

완전 탐색 알고리즘

정의

- 가능한 경우를 모두 탐색하는 알고리즘
- 정확도가 높음
- 시간복잡도 계수가 높음

예시

```
#include <cstdio>
#include <algorithm>
using namespace std;

int main(){
    int N, arr[1000];
    scanf("%d", &N);
    for(int i=0; i<N; i++)
        scanf("%d", arr+i);

    int result = 0;
    for(int i=0; i<N; i++)
        for(int j=0; j<N; j++)
            if(i!=j) result = max(result, arr[i]+arr[j]);
    printf("%d\n", result);
}
//시간 복잡도 O(N^2)
```

```
#include <cstdio>
#include <algorithm>
using namespace std;

int main(){
    int N, arr[1000];
    scanf("%d", &N);
    for(int i=0; i<N; i++)
        scanf("%d", arr+i);
    sort(arr, arr+N);
    printf("%d\n", arr[N-1]+arr[N-2]);
}
//시간복잡도 O(NlogN)
```

문제풀이

[acmicpc 2309] [acmicpc 2309](<https://www.acmicpc.net/problem/2309>)

문제 종류

완전탐색 - 1476, 1107, 1451, 9095, 10819, 10971, 1697, 1963, 9019, 1525, 2251, 2186, 3108, 5014, 1759, 2580, 1987, 6603, 1182, 2003, 1806(풀기), 1644, 1261, 1208, 7453, 2632, 2143