Insertion

Like 5 Tweet 0 g+1 1

14 Comments GeeksQuiz Login

Recommend Share Sort by Best

Join the discussion…

mb1994 • 8 months ago

The value stored should be Val 2 (op) Val1. You've done the opposite here. It's incorrect.

4 • Reply • Share ›

Emmanuel Livingstone • 8 months ago

Infix expression a-b converts to postfix ab-
So I think for subtraction and division, the code should do val2-val1 and val2/val1. Also the answer for the given test case would be -4.
Correct me if I'm wrong.

4 • Reply • Share ›

GeeksforGeeks Mod • 7 months ago

@All Thanks for pointing this out We have updated expressions to val2 op val1.

3 ⌃ | ⌄ • Reply • Share ›

abhishek • 2 months ago

i am getting error in switch case
error: subscripted value is neither array nor pointer nor vector
switch(a[i])
can anybody tell me why this is coming?

1 ⌃ | ⌄ • Reply • Share ›

Sahdev • 8 months ago

how would it deal with unary operators and numbers greater than 9 like 12 or 152

1 ⌃ | ⌄ • Reply • Share ›

monica • 8 months ago

Answer should be -4!

1 ⌃ | ⌄ • Reply • Share ›

Vijay Yadav ➦ monica • 8 months ago

no,its correct

⌃ | ⌄ • Reply • Share ›

Gaurav ➦ Vijay Yadav • 7 months ago

yeah. the answer should be -4. Reason being that the first pop goes to RHS of operator and next pop goes to LHS. so in the code there's an error
case '+': push(stack, val2 + val1); break;
case '-': push(stack, val2 - val1); break;
case '*': push(stack, val2 * val1); break;
case '/': push(stack, val2/val1); break;
// see the change in val2 and val1 position

⌃ | ⌄ • Reply • Share ›

Emmanuel Livingstone ➦ Vijay Yadav • 8 months ago

Try the postfix expression evaluator at http://www.abecedarical.com/ja... with the value "2 3 1 * + 9 -"
It returns -4.
Also it is logical. See my comment below.

⌃ | ⌄ • Reply • Share ›

navin • 2 months ago

what is the exact meaning of using struct Stack* stack

⌃ | ⌄ • Reply • Share ›

sarat chandra • 6 months ago

why is exp[i]-'0' has been used in the push

• Reply • Share ›

mast monsoon ➦ sarat chandra • 6 months ago

SARAT CHANDRA :
character '2' value is 42 char '0' is 40 so exp[i]-'0' is 42-40=2
other wise char'2' x char'3' = 42x43=1806 which is incorrect

• Reply • Share ›

soumya • 6 months ago

what if the entered operands are 2 digit or 3 digit variations.. then it your algo won't work as it takes single digit numbers only as operands.. is there any way to workaround this problem?

• Reply • Share ›

baap_ka_baap • 7 months ago

switch case should have break statement in every case

• Reply • Share ›

Subscribe Add Disqus to your site Privacy

GeeksQuiz

Like

13,821 people like GeeksQuiz.

Facebook social plugin

-
-
- # Categories
  - Articles (124)
    - Algorithms (24)

        - C (16)
        - C++ (17)
        - Data Structures (29)
        - DBMS (1)
        - Interview Experiences (6)
        - Java (2)
        - Operating Systems (1)
        - Puzzle (12)
        - Searching and Sorting (10)
    - Programs (35)
    - Quizzes (2,106)
        - Aptitude (2)
        - Computer Science Quizzes (2,103)
            - Algorithms (147)
            - C (207)
            - C++ (129)
            - Computer Organization and Architecture (1)
            - Data Structures (132)
            - DBMS (2)
            - GATE (1,406)
            - Java (51)
            - Operating Systems (28)
        - Web technologies (1)
-

- ## Recent Discussions

    - Vidhi

      Process P2 will start it's last I/O operation...

      Operating Systems | Process Synchronization | Question 4 · 1 day ago

    - xerosanyam

      This article is slightly less clear in one...

      A Programmer's approach of looking at Array vs. Linked List · 1 day ago

    - Aditya Goel

      Please include Heap overflow condition as well....

      Queue | Set 2 (Linked List Implementation) · 1 day ago

    - Aditya Goel

      We can initialize rear as -1 with no change in...

      Queue | Set 1(Introduction and Array Implementation) · 1 day ago

  - Sg

    Implementing this question leaves an extra...

    GATE | GATE-CS-2015 (Set 3) | Question 22 · 2 days ago

  - jyuan92

    What if I want to handle the situation that,...

    Remove comments from a given C/C++ program · 2 days ago

-

Valid XHTML Strict 1.0
Powered by WordPress & MooTools | MiniMoo 1.3.4