Shihang Zhang
161000875
Tuesday, February 10, 2015

Design

I use two pointers to manipulate the all operations which defined in the struct TokenzierT.

```
typedef struct TokenizerT_ {
    char *head, *tail;
}TokenizerT;
```

Pointer head will find the first character of a token and pointer tail marks the next character to the last character of the token. This operation will repeat until it reaches the end of this string. It is implemented in function below.

```
char *TKGetNextToken( TokenizerT * tk )
```

Explanation

My program follows the rule in tokenizer.pdf that every token is separated by one or several white spaces and each token is printed in one line with its identification. However, the examples in tokenizer.pdf is not always consistent with this rule. For example, the last token in this string is "[]" and the output should be leftbrace rightbrace "[]". If you want the output of "[]" in two lines like the pic, I think it should be separated by white space in the input string(aka. ./tokenizer "0x4356abdc 0777 [ ]").

- Input:

    ./tokenizer "0x4356abdc 0777 [] "

- Output

    hex constant "0x4356abdc"
    octal constant "0777"
    leftbrace "["
    rightbrace "]"

Complexity

The time complexity is O(n). n is the length of input string.