

Compilers-PA1

叶嘉睿
20307130048

李元昊
20307130062

2023 年 3 月 4 日

Code

```
1 (*
2  *   CS164 Fall 94
3  *
4  *   Programming Assignment 1
5  *   Implementation of a simple stack machine.
6  *
7  *   Skeleton file
8  *)
9
10 --/root/assignments/PA1/stack.cl
11
12 class List {
13   isNil() : Bool { true };
14   head()  : String { { abort(); "" }; };
15   tail()  : List { { abort(); self; } };
16   newNode(i : String) : List {
17     (new Node).init(i, self)
18   };
19 };
20
21 class Node inherits List {
22   value : String; -- The element in this list cell
23   next  : List; -- The rest of the list
24   isNil() : Bool { false };
25   head()  : String { value };
26   tail()  : List { next };
27   init(i : String, rest : List) : List {
28     {
29       value <- i;
30       next <- rest;
31       self;
32     }
33   };
34 };
35
36 class Main inherits IO {
37   print_list(l : List) : Object {
38     if l.isNil() then out_string("")
```

```

39     else {
40         out_string(l.head());
41         out_string("\n");
42         print_list(l.tail());
43     } fi
44 };
45 stack: List;
46 str: String;
47 str1: String;
48 str2: String;
49 i: Int;
50 j: Int;
51 k: Int;
52 flag: Bool<-true;
53
54 main() : Object {
55     {
56         stack<-new List;
57         while flag loop
58         {
59             out_string(">");
60             str<-in_string();
61
62             -- out_string(str);
63             -- out_string("\n");
64             if str.length()=1 then {
65                 if str = "+" then
66                     stack<-stack.newNode(str)
67                 else if str="s" then
68                     stack<-stack.newNode(str)
69                 else if str="e" then
70                     {
71                         if (not stack.isNil()) then
72                             {
73                                 str<-stack.head();
74                                 if str="+" then{
75                                     stack<-stack.tail();
76                                     if (not stack.isNil()) then
77                                         {
78                                             str1<-stack.head();
79                                             stack<-stack.tail();
80                                             if (not stack.isNil()) then
81                                                 {
82                                                     str2<-stack.head();
83                                                     stack<-stack.tail();
84                                                     i<-(new A2I).c2i(str1);
85                                                     j<-(new A2I).c2i(str2);
86                                                     k<-i+j;
87                                                     str<-(new A2I).i2c(k);
88                                                     stack<-stack.newNode(str);
89                                                     } else stack<-stack.newNode(str1) fi;

```

```

90         } else stack fi;
91     }
92     else if str="s" then{
93         stack<-stack.tail();
94         if (not stack.isNil()) then {
95             str1<-stack.head();
96             stack<-stack.tail();
97             if (not stack.isNil()) then{
98                 str2<-stack.head();
99                 stack<-stack.tail();
100                 stack<-stack.newNode(str1);
101                 stack<-stack.newNode(str2);
102             } else stack<-stack.newNode(str1) fi;
103         } else stack fi;
104     }
105     else if str=" " then stack
106     else{
107         -- stack<-stack.newNode(str);
108         stack;
109     }
110     fi fi fi;
111 } else str fi;
112 }
113 else if str="d" then
114     print_list(stack)
115 else if str="x" then
116     flag<-false
117 else{
118     stack<-stack.newNode(str);
119 }
120 fi fi fi fi fi;
121 }
122 else flag<-false fi;
123 }
124 pool;
125 }
126 };
127 };

```

Sample Output

```
● root@vultr:~/Compilers-coolc/assignments/PA1# spim -file stack.s
SPIM Version 6.5 of January 4, 2003
Copyright 1990-2003 by James R. Larus (larus@cs.wisc.edu).
All Rights Reserved.
See the file README for a full copyright notice.
Loaded: ../lib/trap.handler
>1
>+
>2
>s
>d
s
2
+
1
>e
>e
>d
3
>x
COOL program successfully executed
```

图 1: 运行截图