**计算机科学与工程学院实验报告2**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验课程名称** | | **实验二 栈和队列应用类实验**  **——算术表达式求值** | | | **实验成绩** |  |
| **专业** | **计算机科学与技术** | | **班级** | **\*** | **指导教师签字** |  |
| **姓名** | **\*** | | **姓名** | **\*** | **姓名** |  |
| **学号** | **\*** | | **学号** | **\*** | **学号** |  |
| **附录实验程序及结果：**  **实验分工以及占比：**  **\* 50%：**主要任务为实现中缀表达式转后缀表达式  **\* 50%：**主要任务为实现后缀表达式求值  **实验程序：**    **结果：**    **源代码：**   #include<iostream>  #include<string>  #include<stack>  using namespace **std**;  *//中缀表达式转后缀表达式函数*  **string** **change**(const **string** str)  {  **stack**<char> mystack;  **string** strTarget;  **string** ch = "+-\*/()";  **string** s = "0123456789+-\*/()";  **string** tempstr;       int start, end;       for (int i = 0; i < str.**length**(); i = end)       {            start = i;            end = str.**find\_first\_of**(ch, i);*//肯定是一个符号*            if (start == end)            {                 tempstr **=** str.**substr**(start, 1);                 end++;            }            else            {                 if (end == -1)*//处理遍历到最后的情况*                 {                      end = str.**length**();                 }                 tempstr **=** str.**substr**(start, end - start);            }            if (tempstr **==** "(")*//左括号*            {                 char c = '(';                 mystack.**push**(c);            }            else if (tempstr **==** ")")*//右括号*            {                 while (mystack.**top**() != '(')                 {                      strTarget **+=** mystack.**top**();                      strTarget **+=** " ";                      mystack.**pop**();                 }                 mystack.**pop**();*//右括号出栈*            }            else if ((tempstr **==** "+") || (tempstr **==** "-"))            {                 while ((mystack.**empty**() != true) && (mystack.**top**() != '('))                 {                      strTarget **+=** mystack.**top**();                      strTarget **+=** " ";                      mystack.**pop**();                 }                 char c = tempstr**[**0**]**;                 mystack.**push**(c);            }            else if ((tempstr **==** "\*") || (tempstr **==** "/"))            {                 while ((mystack.**empty**() != true) && ((mystack.**top**() == '\*') || (mystack.**top**() == '/')))                 {                      strTarget **+=** mystack.**top**();                      strTarget **+=** " ";                      mystack.**pop**();                 }                 char c = tempstr**[**0**]**;                 mystack.**push**(c);            }            else*//数字*            {                 strTarget **+=** tempstr;                 strTarget **+=** " ";            }       }       while (mystack.**empty**() != true)       {            strTarget **+=** mystack.**top**();            strTarget **+=** " ";            mystack.**pop**();       }       return strTarget;  }  int **main**()  {  **string** str;  **string** ans;*//用来接受转换后的后缀表达式*       cin **>>** str;       ans **=** **change**(str);       ans**[**ans.**length**()**]** = '#';  **stack**<int> st;       int i = 0,t=-1,m,n,ss;       while (ans**[**i**]**!='#') {            if (ans**[**i**]** >= '0' && ans**[**i**]** <= '9')            {                 if (t == -1) {                      t = ans**[**i**]** - '0';                      st.**push**(t);                 }                 else {                      t = st.**top**();                      st.**pop**();                      t = t \* 10 + (ans**[**i**]** - '0');                      st.**push**(t);                 }            }            if (ans**[**i**]** == ' ')                 t = -1;            if (ans**[**i**]** == '+') {                 m = st.**top**();                 st.**pop**();                 n = st.**top**();                 st.**pop**();                 ss = m + n;                 st.**push**(ss);            }            if (ans**[**i**]** == '-') {                 m = st.**top**();                 st.**pop**();                 n = st.**top**();                 st.**pop**();                 ss = n - m;                 st.**push**(ss);            }            if (ans**[**i**]** == '\*') {                 m = st.**top**();                 st.**pop**();                 n = st.**top**();                 st.**pop**();                 ss = n \* m;                 st.**push**(ss);            }            if (ans**[**i**]** == '/') {                 m = st.**top**();                 st.**pop**();                 n = st.**top**();                 st.**pop**();                 ss = n / m;                 st.**push**(ss);            }            i++;       }       cout **<<** ans **<<** **endl**;       cout **<<** st.**top**() **<<** **endl**;  **system**("pause");       return 0;  } | | | | | | |
| **实验课程总结**  通过本次实验课程内容的学习，我了解了有关栈和队列数据结构的相关问题。主要学习的是有关栈的使用，本次实验中利用的是c++ stl模板里的stack。在初步编写时遇到了一些困难，如中缀转后缀的时候符号优先级的问题等。本次实验内容提升了我对栈和队列相关知识的了解，加强了有关栈的应用。 | | | | | | |

**批改时间：**