

# 問題の解答

## ◆ 練習問題

1. 大きさ 10 の配列を宣言し、そのアドレスを一行ずつ表示しなさい。

```
#include<stdio.h>
int main(void) {
    int a[10];
    int i;
    for (i = 0; i < 10; i++)printf("&a[%d] = %p\n", i, &a[i]);
}
```

2. 2 つの変数のアドレスを受け取ってその中身の値を比較し、小さいほうの値を大きいほうの値に統一する関数。

```
void compare(int *p, int *q)
```

を作成しなさい。

```
#include<stdio.h>
void compare(int *p, int *q) {
    if (*p > *q)*q = *p;
    else *p = *q;
}
int main(void) {
    int a,b;
    printf("a : "); scanf("%d", &a);
    printf("b : "); scanf("%d", &b);
    printf("a = %d, b = %d\n", a, b);

    compare(&a, &b);
    printf("a = %d, b = %d\n", a, b);
}
```

## ◆ ∇問題

1. `array_twice` 関数について、元の配列 `a` は書き換えずかつ、別に用意した結果出力用の配列に結果を出力するようにしなさい。ただし、結果出力用の配列がそれに十分な要素数を持つのは呼び出し側の責任としてよい。

```
#include<stdio.h>
#define N 5
void array_twice(const int a[], int b[], int n) {
    int i;
    for (i = 0; i < n; i++)b[i] = a[i] * 2;
}
int main(void) {
    int a[N], b[10];
    int i;
    for (i = 0; i < N; i++) {
        a[i] = i;
        printf("a[%d] = %d\n", i, a[i]);
    }
    array_twice(a, b, N);
    printf("\n");
    for (i = 0; i < N; i++)printf("a[%d] = %d\n", i, a[i]);
    printf("\n");
    for (i = 0; i < N; i++)printf("b[%d] = %d\n", i, b[i]);
}
```

2. リストの例について、**struct** 型の構造体 **d** を新しく宣言、**d.value** を **5** として **b** と **c** の間に挿入しなさい (**a**、**b**、**d**、**c** の順で走査されるようにする)。ただし、挿入の処理は `/* √`  
2. `*/` でコメントアウトしている部分を書き換えること。`b.value = 3;`の次の行を `b.next = &d;`とするのは認めない。

```
#include<stdio.h>
struct cell {
    int value;
    struct cell *next;
};
int main(void) {
    struct cell a, b, c, d, *p;
    a.value = 1;
    a.next = &b;

    b.value = 3;
    b.next = &c;

    c.value = 7;
    c.next = NULL;

    b.next = &d;
    d.value = 5;
    d.next = &c;

    p = &a;
    while (p != NULL) {
        printf("%d¥n", p->value);
        p = p->next;
    }
}
```

`b.next` を `&d` にして `b` の次は `d`、`d.next` を `&c` にすることで `d` の次は `c` とすることで、リストへの要素の挿入を実現しています。