

## 課題3

---

### ソースコード

```
%{
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define STACKSIZE 256

double S[STACKSIZE], x;
int P = -1;

// 変数用の構造体
struct stab {
    char name;
    double val;
} stab[100];
int count=0;

void printStack();
void assignment(char name, double val);
void push(char name);
void listing();
}%

%option noyywrap
DIGIT [0-9]

%%
[ \t\r]+      { /* ignored */ }
\n            { printStack(); }
"-"?{DIGIT}+("."{DIGIT}*)? { sscanf(yytext, "%lf", &S[++P]); }
"-"?"."{DIGIT}+ { sscanf(yytext, "%lf", &S[++P]); }
"+"          { if (P>0) { S[P-1] += S[P]; --P; } }
"- "         { if (P>0) { S[P-1] -= S[P]; --P; } }
"*"          { if (P>0) { S[P-1] *= S[P]; --P; } }
"/"          { if (P>0) { S[P-1] /= S[P]; --P; } }
"+- "        { if (P>-1) S[P] = -S[P]; }
"? "         { puts("p スタックの先頭を除去する。(droP or Pop)");
               puts("d スタックの先頭を複製する。(Dupricate)");
               puts("o スタックの2番目の要素を複製する。(Over)");
               puts("s スタックの先頭と2番目を入れ替える。(Swap)");
               puts("r スタック冒頭の3要素を順に入れ替える。(Rotate)");
               puts("c スタックを全てクリアして、空にする。(Clear)");
               puts("q プログラムを終了する。(Quit)"); }
[A-Z]("=")    { if (P>-1) { assignment(yytext[0], S[P]); } }
[A-Z]         { push(yytext[0]); }
v            { listing(); }
p            { if (P>-1) --P; }
```

```

d      { if (P>-1) { ++P; S[P] = S[P-1]; } }
o      { if (P>0) { ++P; S[P] = S[P-2]; } }
s      { if (P>0) { x=S[P-1]; S[P-1]=S[P]; S[P]=x; } }
r      { if (P>1) { x=S[P-2]; S[P-2]=S[P-1]; S[P-1]=S[P]; S[P]=x;
} }
c      { P=-1; }
q      { return; }
[^ \t\r\n]+ { fprintf(stderr, "unknown token (%s)\n", yytext); }
%%

main() {
    printf("*** Reverse Polish Calculator \n");
    printf("> ");
    yylex();
}

void printStack() {
    int i;
    for(i=0; i<=P; i++)
        printf("%.8g ", S[i]);
    printf("> ");
}

// 変数への値の代入
void assignment(char name, double val) {
    int i;
    for(i=0; i<count; i++) {
        if(stab[i].name==name) {
            stab[i].val = val;
            return;
        }
    }
    stab[count].name = name;
    stab[count++].val = val;
}

// 変数の値をスタックの先頭へプッシュ
void push(char name) {
    int i;
    for(i=0; i<count; i++) {
        if(stab[i].name==name) {
            S[++P] = stab[i].val;
            return;
        }
    }
}

// 変数名とその値を列挙
void listing() {
    int i;
    for(i=0; i<count; i++)
        printf("%c = %f\n", stab[i].name, stab[i].val);
}

```

```
}
```

## 説明

変数機能を構造体を用いて実装した。

スタックの先頭へのプッシュと変数の列挙の機能を追加した。

今回は変数名は大文字アルファベット一文字のみに限ったが、char型で変数名を定義しているところをchar\*などにすればその他の変数名でも使用できると思う。