

# PA1x报告

韩晓宇18300750006

刘明君18307130064

## 1 代码核心逻辑

```
class St {
  cont : String;
  next : St;
  setCont(z : String): Object {cont <- z};
  setNext(nx : St): Object {next <- nx};
  getCont(): String {cont};
  getNext(): St {next};
};

class Main inherits IO {
  z : String;
  head : St <- new St;
  aux : A2I <- new A2I;
  push ( s : String ): Object {{
    let tmp:St <- new St in {
      tmp.setCont(s);
      tmp.setNext(head);
      head <- tmp;
    };
  }};
  eval (): Object {{
    let tmp1:St, tmp2:St in {
      if head.getCont () = "+" then {
        tmp1 <- head.getNext();
        tmp2 <- head.getNext().getNext();
        tmp2.setCont ( aux.i2a ( aux.a2i(tmp1.getCont()) +
aux.a2i(tmp2.getCont()) ) );
        head <- tmp2;
      }
      else {
        if head.getCont () = "s" then {
          tmp1 <- head.getNext();
          tmp2 <- head.getNext().getNext();
          tmp1.setNext ( tmp2.getNext() );
          tmp2.setNext ( tmp1 );
          head <- tmp2;
        }
        else {
          0;
        }
        fi;
      }
      fi;
    };
  }};
  output ( now : St ): Object {{
```

```

        if now.getCont() = "End" then {
            0;
        }
        else {
            out_string ( now.getCont () );
            out_string ( "\n" );
            output ( now.getNext () );
        }
        fi;
    }
};

main() : Object {{
    z <- "z";
    head.setCont ( "End" );
    while not z = "x" loop {
        out_string(">");
        z <- in_string();
        if z = "+" then {
            push ( z );
        }
        else {
            if z = "s" then {
                push ( z );
            }
            else {
                if z = "e" then {
                    eval ();
                }
                else {
                    if z = "d" then {
                        output ( head );
                    }
                    else {
                        if z = "x" then {
                            -- break
                            out_string ("stop!");
                            out_string ("\n");
                            abort();
                        }
                        else {
                            push ( z );
                        }
                    }
                    fi;
                }
            }
            fi;
        }
        fi;
    }
    fi;
}
pool;
};
};

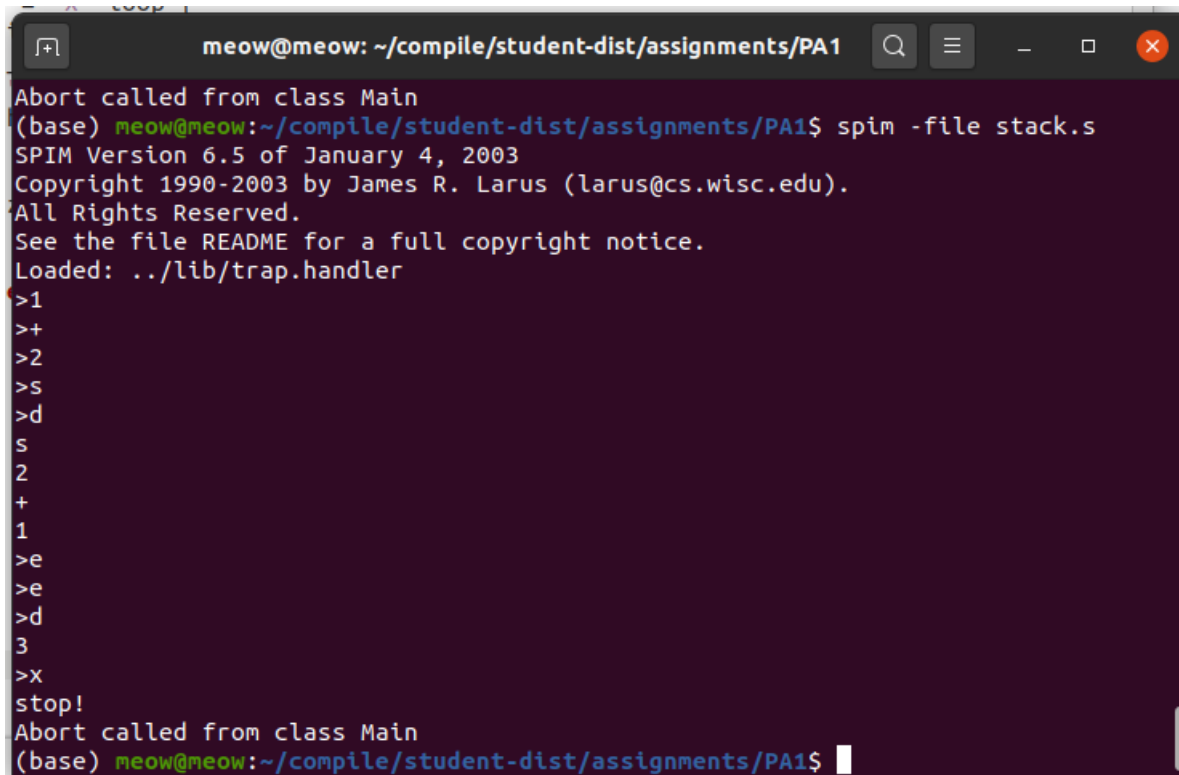
```

cool语言中有诸多不便之处，例如本次PA中需要实现一个栈，但是在cool中没有数组，所以需要使用链表来实现栈。代码中类型 `st` 即为实现的链表，其中存储了下一个节点和当前节点的内容。在栈中，操作符号和数字都用字符串的形式存储，只有当进行加法求值时才转化为整数类型计算。

同时cool中的类型属性不能从外部调用，需要使用面向对象的方法来获取和修改，如 `setName()` , `getName()` 等。

具体的操作只需要讨论当前的指令进行压栈或者获取栈顶的前若干元素进行处理。

## 2 代码实现结果



```
meow@meow: ~/compile/student-dist/assignments/PA1
Abort called from class Main
(base) meow@meow:~/compile/student-dist/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
stop!
Abort called from class Main
(base) meow@meow:~/compile/student-dist/assignments/PA1$
```