1. How Will You Compare?

Write a *Comparator* class with the following *3* overloaded *compare* methods:

1. *boolean compare(int a, int b)*: Return *true* if *int a = int b*, otherwise return *false*.
2. *boolean compare(string a, string b)*: Return *true* if *string a = string b*, otherwise return *false*.
3. *boolean compare(int[] a, int[] b)*: Return *true* if both of the following conditions hold *true*:
   - Arrays *a* and *b* are of equal length.
   - For each index *i* (where $0 \le i < |a|$, $|b|$), *a[i] = b[i]*.

   Otherwise, return *false*.

Note: For *C++*, both parameters are of type *Vector<int>*.

Constraints

- For strings, $1 \le |a|, |b| \le 2000$
- For integers, $0 \le a, b \le 10000000$
- For integer arrays, $0 \le |a|, |b| \le 10$

▼ Input Format for Custom Testing

Input from stdin will be processed as follows and passed to the function.

The first line contains an integer *T*, the number of test cases.
Each of the next *T* sets of lines is in one of the following formats:

- The first line contains the integer ~~*1*~~ *representing the comparison type (1, 2 or 3 for int, string or array comparison respectively).*
  The next two lines contain strings *a* and *b*.
- The first line contains the integer ~~*2*~~ *representing the overloaded function type.*
  The next two lines contain integers *a* and *b*.
- The first line contains the integer ~~*3*~~ *representing the overloaded function type.*
  The next three lines contain the following:
  1. Two space-separated integers ~~*n*~~ n and ~~*m*~~ *m*, the lengths of arrays *a* and *b*.
  2. A line of ~~*n*~~ n space-separated integers *a[i]*.
  3. A line of ~~*m*~~ *m* space-separated integers *b[i]*.

▼ Sample Case 0

Sample Input 0

```
STDIN          Function
-----          --------
3          →   T = 3 number of test cases.
1          →   Comparison type 1
hello world →  a = "hello world"
hello world →  b = "hello world"
2          →   Comparison type 2
3          →   a = 3
4          →   b = 4
3          →   Comparison type 3
3 3        →   a[] size n=3  b[] size m=3
1 2 3      →   a = [1, 2, 3]
1 2 3      →   b = [1, 2, 3]
```

Sample Output 0

```
Same
Different
Same
```

Explanation 0

There are *3* test cases:

Test Case	~~condition~~ comparison type	a	b	Output	Explanation
1	*1*	*"hello world"*	*"hello world"*	*"Same"*	Both strings are the same.
2	*2*	*3*	*4*	*"Different"*	The two integers are different ($3 \ne 4$).
3	*3*	*{1,2,3}*	*{1,2,3}*	*"Same"*	Both arrays have the same number of elements and each element *a[i] = b[i]*

▼ Sample Case 1

Sample Input 1

```
STDIN       Function
-----       --------
2           → T = 2 number of test cases.
3           → Comparison type 3
3 4         → a[] size=3 b[] size=4
1 2 3       → a = [1, 2, 3]
1 2 3 4     → b = [1, 2, 3, 4]
1           → Comparison type 1
HackerRank → a = "HackerRank"
HackerRank → b = "HackerRank"
```

Sample Output 1

```
Different
Different
```

Explanation 1

There are *2* test cases.

Test Case	comparison type	a	b	Output	Explanation
1	*3*	*{1, 2, 3}*	*{1, 2, 3, 4}*	"Different"	The arrays are different.
2	*1*	*HackerRank*	*hackerRank*	"*Different*"	The two strings are different.