

mini ML
interpreter
in C

10th 修了生 飯田 圭祐

ML = Meta
Language

```
let a' = 10 in
```

```
let b' = 20 in
```

```
  stdout.(a' + b' * a' - b');
```

```
(\x -> \y -> \z ->
  x * y * z) .100.200.300;
```

```
let add = \x -> \y -> x + y in
let inc = add.1 in
stdout.(inc.(inc.(inc.(inc.(inc.(inc.(inc.(inc.0))))))));
```

```
let Y = \f -> (\x -> f.(x.x)).(\x -> f.(x.x)) in
let factgen = \f -> \n -> if.(eq.n.0).1.(n * (f.(n-1))) in
let fact = Y.factgen in
stdout.(fact.6);
```

```
let True = \t -> \f -> t in True;
let False = \t -> \f -> f in False;
let if = \bool -> \p -> \q -> bool.p.q in if;

let eq = \n -> \m -> match n {
    m => True |
    _ => False
} in eq;
```

Lexer: REからDFAを構築

```
def(l, "!", symex);
def(l, "?", symquestion);
def(l, ".", symdot);
def(l, ":", symcolon);
def(l, ";", symsemi);
def(l, "(", symlparen);
def(l, ")", symrparen);
def(l, "=", symassign);
def(l, "==", symeq);
def(l, "=>", symfatarr);
def(l, "|", symbar);
def(l, "!=", symneq);
def(l, "{", symlbrace);
def(l, "}", symrbrace);
def(l, "\\", symbs);
defr(l, "[abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ_][abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789);
defr(l, "0x[0123456789abcdefABCDEF]+", hex);
defr(l, "0|[123456789][0123456789]*", nat);
```

Parser: LL(1)

1引数のLambda式

関数適用

四則演算

let式

match式

Exec: 抽象構文木の実行

名前呼び戦略による関数適用

github.com/k-mrm/ml