

学籍番号: _____

氏名: _____

練習 1 次のプログラム「ハノイの塔 (再帰的)」の第 18 行～第 20 行の空欄を埋め、プログラムを完成しなさい。裏面の**コンパイルと実行例**も参考にすること。

● ハノイの塔 (再帰的)

hanoi.c

```

1: #include <stdio.h>
2:
3: void hanoi(int n, char x, char y, char z);
4:
5: int main(void)
6: {
7:     int n = 5;
8:
9:     /* printf("円盤の枚数?: "); */
10:    /* scanf("%d", &n); */
11:    hanoi(n, 'A', 'B', 'C');
12:
13:    return 0;
14: }
15:
16: void hanoi(int n, char x, char y, char z)
17: {
18:     if (      ) hanoi(      ,      ,      );
19:     printf("円盤 %d を %c から %c に移す\n",      ,      ,      );
20:     if (      ) hanoi(      ,      ,      );
21: }
```

練習 2 上記のプログラム「ハノイの塔 (再帰的)」(ファイル名「hanoi.c」)を参考に、出力が裏面の**コンパイルと実行例 1**と一致するようにプログラム(ファイル名「hanoi1.c」)を改良しなさい。ヒント: ステップ数をカウントする変数の関数間の受け渡しにはポインタを利用するとよい(グローバル変数を利用するのが最も簡単である)。

コンパイルと実行例：

```
$ gcc -o hanoi.exe hanoi.c 
```

← コンパイル

```
$ ./hanoi.exe 
```

```
円盤 1 を A から C に移す  
円盤 2 を A から B に移す  
円盤 1 を C から B に移す  
円盤 3 を A から C に移す  
円盤 1 を B から A に移す  
... 省略 ...  
円盤 1 を C から B に移す  
円盤 3 を A から C に移す  
円盤 1 を B から A に移す  
円盤 2 を B から C に移す  
円盤 1 を A から C に移す
```

```
$
```

コンパイルと実行例 1：

```
$ gcc -o hanoi1.exe hanoi1.c 
```

← コンパイル

```
$ ./hanoi1.exe 
```

```
Step 1: 円盤 1 を A から C に移す  
Step 2: 円盤 2 を A から B に移す  
Step 3: 円盤 1 を C から B に移す  
Step 4: 円盤 3 を A から C に移す  
Step 5: 円盤 1 を B から A に移す  
... 省略 ...  
Step 27: 円盤 1 を C から B に移す  
Step 28: 円盤 3 を A から C に移す  
Step 29: 円盤 1 を B から A に移す  
Step 30: 円盤 2 を B から C に移す  
Step 31: 円盤 1 を A から C に移す
```

```
$
```