

## Week 9

- 주의 사항: 부정행위 금지, STL 사용 금지(string, vector 사용가능), 인터넷 금지, STL 벡터를 이용하여 구현할 것.
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

### 문제 2

비정렬 벡터로 구현된 우선순위 큐  $P$ 를 이용하여, 자연수  $n$  ( $n$ 은  $1 \leq n \leq 10,000$ )으로 구성된 수열  $S$ 를 입력 받으면, 특수한 조건에 맞추어 우선순위를 결정하고 우선순위가 높은 순으로 수열을 출력하는 프로그램을 짜보자. 우선순위 큐  $P$ 의 우선순위를 결정하는 조건은 아래와 같다. (단, 수열에 중복되는 수는 없다고 가정한다.)

1. 짝수는 홀수보다 항상 우선순위가 높다
2. 같은 조건에 있는 자연수 중, 작은 수일수록 우선순위가 높다

|            |   |   |   |   |   |   |   |
|------------|---|---|---|---|---|---|---|
| 입력된 수열 $S$ | 7 | 4 | 8 | 2 | 5 | 3 | 9 |
| 결정된 우선순위   | 6 | 2 | 3 | 1 | 5 | 4 | 7 |

우선순위 순 출력 결과 : {2,4,8,3,5,7,9}

그림 1. 수열  $S$ 을 조건에 맞추어 우선순위를 할당한 결과

예시로 수열  $S=(7, 4, 8, 2, 5, 3, 9)$ 을 입력 받았다고 가정해보자. 위 조건에 맞추어 우선순위를 설정할 경우 그림 (1)과 같다. 이에, 우선순위가 높은 순으로 출력할 경우 경우 {2, 4, 8, 3, 5, 7, 9}와 같이 출력된다.

### 입력

1. 첫 번째 줄에 테스트 케이스의 개수  $T$  ( $1 \leq T \leq 1,000$ )을 입력한다.
  - 1) 각 케이스마다 다음이 반복되어 나타난다.
  - 2) 각 케이스의 첫 번째 줄에 입력 받을 자연수의 개수  $N$  ( $5 \leq N \leq 20$ )을 입력한다.
  - 3) 각 케이스의 두 번째 줄에  $N$ 개의 자연수  $n$  ( $1 \leq n \leq 10,000$ )를 띄어쓰기로 구분하여 입력한다.  
(단, 각 케이스마다 입력되는 자연수  $n$ 은 중복되지 않는다.)

### 출력

각 케이스마다 입력 받은 수열을 문제의 우선순위 조건에 맞추어 높은 우선순위 순으로 출력한다.

예제 입출력

| 예제 입력                                | 예제 출력                                |
|--------------------------------------|--------------------------------------|
| 8                                    | 12 22 32 60 62 84 88 9 11 27 51 67   |
| 12                                   | 44 46 52 35 61                       |
| 12 88 84 9 60 62 32 51 27 11 22 67   | 8 52 56 29 35 41 53                  |
| 5                                    | 10 22 26 82 90 98 41 73 75 79        |
| 35 44 46 61 52                       | 22 30 48 60 76 33 57 73              |
| 7                                    | 2 84 43 49 57 87 89 99               |
| 35 53 56 8 29 52 41                  | 26 21 37 43 49                       |
| 10                                   | 24 26 40 64 70 76 5 7 19 39 45 77 89 |
| 41 79 82 98 26 10 73 90 75 22        |                                      |
| 8                                    |                                      |
| 33 76 22 57 60 73 30 48              |                                      |
| 8                                    |                                      |
| 2 99 87 49 84 57 43 89               |                                      |
| 5                                    |                                      |
| 37 26 49 43 21                       |                                      |
| 13                                   |                                      |
| 7 76 5 24 70 39 64 26 89 77 45 19 40 |                                      |