

**X0\_Define\_Expansion Compiler**  
**软件测试说明书**

**作者:钱庭涵**

**版本:v1.0**

**日期:2018/12/16**

Project Name/Model: X0\_Define\_Compiler

## 1. 概述

测试描述	本文档进行的测试是用于测试 <b>X0_Define_Expansion Compiler</b> （即针对扩展后的 <b>X0</b> 语言实现的编译器）的各项语法实现功能、以及对于各种可能出现的符合语法规则的情况的处理，附带一些出错处理检验。扩展后 <b>X0</b> 语言的文法及功能、编译器使用说明，请看编译器设计说明文档。
测试环境	编程语言: <b>Python 3.5</b> IDE: <b>JetBrains PyCharm</b> 运行环境: <b>Windows 10 64bit</b>

## 2. 测试用例描述

Test No.	1	test_1.txt
测试目标	测试基本语法功能，输入输出、if 语句、四则运算等	
测试用例的源程序	<pre>main{     int x;     int y;     int m[3];     read x;     read y;     if(x &gt; y) m[0]=x-y;     else m[0]=y-x;     m[1]=x+y;     m[2]=x*y;     write m[0];     write m[1];     write m[2]; }</pre> <p>/*输入两个数 x,y，依次输出它们的差的绝对值、和、乘积*/</p>	
测试输入	6 7	

Project Name/Model: X0\_Define\_Compiler

测试结果	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 6</div><div>Input? 7</div><div>1</div><div>13</div><div>42</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试是否通过	PASS


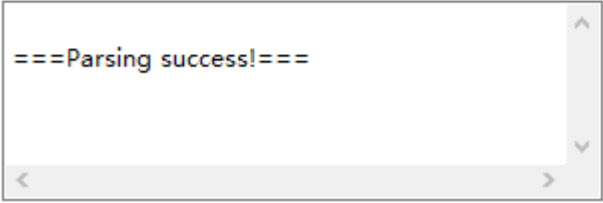
Test No.	2	test_2.txt
测试目标	测试 while 循环语句	
测试用例的源程序	<pre>main{   int i;   int sum;   int mul;   i=1;   sum=0;   while(i&lt;=10){     sum = sum+i;     i=i+1;   }   write sum; }  /*输出 1~10 的和*/</pre>	

Project Name/Model: X0\_Define\_Compiler

测试输入	无
测试结果	<div> <b>fresult</b> <div> Start x0 55 End x0 </div> </div> <div> <b>foutput</b> <div> ===Parsing success!=== </div> </div>
测试是否通过	PASS

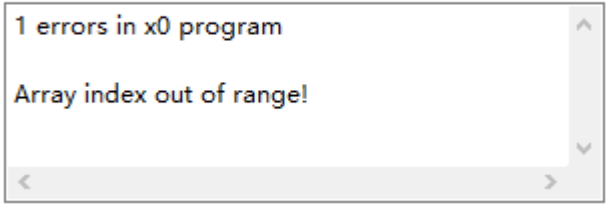
Test No.	3	test_3.txt
测试目标	测试 write 语句的输出、整型变量的输出与字符型变量的输出	
测试用例的源程序	<pre> main{ char c; int x; int y; int m; read x; read y; m=(x*x+y*y)/2; c=97; write m; write c; /*应该输出对应字符*/ write (c+1); /*应该输出数值*/ }  /*输入两个数 x,y; 输出两个数平方的平均数。检测字符型变量的值*/ </pre>	

Project Name/Model: X0\_Define\_Compiler

测试输入	3 5
测试结果	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试是否通过	PASS

Test No.	4	test_4.txt
测试目标	测试连续赋值、while 循环、带变量索引的数组元素的输出、数组越界出错处理、数组元素未赋值就使用出错处理	
测试用例的源程序	<pre> main{   int i;   int p[20];   int now;   i=now=p[0]=1; /*根据实际情况给分*/   while(i&lt;=10){     now=now*i;     p[i]=now;     i=i+1;   }   while(i&gt;0){ </pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>i=i-1; write p[i]; } write p[20];/*应该报错*/ } /*计算并依次输出 1~10 的阶乘*/</pre>
测试输入	无
测试结果	<p><b>foutput</b></p>  <p>即数组越界出错处理</p>
修改后源程序 1	<pre>main{ int i; int p[20]; int now; i=now=p[0]=1; while(i&lt;=10){ now=now*i; p[i]=now; i=i+1; } while(i&gt;0){ i=i-1; write p[i]; } write p[15];/*改为 15*/ }</pre>
测试输入 1	无

Project Name/Model: X0\_Define\_Compiler

测试结果 1	<div><h3>foutput</h3><div>1 errors in x0 program</div><div>Array or variable not assigned!</div><div>&lt; &gt;</div></div> <p>即数组元素未赋值就使用出错处理</p>
修改后源程序 2	<pre>main{ int i; int p[20]; int now; i=now=p[0]=1; while(i&lt;=10){     now=now*i;     p[i]=now;     i=i+1; } while(i&gt;0){     i=i-1;     write p[i]; } /*删除这行*/ }</pre>
测试输入 2	无


Project Name/Model: X0\_Define\_Compiler

测试结果 2	<div><h3>fresult</h3><div><div>Start x0</div><div>3628800</div><div>362880</div><div>40320</div><div>5040</div><div>720</div><div>120</div><div>24</div><div>6</div><div>2</div><div>1</div><div>1</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试是否通过	PASS

Test No.	5	test_5.txt
测试目标	测试字符型变量的输出、for 循环、for 循环表达式缺失的情况	
测试用例的源程序	<pre>main{ int i; int j; char c; c=97; for(i=0;i&lt;10;i=i+1){     for(j=0;j&lt;10;){         if(j&gt;i)             write(10*i+j);     } }</pre>	



Project Name/Model: X0\_Define\_Compiler

	<pre>        else             write(j);             j=j+1;         }     }     write(c); }</pre> <p>/*遍历 0-99, 若为两位数输出 1, 对于个位数大于十位数的数输出其本身, 否则只输出个位*/</p>
测试输入	无
测试结果	<p><b>fresult</b></p>  <p>由于输出结果过长, 结果截图只截取了一部分, 完整输出如下:</p> <p>Start x0</p> <p>0</p> <p>1</p> <p>2</p> <p>3</p> <p>4</p> <p>5</p> <p>6</p> <p>7</p> <p>8</p> <p>9</p> <p>0</p> <p>1</p> <p>12</p>

Project Name/Model: X0\_Define\_Compiler

	13
	14
	15
	16
	17
	18
	19
	0
	1
	2
	23
	24
	25
	26
	27
	28
	29
	0
	1
	2
	3
	34
	35
	36
	37
	38
	39
	0
	1
	2
	3
	4
	45
	46

Project Name/Model: X0\_Define\_Compiler

	47
	48
	49
	0
	1
	2
	3
	4
	5
	56
	57
	58
	59
	0
	1
	2
	3
	4
	5
	6
	67
	68
	69
	0
	1
	2
	3
	4
	5
	6
	7
	78
	79
	0

Project Name/Model: X0\_Define\_Compiler

	<pre>1 2 3 4 5 6 7 8 89 0 1 2 3 4 5 6 7 8 9 a End x0  foutput ===Parsing success!===</pre>
测试是否通过	PASS

Test No.	6	test_6.txt
测试目标	测试负整型值的输入、求余运算符%、switch case 语句	

Project Name/Model: X0\_Define\_Compiler

测试用例的源程序	<pre>main{     int a;     int b;     read a;     if(a&lt;0)         b=1;     else{         if(a&lt;100)             b=a%3;         else             b=a%4;     }     switch(b){         case 1: write 1;break;         case 2: write 2;break;         case 3: write 3;break;         default: write 0;     } }</pre> <p>/*读入一个数 a,a 小于 0: 输出 1, a 大于 0 小于 100: 输出除以 3 的余数, a 大于 100: 输出 a 除以 4 的余数*/</p>
测试输入 1	-3

Project Name/Model: X0\_Define\_Compiler

测试结果 1	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? -3</div><div>1</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试输入 2	15
测试结果 2	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 15</div><div>0</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试输入 3	123

Project Name/Model: X0\_Define\_Compiler

测试结果 3	<div>fresult</div> <div> Start x0  Input? 123  3  End x0 </div> <div>foutput</div> <div> ===Parsing success!=== </div>
测试是否通过	PASS

Test No.	7	test_7(1).txt
测试目标	测试 repeat until 语句、自增自减运算符	
测试用例的源程序	<pre> main{   int i;   int p[10];   i=0;   repeat{     p[i]=i;     ++i;   }until(i==10)   --i;   repeat{     write p[i];     --i;   }until(i&lt;0) }  /*输出 0~9;可根据语法定义将自增/自减符号移动至变量后面*/ </pre>	

Project Name/Model: X0\_Define\_Compiler

测试输入	无
测试结果	<div><h3>fresult</h3><div><div>Start x0</div><div>9</div><div>8</div><div>7</div><div>6</div><div>5</div><div>4</div><div>3</div><div>2</div><div>1</div><div>0</div><div>End x0</div></div></div> <div><h3>foutput</h3><div>===Parsing success!===</div></div>
修改后源程序 1	<pre>main{ int i; int p[10]; i=0; repeat{ p[i]=i; i++; }while(i==10) i--; repeat{ write p[i]; i--; }while(i&lt;0) } /*将自增自减符号移到变量右边*/</pre>

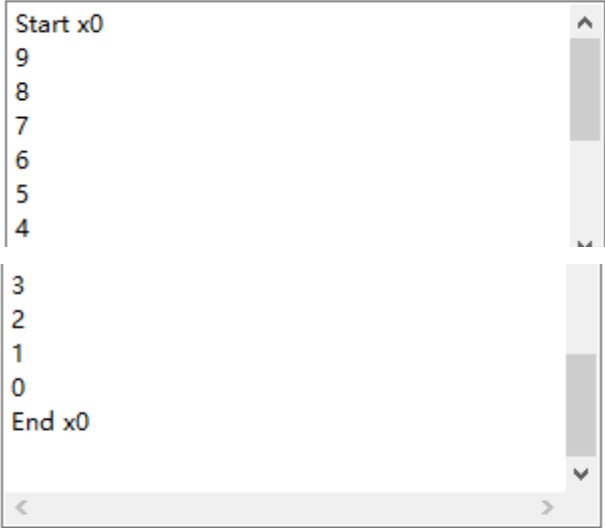
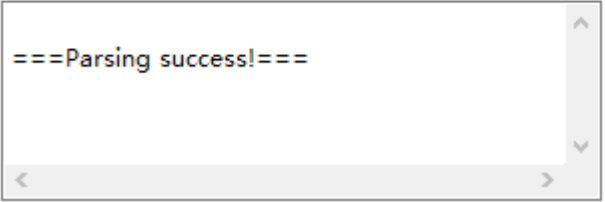


Project Name/Model: X0\_Define\_Compiler

测试输入 1	无
测试结果 1	<div><h3>fresult</h3><div><div>Start x0</div><div>9</div><div>8</div><div>7</div><div>6</div><div>5</div><div>4</div><div>3</div><div>2</div><div>1</div><div>0</div><div>End x0</div></div></div> <div><h3>foutput</h3><div>===Parsing success!===</div></div>
测试是否通过	PASS

Test No.	8	test_7(2).txt
测试目标	测试 repeat until 语句、自增自减运算符	
测试用例的源程序	<pre>main{ int i; int p[10]; i=0; repeat{ p[i]=i; }while(++i==10) --i; repeat{</pre>	

Project Name/Model: X0\_Define\_Compiler

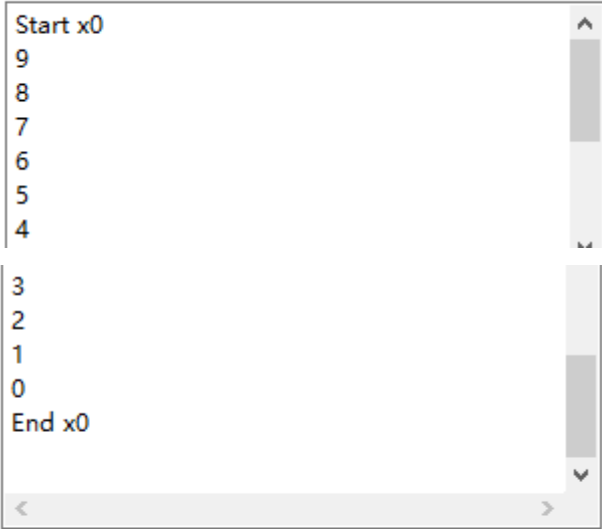
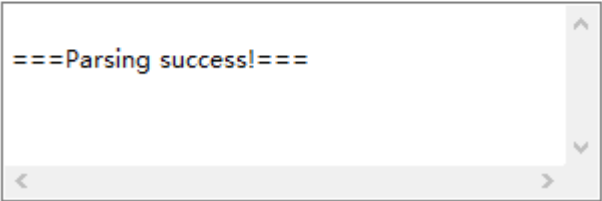
	<pre>write p[i]; }until(--i&lt;0) } /*输出 0~9;可根据语法定义将自增/自减符号移动至变量后面*/</pre>
测试输入	无
测试结果	<div><h3>fresult</h3><h3>foutput</h3></div>
修改后源程序 1	<pre>main{ int i; int p[11];/*改为 11*/ i=0; repeat{ p[i]=i; }until(i++==10)/*自增符号移到右边*/ i--;/*自减符号移到右边*/ repeat{ write p[i];</pre>

Project Name/Model: X0\_Define\_Compiler

	<pre> }until(i--&lt;1) /*自减符号移到右边, 1 改为 0*/ } /*将自增自减符号移到变量右边, 且调整数值防止数组越界或未赋值就引用*/ </pre>
测试输入 1	无
测试结果 1	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试是否通过	PASS


Test No.	9	test_7(3).txt
测试目标	测试 repeat until 语句、自增自减运算符	
测试用例的源程序	<pre> main{ int i; int p[10]; i=-1; i++; </pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>repeat{ p[i]=i; }while(++i==10) --i; repeat{ write p[i]; }while(i--==0) } /*输出 0~9*/</pre>
测试输入	无
测试结果	<div><h3>fresult</h3><h3>foutput</h3></div>
测试是否通过	PASS

Test No.	10	test1.txt
测试目标	测试嵌套 for 循环、求余%运算符、for break 语句	

Project Name/Model: X0\_Define\_Compiler


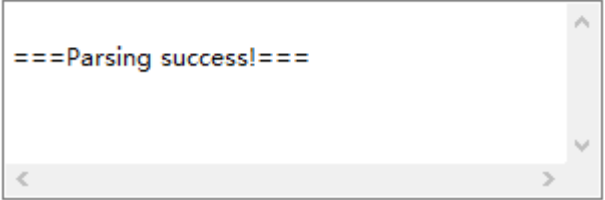
测试用例的源程序	<pre>main{     int num;     int i;     int flag;     for (num = 2; num &lt;= 100; num++) {         flag = 0;         for (i = 2; i &lt;= num / 2; i++) {             if ((num % i) == 0) {                 flag = 1;                 break;             }         }         if (flag == 0)             write num;     } }/*输出 1-100 之间的素数*/</pre>
测试输入	无
测试结果	<p><b>fresult</b></p>  <p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p> <pre>Start x0 2 3 5 7 11 13 17</pre>

Project Name/Model: X0\_Define\_Compiler

	<pre> 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 End x0  foutput ===Parsing success!=== </pre>
测试是否通过	PASS

Test No.	11	test2.txt
测试目标	测试 read 语句、write 语句、while 循环	
测试用例的源程序	<pre> main{     int a; </pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>int b; int c; int m; int n; read a; read b; m = a; n = b; while (b != 0) {     c = a % b;     a = b;     b = c; } write (m * n / a); } /*求两个整数的最小公倍数*/</pre>
测试输入	6 8
测试结果	<p><b>fresult</b></p>  <p>Start x0 Input? 6 Input? 8 24.0 End x0</p> <p><b>foutput</b></p>  <p>===Parsing success!===</p>
测试是否通过	PASS

Project Name/Model: X0\_Define\_Compiler

Test No.	12	test3.txt
测试目标	测试各种不同的 write 语句能否正常输出	
测试用例的源程序	<pre>main {     int x;     char y;     int i;     int p[20];     const int a = 37;     read x;     read y;     for(i = 0; i &lt; 20; i++)         p[i] = i;     i = 5;     write 10;     write (3 + 10);     write (x + 10);     write x;     write y;     write p[0];     write p[i];     write p[3 + 10];     write p[i + 10];     write p;     write a; }</pre>	
测试输入	5 a	



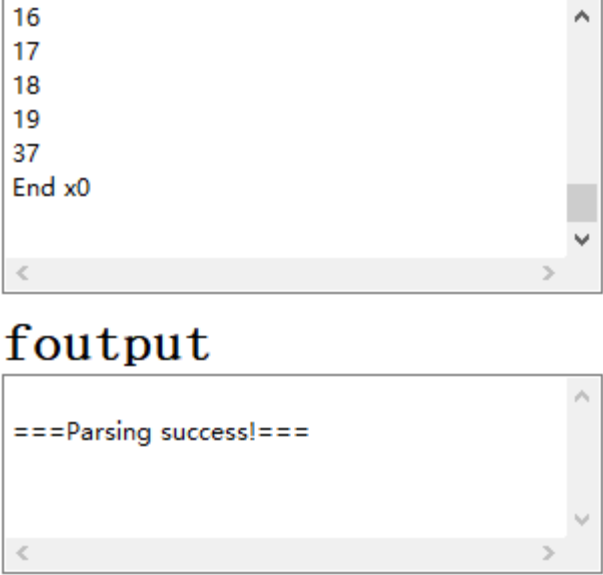
Project Name/Model: X0\_Define\_Compiler

测试结果

# fresult

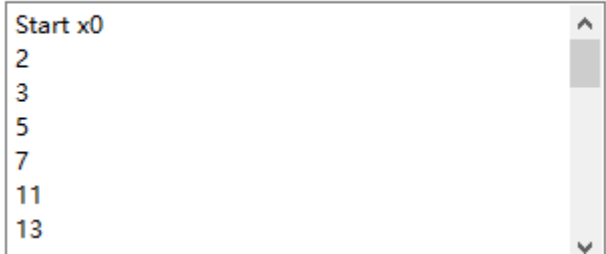
Start x0	^
Input? 5	
Input? a	
10	
13	
15	
5	^
a	^
0	
5	
13	
15	
0	
1	^
2	^
3	
4	
5	
6	
7	
8	^
9	^
10	
11	
12	
13	
14	
15	^

Project Name/Model: X0\_Define\_Compiler

	
测试是否通过	PASS

Test No.	13	test4.txt
测试目标	测试 for 嵌套 while 循环、while break 语句	
测试用例的源程序	<pre> main{     int num;     int i;     int flag;     for (num = 2; num &lt;= 100; num++) {         flag = 0;         i = 2;         while (i &lt;= num / 2) {             if ((num % i) == 0) {                 flag = 1;                 break;             }             i++;         }         if (flag == 0)             write num;     } } </pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>} }/*输出 1-100 之间的素数*/</pre>
测试输入	无
测试结果	<p><b>fresult</b></p>  <p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p> <p>Start x0</p> <pre>2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73</pre>

Project Name/Model: X0\_Define\_Compiler

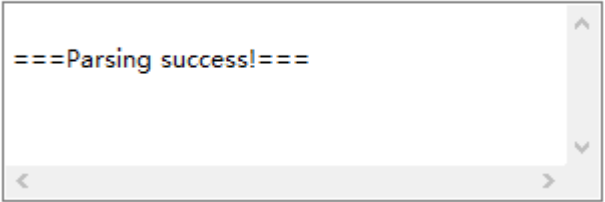
	<pre> 79 83 89 97 End x0  foutput ===Parsing success!=== </pre>
测试是否通过	PASS

Test No.	14	test5.txt
测试目标	测试 while 嵌套 repeat until 语句、负数的赋值与读入	
测试用例的源程序	<pre> main {     int i;     int j;     int k;     read i;     read j;     while (i &lt; j) {         k = -5;         repeat {             write k;             k++;         } until (k == i)         i++;         write i;     } } </pre>	

Project Name/Model: X0\_Define\_Compiler

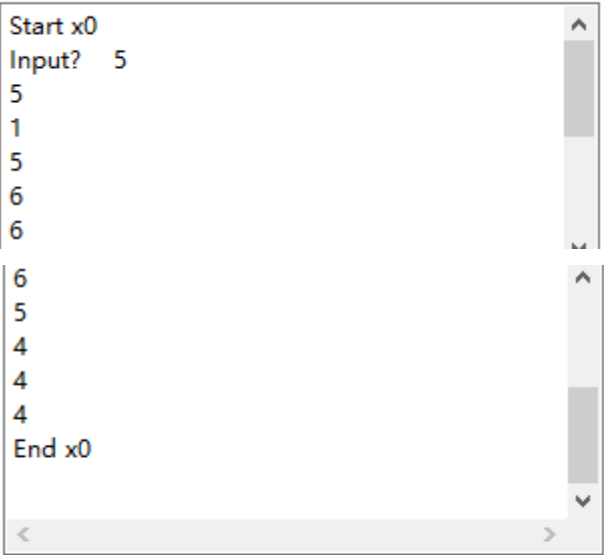
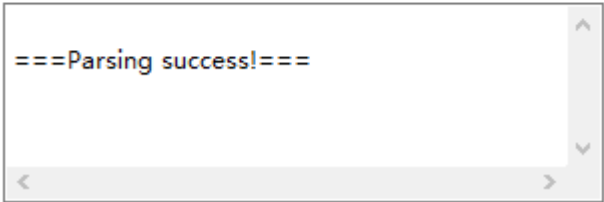
测试输入	-1 3
测试结果	<div><div>fresult</div><div><div>Start x0</div><div>Input? -1</div><div>Input? 3</div><div>-5</div><div>-4</div><div>-3</div><div>-2</div></div></div> <p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p> <div>Start x0</div> <div>Input? -1</div> <div>Input? 3</div> <div>-5</div> <div>-4</div> <div>-3</div> <div>-2</div> <div>0</div> <div>-5</div> <div>-4</div> <div>-3</div> <div>-2</div> <div>-1</div> <div>1</div> <div>-5</div> <div>-4</div> <div>-3</div> <div>-2</div> <div>-1</div> <div>0</div> <div>2</div> <div>-5</div> <div>-4</div> <div>-3</div>

Project Name/Model: X0\_Define\_Compiler

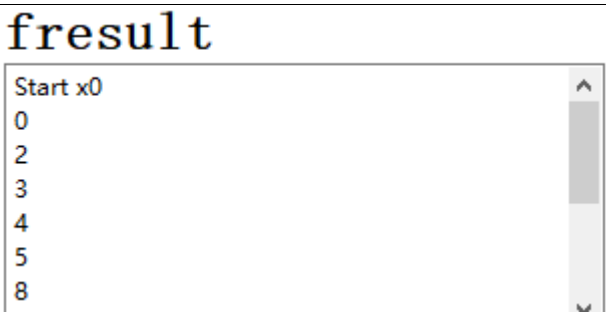
	<pre>-2 -1 0 1 3 End x0</pre> <p><b>foutput</b></p> 
测试是否通过	PASS

Test No.	15	test6.txt
测试目标	测试 x++、x--、++x、--x 的区别、判断奇偶运算符 ODD	
测试用例的源程序	<pre>main {     int x;     int copy;     read x;     write x;     write (ODD x);      copy = x;     write (x++);     write x;      x = copy;     write (++x);     write x;</pre>	

Project Name/Model: X0\_Define\_Compiler

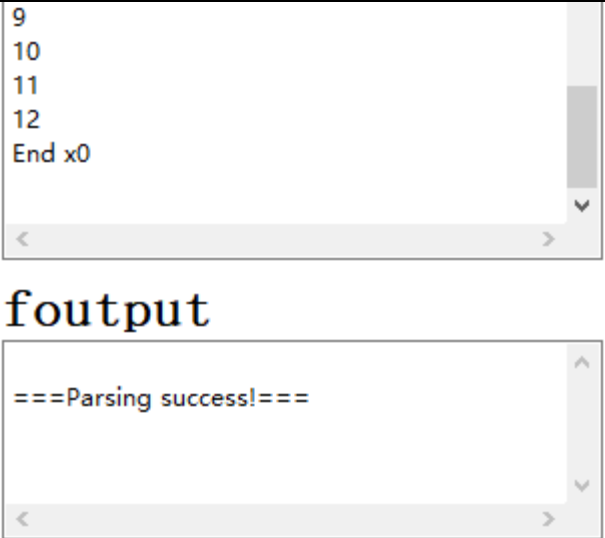
	<pre>x = copy; write (x--); write x;  x = copy; write (--x); write x;  }</pre>
测试输入	5
测试结果	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试是否通过	PASS

Project Name/Model: X0\_Define\_Compiler

Test No.	16	test7.txt
测试目标	测试 do while 循环、、 exit 语句、 if else 嵌套	
测试用例的源程序	<pre>main {     int i;     int j;     i = 0;     do {         if (i==0)             write i;         else {             if (i &lt; 5)                 write (i + 1);             else                 write (i + 3);         }         i++;     } while (i &lt; 10)     exit();     write 101; }</pre>	
测试输入	无	
测试结果		



Project Name/Model: X0\_Define\_Compiler

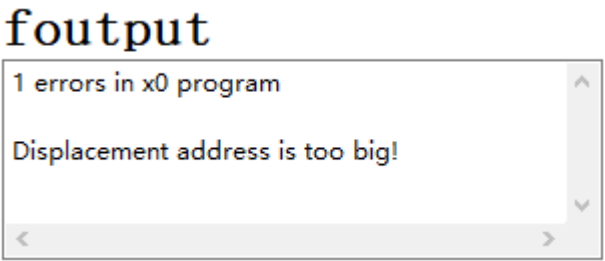
	
测试是否通过	PASS

Test No.	17	test8.txt
测试目标	for 语句表达式有缺失的情况、bool 类型及 bool 数组的输入输出	
测试用例的源程序	<pre>main {     int i;     bool a;     bool p[10];     i = 0;     for ( ; i &lt; 10; ) {         i++;         write i;     }     read a;     for (i = 0; i &lt; 10; i++) {         p[i] = i % 2;     }     write a;     write 101;     write p; }</pre>	

Project Name/Model: X0\_Define\_Compiler

测试输入	true
测试结果	<div><h3>fresult</h3><div><div>Start x0</div><div>1</div><div>2</div><div>3</div><div>4</div><div>5</div><div>6</div><div>7</div><div>8</div><div>9</div><div>10</div><div>Input? true</div><div>true</div><div>101</div><div>false</div><div>true</div><div>false</div><div>true</div><div>false</div><div>true</div><div>false</div><div>true</div><div>false</div><div>End x0</div></div><div><h3>foutput</h3><div>===Parsing success!===</div></div></div>
测试是否通过	PASS

Project Name/Model: X0\_Define\_Compiler

Test No.	18	test9.txt
测试目标	int、char、bool 类型输入值越界或错误、设定值越界的情况	
测试用例的源程序	<pre>main {     int a;     int x;     char b;     char y;     bool c;     bool z;      read a;     write a;     read b;     write b;     read c;     write c;      x = 10101010101;     y = 128;     z = 2;     write x;     write y;     write z; }</pre>	
测试输入	10101010101 12 1	
测试结果	 <p>The screenshot shows a window titled 'foutput'. Inside, there is a text area with the message '1 errors in x0 program' and 'Displacement address is too big!'. The window has a scroll bar on the right and a status bar at the bottom with left and right arrow icons.</p>	

Project Name/Model: X0\_Define\_Compiler

修改后源程序 1	<pre>main {      int a;     int x;     char b;     char y;     bool c;     bool z;      read a;     write a;     read b;     write b;     read c;     write c;      x = 1010; /*改为 1010*/     y = 128;     z = 2;     write x;     write y;     write z; }</pre>
测试输入 1	<pre>10101010101 12 1</pre>

Project Name/Model: X0\_Define\_Compiler

测试结果 1	<div><h3>fresult</h3><div>Start x0</div></div> <div><h3>foutput</h3><div>2 errors in x0 program</div><div>The range of type 'int' is -2147483648 ~ 21474</div><div>Array or variable not assigned!</div></div> <p>完整输出如下：</p> <p>2 errors in x0 program</p> <p>The range of type 'int' is -2147483648 ~ 2147483647!</p> <p>Array or variable not assigned!</p>
测试输入 2	1010 12 1

Project Name/Model: X0\_Define\_Compiler

测试结果 2	<div><h3>fresult</h3><div>Start x0 Input? 1010 1010</div></div> <div><h3>foutput</h3><div>2 errors in x0 program  The type of input should be 'char'! Array or variable not assigned!</div></div>
测试输入 3	1010 c 1
测试结果 3	<div><h3>fresult</h3><div>Start x0 Input? 1010 1010 Input? c c</div></div> <div><h3>foutput</h3><div>2 errors in x0 program  The type of input should be 'bool'! Please input 'true' or 'false'!</div></div>
测试输入 4	1010 c true

Project Name/Model: X0\_Define\_Compiler

测试结果 4	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 1010</div><div>1010</div><div>Input? c</div><div>c</div><div>Input? true</div><div>true</div><div>1010</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>2 errors in x0 program</div><div>The range of type 'char' is -128 ~ 127!</div><div>The range of type 'bool' is 0 ~ 1!</div></div></div>
修改后源程序 2	<pre>main {     int a;     int x;     char b;     char y;     bool c;     bool z;      read a;     write a;     read b;     write b;     read c;     write c;      x = 1010;     y = 97; /*改为 97*/</pre>

Project Name/Model: X0\_Define\_Compiler

	<pre>z = 2; write x; write y; write z; }</pre>
测试输入 5	<pre>1010 c true</pre>
测试结果 5	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 1010</div><div>1010</div><div>Input? c</div><div>c</div><div>Input? true</div><div>true</div></div><div><div>1010</div><div>a</div><div>End x0</div></div></div> <div><h3>foutput</h3><div>1 errors in x0 program</div><div>The range of type 'bool' is 0 ~ 1!</div></div>



Project Name/Model: X0\_Define\_Compiler

	<pre>write a; read b; write b; read c; write c;  x = 1010; y = 97; z = 0; /*改为 0*/ write x; write y; write z; }</pre>
测试输入 6	<pre>1010 c true</pre>
测试结果 6	<p><b>fresult</b></p>  <p>The screenshot shows a window titled 'fresult' with a scrollable text area. The text inside is as follows:</p> <pre>Start x0 Input?  1010 1010 Input?  c c Input?  true true  1010 a false End x0</pre> <p><b>foutput</b></p>  <p>The screenshot shows a window titled 'foutput' with a scrollable text area. The text inside is as follows:</p> <pre>===Parsing success!===</pre>

Project Name/Model: X0\_Define\_Compiler

测试是否通过	PASS
--------	------

Test No.	19	test10.txt
测试目标	and、or、not、XOR 的测试	
测试用例的源程序	<pre>main {     int a;     int b;     read a;     read b;      if (a &gt; 3 and b &gt; 3)         write 1;     else         write 0;      if (a &gt; 3 or b &gt; 3)         write 11;     else         write 10;      if (a &gt; 3 XOR b &gt; 3)         write 21;     else         write 20;      if (not a and b &gt; 3)         write 31;     else         write 30; }</pre>	
测试输入 1	4 5	

Project Name/Model: X0\_Define\_Compiler

测试结果 1	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 4</div><div>Input? 5</div><div>1</div><div>11</div><div>20</div><div>30</div><div>End x0</div></div><div></div></div> <div><h3>foutput</h3><div>===Parsing success!===</div><div></div></div>
测试输入 2	2 5
测试结果 2	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 2</div><div>Input? 5</div><div>0</div><div>11</div><div>21</div><div>30</div><div>End x0</div></div><div></div></div> <div><h3>foutput</h3><div>===Parsing success!===</div><div></div></div>
测试输入 3	4

Project Name/Model: X0\_Define\_Compiler


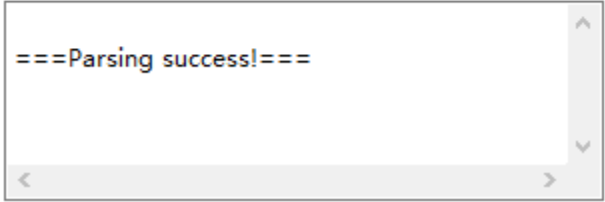
	2
测试结果 3	<div> <div>fresult</div> <div> <div>Start x0</div> <div>Input? 4</div> <div>Input? 2</div> <div>0</div> <div>11</div> <div>21</div> <div>30</div> <div>End x0</div> </div> </div> <div> <div>foutput</div> <div> <div>===Parsing success!===</div> </div> </div>
测试输入 4	2 1
测试结果 4	<div> <div>fresult</div> <div> <div>Start x0</div> <div>Input? 2</div> <div>Input? 1</div> <div>0</div> <div>10</div> <div>20</div> <div>30</div> </div> </div>

Project Name/Model: X0\_Define\_Compiler

	<div>End x0</div> <div>foutput</div> <div>===Parsing success!===</div>
测试输入 5	<div>0</div> <div>5</div>
测试结果 5	<div>fresult</div> <div> Start x0  Input? 0  Input? 5  0  11  21  31  End x0 </div> <div>foutput</div> <div>===Parsing success!===</div>
测试是否通过	PASS

Test No.	20	test11.txt
测试目标	switch case 有没有 default、switch case 嵌套	
测试用例的源程序	<pre>main {     int i;</pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>int j; read i; read j; switch (i + 1) {     case 1 : write 1; break;     case 2 : write 2; break;     case 3 : {         switch (j * 3) {             case 3 : write 3; break;             default : write 4;         }     } break; }</pre>
测试输入 1	0 1
测试结果 1	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试输入 2	1 1

Project Name/Model: X0\_Define\_Compiler

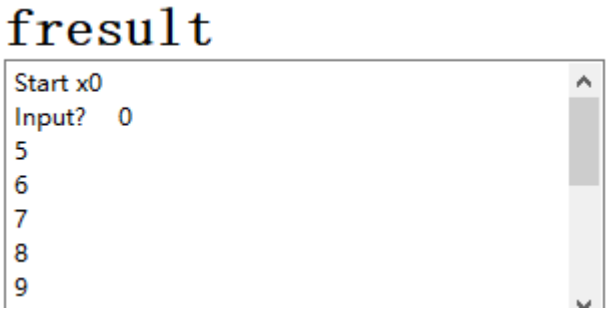
测试结果 2	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 1</div><div>Input? 1</div><div>2</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试输入 3	2 1
测试结果 3	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 2</div><div>Input? 1</div><div>3</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试输入 4	2 2

Project Name/Model: X0\_Define\_Compiler

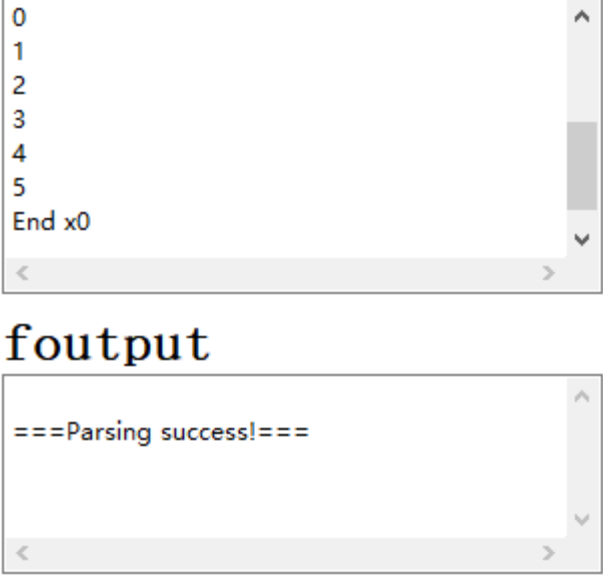
测试结果 4	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 2</div><div>Input? 2</div><div>4</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试输入 5	3 1
测试结果 5	<div><h3>fresult</h3><div><div>Start x0</div><div>Input? 3</div><div>Input? 1</div><div>End x0</div></div></div> <div><h3>foutput</h3><div><div>===Parsing success!===</div></div></div>
测试是否通过	PASS



Project Name/Model: X0\_Define\_Compiler


Test No.	21	test12.txt
测试目标	while continue 和 for continue 语句	
测试用例的源程序	<pre>main {     int i;     int j;     read i;     while (i &lt; 10) {         if (i &lt; 5) {             i++;             continue;         }         write i;         i++;     }      for (j = 0; j &lt; 10; ) {         if (j &gt; 5) {             j++;             continue;         }         write j;         j++;     } }</pre>	
测试输入	0	
测试结果		

Project Name/Model: X0\_Define\_Compiler

	 <p>The screenshot shows a compiler window with two panes. The top pane contains the input code: <code>0</code>, <code>1</code>, <code>2</code>, <code>3</code>, <code>4</code>, <code>5</code>, and <code>End x0</code>. The bottom pane, titled "foutput", shows the output: <code>===Parsing success!===</code>.</p>
测试是否通过	PASS

Test No.	22	test13.txt
测试目标	while 嵌套含 continue	
测试用例的源程序	<pre> main {     int i;     int j;     i = 0;     while (i &lt; 10) {         write i;         j = i - 1;         while (j &gt; 0) {             if (j &gt; 5) {                 j--;                 continue;             }             write j;             j--;         }         i++;     } } </pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>        write 101;      } }</pre>
测试输入	无
测试结果	<div><h3>fresult</h3><p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p><pre>Start x0 0 101 1 101 2 1 101 3 2 1 101 4 3 2 1 101 5 4 3 2 1</pre></div>

Project Name/Model: X0\_Define\_Compiler

	101
	6
	5
	4
	3
	2
	1
	101
	7
	5
	4
	3
	2
	1
	101
	8
	5
	4
	3
	2
	1
	101
	9
	5
	4
	3
	2
	1
	101
	End x0

Project Name/Model: X0\_Define\_Compiler

	<div><b>foutput</b></div> <div>===Parsing success!===</div>
测试是否通过	PASS

Test No.	23	test14.txt
测试目标	for 嵌套含 continue	
测试用例的源程序	<pre>main {     int i;     int j;     for (i = 0; i &lt; 10; i++) {         write i;         for (j = i - 1; j &gt; 0; j--) {             if (j &gt; 5) {                 j--;                 continue;             }             write j;         }         write 101;     } }</pre>	
测试输入	无	
测试结果	<div><b>fresult</b></div> <div>Start x0 0 101 1 101 2 1</div>	

Project Name/Model: X0\_Define\_Compiler

由于输出结果过长，结果截图只截取了一部分，完整输出如下：

Start x0

0

101

1

101

2

1

101

3

2

1

101

4

3

2

1

101

5

4

3

2

1

101

6

5

4

3

2

1

101

7

5

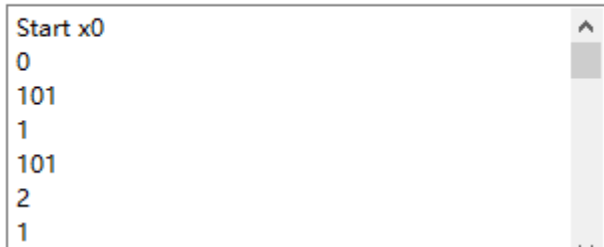
4

Project Name/Model: X0\_Define\_Compiler

	<pre> 3 2 1 101 8 5 4 3 2 1 101 9 5 4 3 2 1 101 End x0  foutput ===Parsing success!=== </pre>
测试是否通过	PASS

Test No.	24	test15.txt
测试目标	while 嵌套含 break	
测试用例的源程序	<pre> main {     int i; </pre>	

Project Name/Model: X0\_Define\_Compiler


	<pre>int j; i = 0; while (i &lt; 10) {     write i;     j = i - 1;     while (j &gt; 0) {         if (j &gt; 5)             break;         write j;         j--;     }     i++;     write 101; }</pre>
测试输入	无
测试结果	<p><b>fresult</b></p>  <p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p> <pre>Start x0 0 101 1 101 2 1 101 3 2</pre>



Project Name/Model: X0\_Define\_Compiler

	<pre>1 101 4 3 2 1 101 5 4 3 2 1 101 6 5 4 3 2 1 101 7 101 8 101 9 101 End x0  foutput ===Parsing success!===</pre>
测试是否通过	PASS

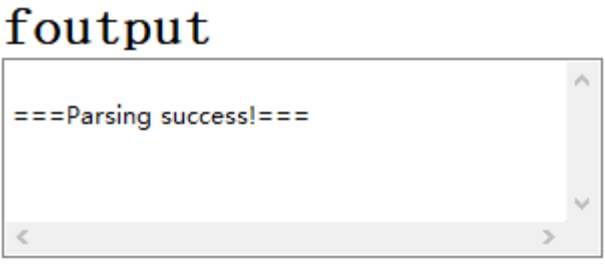
Project Name/Model: X0\_Define\_Compiler

Test No.	25	test16.txt
测试目标	for 嵌套含 break	
测试用例的源程序	<pre>main {     int i;     int j;     for (i = 0; i &lt; 10; i++) {         write i;         for (j = i - 1; j &gt; 0; j--) {             if (j &gt; 5)                 break;             write j;         }         write 101;     } }</pre>	
测试输入	无	
测试结果	<p><b>fresult</b></p>  <p>由于输出结果过长，结果截图只截取了一部分，完整输出如下：</p> <pre>Start x0 0 101 1 101 2 1 101 3</pre>	

Project Name/Model: X0\_Define\_Compiler


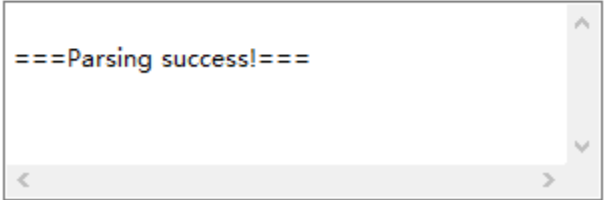
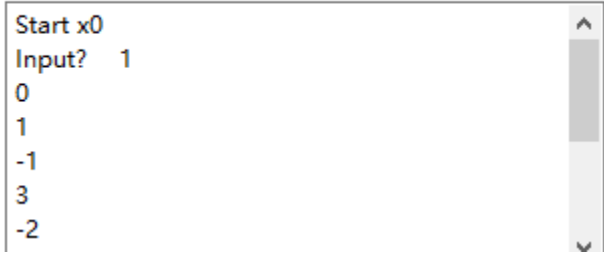
	2
	1
	101
	4
	3
	2
	1
	101
	5
	4
	3
	2
	1
	101
	6
	5
	4
	3
	2
	1
	101
	7
	101
	8
	101
	9
	101
	End x0

Project Name/Model: X0\_Define\_Compiler


	
测试是否通过	PASS

Test No.	26	test17.txt
测试目标	循环中含多个 continue、break 语句	
测试用例的源程序	<pre>main {     int i;     int j;     read j;     for (i = 0; i &lt; 10; i++) {         if (j == 0)             break;         if (i == 2) {             write -1;             i++;             continue;         }         if (i == 4) {             write -2;             i++;             continue;         }         write i;         if (i == 7) {             write 0;             break;         }     } }</pre>	

Project Name/Model: X0\_Define\_Compiler


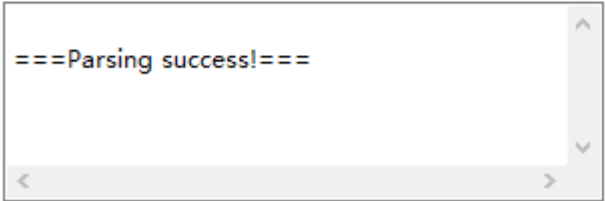
	<pre>} }</pre>
测试输入 1	0
测试结果 1	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试输入 2	1
测试结果 2	<p><b>fresult</b></p> 

Project Name/Model: X0\_Define\_Compiler

	 <p>The screenshot shows a code editor with the following text:</p> <pre>5 6 7 0 End x0</pre> <p>Below the editor, the output window displays:</p> <pre>foutput ===Parsing success!===</pre>
测试是否通过	PASS

Test No.	27	test18.txt
测试目标	测试过程嵌套、常量定义	
测试用例的源程序	<pre>main{     const int m = 5;     int a;     int b;     procedure g {         procedure h {             int i;             int x;             i = a;             x = (m + i) / (m - i);             if (x &lt; (a + 1))                 b = x;             write a;             write i;             write x;             write b;         }     } }</pre>	

Project Name/Model: X0\_Define\_Compiler

	<pre>a++; call h;  } a = 2; b = 1; call g;  }</pre>
测试输入	无
测试结果	<p><b>fresult</b></p>  <p><b>foutput</b></p> 
测试是否通过	PASS