

ソフトウェア 1 lecture6 課題 3

340481H 電子情報工学科内定 中里徳彦

2014/11/13

1 問題

```
#include <stdio.h>

int main()
{
    int **p;
    int *s;
    int a[3][2];
    int i, j;

    p = &s;
    s = &a[0][0];

    for (i = 0; i < 3; i++) {
        for (j = 0; j < 2; j++) {
            a[i][j] = 2*i + j;
        }
    }

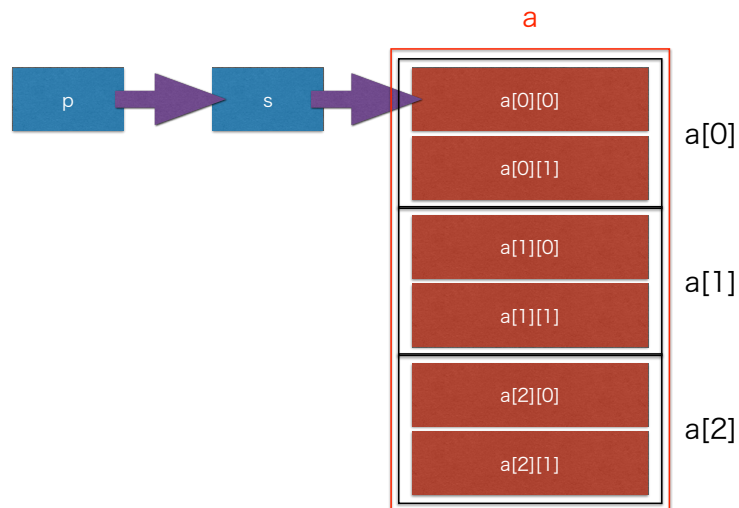
    printf("%d %d %d %d\n", **p, *(p+1), *(s+1), (*(a+2)+1));
    return 0;
}
```

上のプログラムをみて次の問題に答えよ。

1. 5行目の `int **p;` と 19行目の `**p` の意味の違いについて述べよ。
2. `p,s,a` の関係を図で表せ。
3. 19行目の出力結果を書け。
4. `a[3][2]` がどのような構造になっているか、プログラム終了時にどのような数が入っているかを図で書け。
5. 19行目を `a` を用いてポインタを使わずわかりやすく書き直せ。ただし出力結果が不定となる場合はのぞいてよい。

2 回答

1. `int **p;` は `int` 型変数のアドレスを持つポインタのアドレスを持つポインタをメモリに確保することを表している。`**p` はアドレス `p` にあるポインタ `s` が持っているアドレスにある `int` 型変数の値を返すという意味である。
2. 下にある図の通り
3. 0 不定 1 5
4. `[[0][1]] [[2][3]] [[4][5]]`
5. `printf("%d %d %d\n", a[0][0], a[0][1], a[2][1]);`



3 効果的と考える理由

1. C 言語のポインタがわかりづらい理由の一つは、同じ表現であっても式と宣言で全く違う意味になることがあるからであり、それを問うことでポインタをどの程度理解しているかを把握できると考えた。
2. どのポインタがどの変数を指すのかがわかるのは、ポインタで最も基本的な部分でありそれを問うことは重要だと思うから。
3. ポインタ演算がどのような結果を示すかを理解していることは、配列とポインタの関係を理解するうえで重要だから。
4. `a[3][2]` が要素数 2 の `int` 型配列が 3 つ並んだ配列を意味するのか、それとも、要素数 3 の `int` 型配列が 2 つ並んだ配列を意味するのかがわかりにくいから。
5. ポインタ演算の直感的なわかりにくさを配列を用いて表現した場合と比較することで理解する。