주차장

[문제] 그림-1과 같이 한 대씩 들어갈 수 있는 slot이 선형 주차장을 가진 박람회(exhibition)가 있다. 이 주 차장을 정수로 표시된 차들이 사용한다. 들어오는 차들은 비어있는 slot 중에 가장 빠른 번호에 주차 해야 한다. 만일 주차 공간이 꽉 차면 박람회 주최 측에서는 주차 공간을 2배로 늘인 다음 주차시킨 다. 만일 주차된 차가 전체 주차공간의 1/3 이하가 되면 주차공간을 반으로 줄이고 주차된 차들은 왼 쪽으로 모두 모이도록 새로 주차시킨다. 5대 공간 주차장의 예를 보자.

1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
A	В	C	D		A		С	D		A	Т	С	D		E	Т	С	D	В

그림-1 : 자동차 주차 순서가 [A, B, C, D, -B, T, -A, E, B] 인 경우.

만일 이 상황에서 새로운 차 W가 들어오면 주차공간은 다음과 같이 2배로 확장된다. 이후 Q, Z까지 들어온 상황이다. 여기서 -E, -C, -W, -D, -T가 되면 남은 차는 3대, 즉 10/3=3대 이하가 되므로 주차장은 반으로 축소되고 주차된 차들은 모두 왼쪽으로 옮겨져 compaction 된다.

1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
E	Т	С	D	В	w	Q	Z							В		Q	Z		
1	2	3	4	5	6	7	8	9	10	1	2	3	4	5					
В	Q	Z								В	Q	Z							

[입출력] parking.inp의 첫 줄에는 초기 주차장의 면(slot)수 K가 주어지고 오간 차들의 총 수 N이 주어진다. 단 $2 \leq K \leq 20$, $10 \leq N \leq 2^{16}$, $1 \leq W_i \leq 2^{16}$ 이다. 이어지는 N개의 각 줄에는 자동차 번호 W_i 가 주어진 다. 양수는 입차, 음수 '- W'는 출차를 의미한다. 출력은 최종 상황에서 주차된 slot의 번호와 자동차 번호, 이 2개의 정수를 주차 slot 번호의 오름차순으로 출력해야 한다. 한계 대수는 버림 정수로 계산된다. 즉 100대 공간이면 33대부터, 50대 공간이면 16대부터 주차장 축소 compaction을 한다. 단, 주차 공간 크기는 항상 K보다 크거나 같은 상태를 유지한다.

[예제]

parking.inp	parking.out
5 10 //K=5, N=10 10000 20345 30111 45010 -20345 //출자	1 30111 // s1에 자동차 30111 주차 2 637282 8 32302 10 45010 // 4대만 남은 상황
435168	

[조건] 프로그램의 이름은 parking.{c, cpp, java, py}, 제출횟수는 15회, 제한시간은 1초이다. 마감 시간은 9월 22일(토) 저녁 10시, 제출은 9월 19일(수)부터 가능하다. 연습용 데이터를 이용해서 올바 른 답이 나오는지 확인해 볼 수 있다.