

1. 整数関連

1.1 値の交換

変数 A と B の値を交換するには、次のように作業変数を介する方法が一般的です。

```
void swap(int* A, int* B) {
    int t; t = *A; *A = *B; *B = t;
}
```

どうしても作業変数を使いたくないときは、排他的論理和を使います。すなわち、

$$A \wedge (A \wedge B) = B, \quad (A \wedge B) \wedge B = A$$

となることを利用します。以下が排他的論理和を使った値の交換です。

```
void swap(int* A, int* B) {
    *B = *A ^ *B; *A = *A ^ *B; *B = *A ^ *B;
}
```

以上の値の交換を使って、配列要素の順序を逆転する処理ができます。この例は、上記の呼出し側の例にもなっています。

```
void reverseArray(int N, int A[]) {
    for(int i=0; i < 1 / 2; i++) swap(&A[i], &A[N-i-1]);
}
```

1.2 2 値の最大・最小

2 値の最大・最小を求めるには、単純に if 文を使います。

```
int max2(int A, int B) {
    if(A>=B) return A; else return B;
}
int min2(int A, int B) {
    if(A<=B) return A; else return B;
}
```

古い BASIC のように関数中で条件判断ができない場合、次のように処理する方法もありますが、現在では、分かりづらいとして敬遠されるかもしれません。

```
int max2(int A, int B) {
    return (A + B + abs(A - B)) / 2;
}
int min2(int A, int B) {
    return (A + B - abs(A - B)) / 2;
}
```