Practice Midterm

Thursday, March 12, 12:30 - 1:45pm

This exam is:

- open notes/books/any printed resources
- open laptop (anything that you have on your own computer, not online storage)

Online resources that you are allowed to access:

- Gradescope
- course website
- language documentation:
  - Java: https://docs.oracle.com/javase/10/docs/api/
  - C++: https://www.cplusplus.com/reference/ and/or https://en.cppreference.com/w/

Instructions:

Solve three out of the four problems given on the next pages. You will **not** get extra credit for solving all four problems.

Grading:

Every exam problem is graded out of 10 points. The total exam grade is the weighted sum computed as follows (assume *scoreN* is a score for a particular problem with *score1* >= *score2* >= *score3*):

*exam = 5 * score1 + 3 * score2 + 2 * score3*

The total score for a problem is determined by the maximum between zero and the sum of scores for individual tests based on their results. The maximum score for each test is determined by *max_score = 10/number_of_tests*.

test outcome	test score
passed test	*max_score*
wrong answer	*- 0.5 max_score*
runtime error	*- 0.5 max_score*
timeout error	*- 0.5 max_score*
presentation error	*0.75 max_score*

Giant Squad

Conflict Empire is fighting a war against Light Kingdom. The commander of Conflict Army is building a squad of giants. Giants are huge creatures, typically a dozen feet tall, inhabiting the territory of Conflict Empire. Giants are so tall that their height differences are also very large, preventing orders from being effectively transmitted. To address this issue, the commander decides to form squads with only odd number of giants with different heights and select the captain in such a way so that the number of giants in the squad who are shorter than that captain is equal to the number of giants who are taller than that captain. You are given heights of all giants in that squad and you are asked to determine the height of the captain.

Input

The input consists of a single line, starting with an integer N ($1 < N < 11$, N is odd), the number of giants in that squad, followed by N integers representing heights of the giants of the squad. Those N integers will be between 11 and 20 (inclusive). Additionally, heights will be given in strictly increasing or decreasing order.

Output

You should print a single integer x, where x is the height of the captain.

Example 1

```
Input:
5 19 17 16 14 12

Output:
16
```

Example 2

```
Input:
5 12 14 16 17 18

Output:
16
```

Prime Pair

A prime pair is a pair of the form (p, p + 2) where p and (p + 2) are both prime. The first few prime pairs are (3, 5), (5, 7), (11, 13). In this problem, you are given an integer N and you should find the N'th prime pair.

Input
The input will contain a single integer N (1 <= N <= 100,000).

Output
The output should be a single **line**, the N'th prime pair formatted as `(p1, p2)` (there is a space after the comma). It is guaranteed that primes in the 100,000th prime pair are less than 20,000,000.

Example 1

```
Input:
1

Output:
(3, 5)
```

Example 2

```
Input:
2

Output:
(5, 7)
```

Example 3

```
Input:
3

Output:
(11, 13)
```

Example 4

```
Input:
4

Output:
(17, 19)
```

Transitive Relation

In mathematics, a relation R over a set X is *transitive* if for all elements a, b, c in X, whenever R relates a to b and b to c, then R also relates a to c. Transitivity is a key property of both partial orders and equivalence relations.

For example, whenever x = y and y = z, then also x = z, thus "is equal to"(equality) is a transitive relation over real numbers.

One may naturally split the set S into equivalence classes with transitive relations.

Formally, given a set S and an transitive relation ~ over S, the equivalence class of an element a in S is the set `EC(a) = { x ∈ S | x ~ a }`

Given a finite set S and a transitive relation R over S. Find the largest equivalent class, i.e., `find max{ |EC(a)| | a ∈ S }`.

Input

The first line contains tho numbers N and M. N is the number of elements in S and M is the number of pairs of elements, which are known to be in the relation R. You can assume that 1 ≤ N ≤ 30,000 and 0 ≤ M ≤ 500,000.

Elements in S are indexed with integer from 1 to N. Each of the following M lines contain a pair of integers A and B (1 ≤ A,B ≤ N, A ≠ B) which states that (A,B) belongs to the transitive relation R. There could be repetitions among the given pairs.

Output

Output 1 number followed by a newline. The size of the largest equivalent class.

Example 1

```
Input:
3 2
1 3
2 3

Output:
3
```

Example 2

```
Input:
4 3
1 2
2 3
1 3

Output:
3
```

Example 3

```
Input:
4 0

Output:
1
```

Find Sums

You are given a multiset of N integers (multiset means that the repeated values are allowed) and a target value S. Your task is to find all distinct subsets of the given multiset for which the sum of all the elements in the subset is equal to S.

Input

The first line of the input contains two integers S (1 <= S <= 1,000) and N (1 <= N <= 12), indicating the target sum and the number of values in the multiset, respectively.

The second line contains N integers, all of which are between 1 and 100 - these are the elements of the multiset.

Output

First, print a line `Sums of S:` where S is the value given in the input. Then print one line for every subset satisfying the condition or a line containing `NONE` if there is no such subset.

For every subset, numbers are printed in decreasing order and separated by a plus sign (`+`). The subsets themselves are sorted lexicographically in decreasing order, i.e. they are sorted by their first integer, then the second integer in case of tie, and so on. Additionally, the subsets you print should not contain repetitions (i.e., you should never print two lines that are identical).

Example 1

```
Input:
4 6
4 3 2 2 1 1

Output:
Sums of 4:
4
3+1
2+2
2+1+1
```

Example 2

```
Input:
5 3
2 1 1

Output:
Sums of 5:
NONE
```

Example 3

```
Input:
400 12
50 50 50 50 50 50 25 25 25 25 25 25

Output:
Sums of 400:
50+50+50+50+50+50+25+25+25+25
50+50+50+50+50+25+25+25+25+25+25
```