

システムソフトウェア特論 課題# 1-2

1931125 廣瀬海太

2020/02/04

1 演習 1-2

演習の a,b,d の課題について、機能を実装したコードを以下に載せる。次のサブセクションで工夫点などをコメントする。

```
1 import java.util.*;
2 import java.io.*;
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class Sam12 {
7     public static void main(String[] args) throws Exception {
8         Toknizer tok = new Toknizer(args[0]);
9         Scanner sc = new Scanner(System.in);
10        while(true) {
11            System.out.print("> ");
12            String line = sc.nextLine();
13            if(line.equals("p")) {
14                System.out.println(tok.curTok());
15            } else if(line.equals("e")) {
16                System.out.println(tok.isEof());
17            } else if(line.equals("r")) {
18                tok.read(); System.out.println(tok.curTok());
19            } else if(line.equals("f")) {
20                tok.fwd(); System.out.println(tok.curTok());
21            } else if(line.equals("b")) {
22                tok.back(); System.out.println(tok.curTok());
23            } else if(line.equals("push")) {
24                tok.push(); System.out.println(" read");
25            } else if(line.equals("q")) {
26                System.exit(0);
27            }
28        }
29    }
30 }
31 class Toknizer {
32     Scanner sc;
33     List<String> tok = new ArrayList<String>();
```

```

34   BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
35   int counter=-1;
36   int curpoint=-1;
37   boolean eof = false;
38   public Tokenizer(String fname) throws Exception {
39       sc = new Scanner(new FileInputStream(fname)); read();
40   }
41
42   public boolean isEOF() { return eof; }
43   public String curTok() { return tok.get(curpoint); }
44   public void read() {
45       if(!eof && sc.hasNext() && curpoint==counter) { tok.add(sc.next()); counter=c
46       else if(!eof && sc.hasNext() && curpoint<counter) {
47           if(curpoint>0&&curpoint<counter) {curpoint=curpoint+1;}
48           else{System.out.println("既に読み込まれた値を再出力しています");}
49       }else{
50           if(!eof){counter=counter+1;curpoint=curpoint+1;}
51           eof = true; tok.add("$");System.out.println("最後のトークンを読み込みました");
52       }
53   public void fwd() {
54       if(curpoint>0&&curpoint<counter) {curpoint=curpoint+1;}
55       else{System.out.println("既に読み込まれた値を再出力しています");}
56   }
57   public void back() {
58       if(curpoint>0) { curpoint=curpoint-1;}
59       else{System.out.println("一行目です");}
60   }
61
62   public void push(){
63       try {
64       // キーボード入力を受け付ける
65           tok.add(br.readLine());
66           counter=counter+1;curpoint=curpoint+1;
67       } catch (IOException e) {
68           System.out.println(e.getMessage());
69       }
70   }
71 }

```

1.1 工夫点

dまで実装することを考え大胆に変更を加えた. まず List クラスを使用し、読み込まれたテキストが逐次 List 型 tok に追加されるようにした.curpoint と counter という参照用の変数を用意し, curTok メソッドは tok リストの curpoint インデックス番号の要素を返す. また Tokenizer クラスに 4 つのメソッドを追加した.

read メソッド :未読のトークンを読み込む. それ以外はももとの fwd 関数と同じ機能

back メソッド :curpoint の値を 1 減らし, 表示を一つ戻す. 一行目に行くとメッセージのみ表示.

fwd メソッド :curpoint の値を 1 増やし, 表示を進める. 最新の行を表示した場合、read を行うまでメッセージのみ表示

push メソッド :最新行に空行以外のトークンを tok リストに追加する.

1.2 実行例

入力ファイルは以下

```
1 1
2 2
3 3
4 4
```

実行すると以下のように表示される.

```
1 java Sam12 z1.min
2 > p
3 1
4 > b
5 一行目です
6 1
7 > f
8 既に読み込まれた値を再出力しています
9 1
10 > r
11 2
12 > push
13 2.5
14 read
15 > r
16 3
17 > r
18 4
19 > b
20 3
21 > b
22 2.5
23 > f
24 3
25 > f
26 4
27 > f
28 既に読み込まれた値を再出力しています
29 4
30 > r
31 最後のトークンを読み込みました
32 $
```

2 感想等

2.1 Q1

プログラミングは好き/得意ですか、苦手ですか。それはなぜですか。どういうところが特にそう思う？

A. アルバイトで使う程度には得意です. 僕の収入源なので好きです.

2.2 Q2

さまざまなプログラミング言語があるということについてどのように考えていますか。

A. アルバイトでは C++, Python を使っていますが、それぞれ一長一短で補い合っている感じでいいと思います.

2.3 Q3

リフレクション (課題をやってみて気付いたこと)、感想、要望など。

A. 自由度があるので試行錯誤のしがいがある.