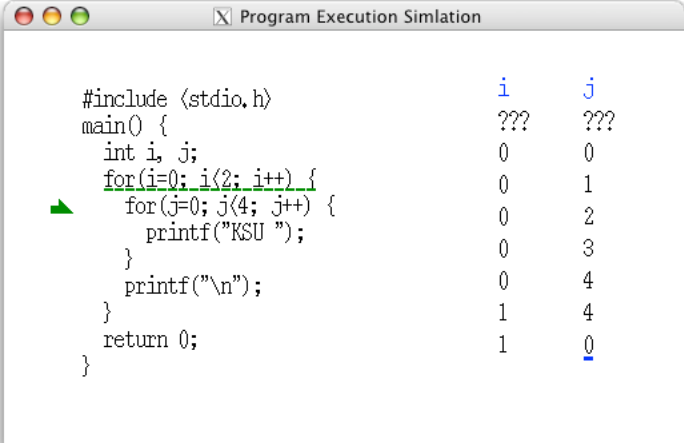


プログラミング演習 A 教材 (#6.5) ループ復習のためのシミュレータ

プログラムは指示に従って一行一行、順序よく実行されています。これを感じてもらうために、プログラム実行のシミュレータを作ってみました。実行するには、以下のようにしてください。

```
% cd ~yasuda/sim      ← カレントディレクトリをここに移動
% ./trace1            ← trace1 というプログラムを実行
```

起動すると右図のような画面が表示されます。操作方法は以下の通りです。



```
Program Execution Simulation

#include <stdio.h>
main() {
    int i, j;
    for(i=0; i<2; i++) {
        for(j=0; j<4; j++) {
            printf("KSU ");
        }
        printf("\n");
    }
    return 0;
}

i      j
???    ???
0      0
0      1
0      2
0      3
0      4
1      4
1      0
```

1. このウィンドウをクリックし、画面の一番上に表示させる。
2. Enter キーを押すごとに、一行ずつ実行される。このときプログラムのどの部分を実行し、変数の値がどのように変化するか図示される。
3. 一行の実行が終わると、次にどこを実行するかが右矢印で示され、そこで停止して次の Enter キーのタイプを待つ。

なお、プログラムが `printf()` を実行すると、`./trace1` を実行したターミナル (`Kterm`) の画面上に、“KSU” という文字が表示されます。この出力とシミュレータウィンドウを見比べながら、プログラムの動きを感じて下さい。

以下のようなプログラムが用意されています。

- `trace1` - `while()` を使った簡単なプログラム (1-3 までの数字を書く)
- `trace2` - `for()` を使った二重ループ (“KSU” を 2 行 4 列に並べる)
- `trace3` - `while()` を使った九九の表 (9x9 までやらず、2x3 に縮めています)
- `trace4` - `do while()` を使った九九の表 (9x9 までやらず、2x3 に縮めています)

こうしたシミュレーションはもちろん紙とペンでも実行できます。プログラムの動きが何かおかしい、納得できないというような状況になったら、ぜひ自分の手元でプログラムの動きと、重要な変数の値を追いかけながらシミュレーションを試みて下さい。

□ 課題 1.

以下のように、上下左右が逆転した九九の表 (左上が 81 で右下が 1 となっている) を出力するプログラムを `for`, `while`, `do while` のいずれかを使った二重ループで作って下さい。

```
81 72 63 54 45 36 27 18 9
72 64 56 48 40 32 24 16 8
63 56 49 42 35 28 21 14 7
54 48 42 36 30 24 18 12 6
45 40 35 30 25 20 15 10 5
36 32 28 24 20 16 12 8 4
27 24 21 18 15 12 9 6 3
18 16 14 12 10 8 6 4 2
9 8 7 6 5 4 3 2 1
```