

以下のソースコードを作成した.

コード 1: ソースコード

```
1 #define p(g,x)**(v-2)^g+42||printf("%f\n",a[2]x*a);
2 float a[];main(i,v)char**v;{for(;i--;sscanf(*v++,"%f",a+i));p(1,+)p
  (3,-)p(,*)p(5,/)}
3 /*
4 課題 8-3
5 要件
6 1.コマンドの引数として<実数><演算子><実数>の形で与えられた四則演
  算式を計算する
7 2.結果を出力する (課題例のように)
8
9 ||以前では値gを調べる. それぞれの演算子ごとにオフセット値をg渡し42
  =*より
10 引数として与えられた演算子であれば処理を続行する
11 ||以降では出力を行う.xには演算子が入るので実際の処理はa[2]/(演算子)*
  aとなる
12 *aはa[0]に等しい
13 #define p(g,x)**(v-2)^g+42||printf("%f\n",a[2]x*a);
14
15 グローバル変数で配列を宣言しているので要素数は1と解釈される
16 アクセス範囲が狭ければメモリ破壊を犯しても動く
17 float a[];
18
19 関数の型は省略するとintとして解釈される 引数の型も省略すると
  intとして扱われる
20 main(i,v)
21
22 ANSI 標準化以前の関数宣言
23 char**v;
24 {
25
26 条件式i--はiを評価した後に減算するのでi+1回の繰り返しとなる
27 for(;i--;)
28
29 sscanfを用いて文字配列を浮動小数点数へ変換
30 i=2のときコマンドの第1引数v+1がa+2へ,
31 i=0のとき第3引数v+3がaへ
32 sscanf(*v++,"%f",a+i);
33
34 定義してあるpを用いて各演算子であるかチェックし出力
35 p(1,+)p(3,-)p(,*)p(5,/);
36 }
37 (136B)
38 */
```

続いて, cc を-o で出力先を指定して実行した.

コード 2: コンパイル

```
1 $ cc -o calc calc.c
```

最後に実行し, 結果を得た.

コード 3: 実行結果

```
1 $ ./calccg 1.2 \* 0.5
2 0.600000
```