

# 2023 年度 言語処理課題 3

## 課題レポート

実験日：提出日：2024 年 2 月 1 日（木）

学修番号：22140003

氏名：佐倉仙汰郎

## 1 課題 1

```
sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> 3 + 5 * 10
(3 + (5 * 10))
```

図 1

```
const (
    _ int = iota
    LOWEST
    EQUALS      // ==
    LESSGREATER // > or <
    PRODUCT     // +
    SUM         // *
    PREFIX      // -X or !X
    CALL        // myFunction(X)
)
```

図 2

```
sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code_1
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> 3 + 5 * 10
((3 + 5) * 10)
>>
```

図 3

## 2 課題 2

```
>> sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> -add(1,2)
(-add(1, 2))
```

図 4

```
sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> -add(1,2)
(-add)(1, 2)
>>
```

図 5

### 3 課題3

```
>> sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> +3

      _ _ _ _ _
     /         \
    /             \
   /               \
  /                 \
 /                   \
/                     \
| 0 | 0 |
|___|___|
  \                   /
   \                 /
  \                 /
   \               /
    \             /
     \         /
      _ _ _ _ _

Woops! We ran into some monkey business here!
parser errors:
  no prefix parse function for + found
>>
```

図 6

```
p.prefixParseFns = make(map[token.TokenType]prefixParseFn)
p.registerPrefix(token.IDENT, p.parseIdentifier)
p.registerPrefix(token.INT, p.parseIntegerLiteral)
p.registerPrefix(token.BANG, p.parsePrefixExpression)
p.registerPrefix(token.MINUS, p.parsePrefixExpression)
p.registerPrefix(token.TRUE, p.parseBoolean)
p.registerPrefix(token.FALSE, p.parseBoolean)
p.registerPrefix(token.LPAREN, p.parseGroupedExpression)
p.registerPrefix(token.IF, p.parseIfExpression)
p.registerPrefix(token.FUNCTION, p.parseFunctionLiteral)

(p.registerPrefix(token.PLUS, p.parsePrefixExpression))
```

図 7

```
>> sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> +3
(+3)
>>
```

図 8

### 4 課題4

```
sentooooooooon@Sentarodell:~/TMU/2/Language_analysis/waiig_code
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> -5 + 10
(-(5 + 10))
```

図 9

```
Hello sentooooooooon! This is the Monkey programming language!
Feel free to type in commands
>> -5 + 10
(-(5 + 10))
```

図 10