

Asteroid Collision

Problem

Submissions

Leaderboard

Discussions

$$|-5| = \text{magnitude.}$$

$$= 5$$

We are given an array asteroids of integers representing asteroids in a row.

For each asteroid, the absolute value represents its size, and the sign represents its direction (positive meaning right, negative meaning left). Each asteroid moves at the same speed.

Find out the state of the asteroids after all collisions. If two asteroids meet, the smaller one will explode. If both are the same size, both will explode. Two asteroids moving in the same direction will never meet.

+ve →

-ve ←

Sample Input 0

3
5
10
-5

5

10

-5

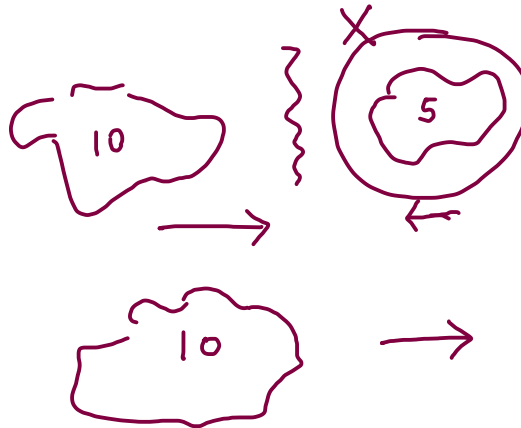
Sample Output 0

5 10

eg.

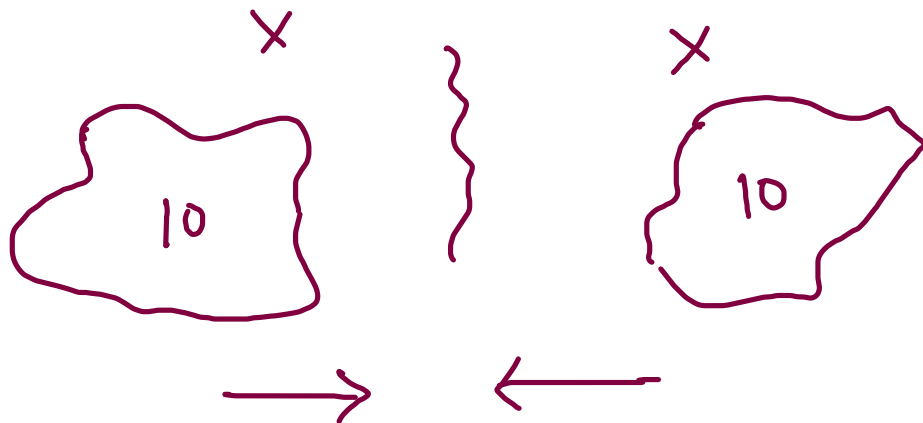
10

-5



eg 2.

10 - 10



Nothing.

5

10

- 5

5

10

3

x

5



5

10



eg 4.

2 1

2^x

(2)

→

(1)

→

eg 5.

-2 -1

(-2)

←

(-1)

←

eg. 6.

-2 { 3

2
←

3
→

only 1 case.

for collision

+ -
→ ←

10 2^x } - 5

→ → ←

10 } -5^x

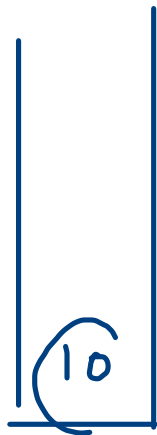
→ ←

10
→

$n=5$.

$$\begin{array}{cccc} 10 \left. \vphantom{\begin{matrix} x \\ 4 \end{matrix}} \right\} c_3 & \begin{matrix} x \\ 4 \end{matrix} \left. \vphantom{\begin{matrix} x \\ 2 \end{matrix}} \right\} c_2 & \begin{matrix} x \\ 2 \end{matrix} \left. \vphantom{\begin{matrix} x \\ -5 \end{matrix}} \right\} c_1 & \begin{matrix} x \\ -5 \end{matrix} \left. \vphantom{\begin{matrix} x \\ -6 \end{matrix}} \right\} c_4 \\ & \downarrow & & \end{array}$$

(10)



$$2 - 5 = \underline{-3} \quad \dots \text{ -ve}$$

$$4 - 5 = \text{ -ve}$$

$$10 - 5 = \text{ +ve}$$

$$10 - 6 = \text{ +ve}$$

n=8.

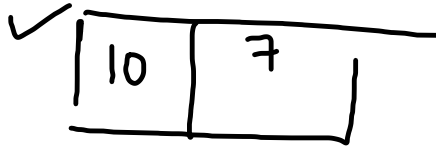
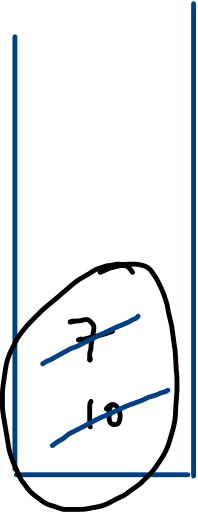
✓

10 7 4 2 1 -2 -5 -6

peek +ve
curr -ve.

↓

7 - 6 = +ve



↓ ↓
[10, 7]


```

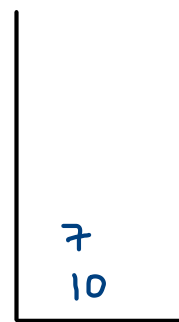
4 public class Solution {
5     public static void main(String[] args) {
6         Scanner scn = new Scanner(System.in);
7         int n = scn.nextInt();
8         int [] A = new int[n];
9         for(int i = 0; i < n; i++){
10             A[i] = scn.nextInt();
11         }
12         Stack<Integer> st = new Stack<>();
13         for(int i = 0; i < n; i++){
14             //peek + && curr -ve : collision
15             int curr = A[i];
16             while(st.size() != 0 && curr < 0 && st.peek() > 0){
17                 int s = curr + st.peek();
18                 if(s > 0){ //ignore curr and move on
19                     curr = 0;
20                 }else if(s < 0){
21                     st.pop();
22                 }else{ //s == 0
23                     st.pop();
24                     curr = 0;
25                 }
26             }
27             if(curr != 0){
28                 st.push(curr);
29             }
30         }
31         for(int e : st){
32             System.out.print(e + " ");
33         }

```

10 7 4 2 1 -2 -5 -6

$$s = -6 + 7 = +$$

s =



$$curr = \cancel{-6} 0$$

10 7

```

4 public class Solution {
5     public static void main(String[] args) {
6         Scanner scn = new Scanner(System.in);
7         int n = scn.nextInt();
8         int [] A = new int[n];
9         for(int i = 0; i < n; i++){
10             A[i] = scn.nextInt();
11         }
12         Stack<Integer> st = new Stack<>();
13         for(int i = 0; i < n; i++){
14             //peek + && curr -ve : collision
15             int curr = A[i];
16             while(st.size() != 0 && curr < 0 && st.peek() > 0){
17                 int s = curr + st.peek();
18                 if(s > 0){ //ignore curr and move on
19                     curr = 0;
20                 }else if(s < 0){
21                     st.pop();
22                 }else{ //s == 0
23                     st.pop();
24                     curr = 0;
25                 }
26             }
27             if(curr != 0){
28                 st.push(curr);
29             }
30         }
31         for(int e : st){
32             System.out.print(e + " ");
33         }

```

-10 7 4 2 1 ~~-2~~ ~~-3~~ ~~-6~~
0 0 0

7 - 6 = +

7
-10

-10 7

Word Meaning

Problem

Submissions

Leaderboard

Discussions

You are required to create a **dictionary** consisting of word and its meaning.

Take an integer **N** as input and **Continue** the process until **Case 4** is not achieved.

- If **N==1**, take **word** and **meaning** as input from user and **add** it to the dictionary.
- If **N==2**, take a **word** as input from the user and **print its meaning**. If the word is not found print -1.
- If **N==3**, take a **word** as input from the user and delete it from the dictionary.
- If **N==4**, **Close** the dictionary. **Exit** the program.

K , V
w , M

1 → w/M → i/p put
2 → w → i/p → get) (-1)
3 → w → i/p → remove

Sample Input 1

```
1
Geekster
Coding
1
Geek
Coder
2
Geek
3
Geek
2
Geekster
2
Geek
4
```

Sample Output 1

```
Coder
Coding
-1
```

true

i=2

while (i < 10)

{

i++

}

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         HashMap<String , String> hm = new HashMap<>();
9         while(true){
10             int N = scn.nextInt();
11             if(N == 1){
12                 String word = scn.next();
13                 String meaning = scn.next();
14                 hm.put(word, meaning);
15             }else if(N == 2){
16                 String word = scn.next();
17                 System.out.println(hm.getOrDefault(word, "-1"));
18             }else if(N == 3){
19                 String word = scn.next();
20                 hm.remove(word);
21             }else if(N == 4){
22                 break;
23             }
24         }
25     }
26 }
```

HashMap

k, v

put

get

size

remove

containsKey

HashSet

k

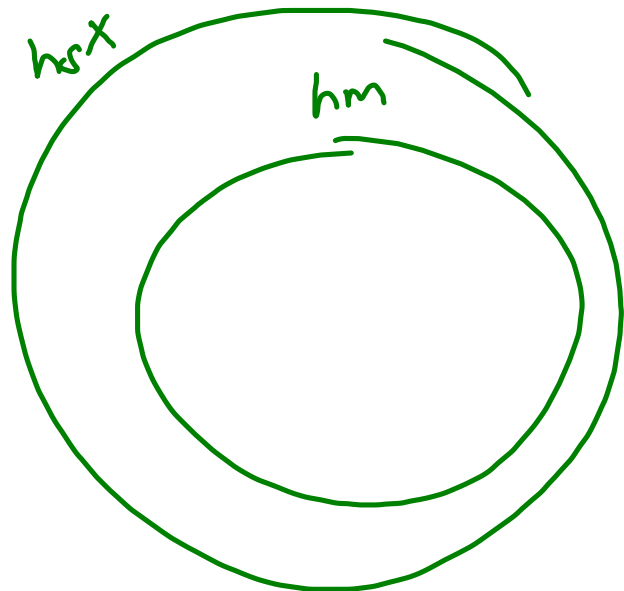
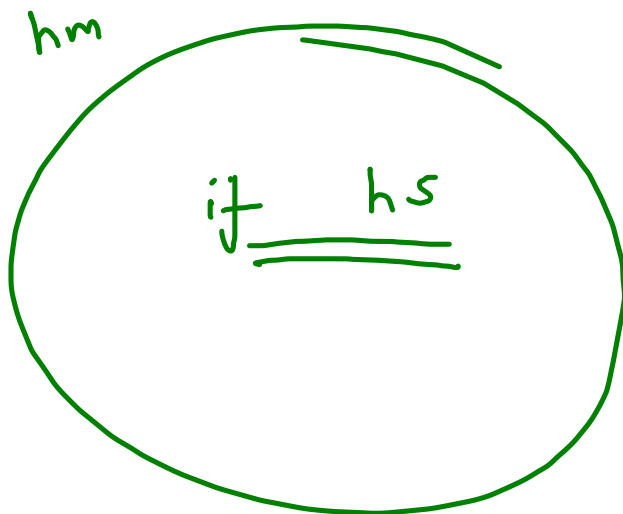
add

size

remove

contains

```
1 import java.util.*;
2 public class Main
3 {
4     public static void main(String[] args) {
5         HashSet<String> hs = new HashSet<>();
6         hs.add("Aman");
7         hs.add("Aman");
8         hs.add("Jatin");
9         hs.add("Abhinav");
10        hs.add("Lalit");
11
12        System.out.println(hs.contains("Lalit"));
13        hs.remove("Lalit");
14
15        // System.out.println(hs);
16        System.out.println(hs.contains("Lalit"));|
17
18
19
20     }
21 }
```



a b c b d c c a

✓

2.

unique.

4

hs

HM



x

a - 2

b - 2

c - 3

d - 1

hm.size()

1. freq.

a - 2

b - 2

c - 3

d - 1