

## A. Requirements

### Code (90%)

You can write your code in Java, Python, C, or C++. The *time limit* may vary among different languages, depending on the performance of the language. Your code must be a complete executable program instead of only a function. We guarantee test data strictly compliance with the requirements in the description, and you do not need to deal with cases where the input data is invalid.

#### Libraries in this assignment:

- For C/C++, you can only include standard library.
- For Java, you can only `import java.util.*`
- For Python, you can only import standard library. In other words, you cannot import libraries such as `numpy`.

We provide an example problem to illustrate the information above better.

### Report (10%)

You also need to write a report in `pdf` type to explain the following:

- What are the possible solutions for the problem?
- How do you solve this problem?
- Why is your solution better than others?

Please note that the **maximum** number of pages allowed for your report is **5 pages**.

Remember that the report is to illustrate your thinking process. Keep in mind that your report is supposed to show your ideas and thinking process. We expect clear and precise textual descriptions in your report, and we do not recommend that you over-format your report.

## B. Example Problem: A + B Problem

### Description

Given 2 integers A and B, compute and print  $A + B$

### Input

Two integers in one line: A, and B

### Output

One integer:  $A + B$

#### Sample Input 1

1 2
-----

#### Sample Output 1

3
---

### Problem Scale & Subtasks

For 100% of the test cases,  $0 \leq A, B \leq 10^6$

## Solutions

### Java

```
import java.util.*;

public class Example {
    public static void main(String[] args) {
        int a, b;
        Scanner scanner = new Scanner(System.in);
        a = scanner.nextInt();
        b = scanner.nextInt();
        scanner.close();
        System.out.println(a + b);
    }
}
```

### Python

```
AB = input().split()
A, B = int(AB[0]), int(AB[1])
print(A + B)
```

### C

```
#include <stdio.h>

int main(int argc, char *argv[])
{
    int A, B;
    scanf("%d%d", &A, &B);
    printf("%d\n", A + B);
    return 0;
}
```

### C++

```
#include <iostream>

int main(int argc, char *argv[])
{
    int A, B;
    std::cin >> A >> B;
    std::cout << A + B << std::endl;
    return 0;
}
```

## C. Submission

After finishing this assignment, you are required to submit your code to the Online Judge System (OJ), and upload your .zip package of your code files and report to BlackBoard.

### C.1 Online Judge

Once you have completed one problem, you can submit your code on the page on the Online Judge platform ([oj.cuhk.edu.cn](http://oj.cuhk.edu.cn), campus only) to gain marks for the code part. You can submit your solution of one problem for **no more than 80 times**.

After you have submitted your program, OJ will test your program on all test cases and give you a grade. The grade of your latest submission will be regarded as the final grade of the corresponding problem. Each problem is tested on multiple test cases of different difficulty. You will get a part of the score even if your algorithm is not the best.

**Note:** The program running time may vary on different machines. Please refer to the result of the online judge system. OJ will show the time and memory limits for different languages on the corresponding problem page.

If you have other questions about the online judge system, please refer to [OJ wiki](#) (campus network only). If this cannot help you, feel free to contact us.

## C.2 BlackBoard

You are required to upload your **source codes and report** to the BlackBoard platform. You need to name your files according to the following rules and compress them into `A1_<Student ID>.zip` :

```
A1_<Student ID>.zip
|-- A1_P1_<Student ID>.java/py/c/cpp
|-- A1_P2_<Student ID>.java/py/c/cpp
|-- A1_Report_<Student ID>.pdf
```

For Java users, **you don't need to consider the consistency of class name and file name.**

For example, suppose your ID is 123456789, and your problem 1 is written in Python, problem 2 is written in Java then the following contents should be included in your submitted `A1_123456789.zip`:

```
A1_123456789.zip
|-- A1_P1_123456789.py
|-- A1_P2_123456789.java
|-- A1_Report_123456789.pdf
```

## C.3 Late Submissions

Submissions after Oct 6 2023 23:59:00(UTC+8) would be considered as LATE.

The LATE submission page will open after deadline on OJ.

Submission time =  $\max\{\text{latest submission time for every problem, BlackBoard submission time}\}$

There will be penalties for late submission:

- 0–24 hours after deadline: final score = your score  $\times$  0.8
- 24–72 hours after deadline: final score = your score  $\times$  0.5
- 72+ hours after deadline: final score = your score  $\times$  0

## FAQs

**Q:** I cannot access to Online Judge.

**A:** First, please ensure that you are using the campus network. If you are not on campus, please use the university VPN. Second, please delete cookies and refresh browser or use other browser. If you still cannot access to Online Judge, try to visit it via the IP address [10.26.200.13](#).

**Q:** My program passes samples on my computer, but not get AC on OJ.

**A:** Refer to [OJ Wiki Q&A](#)

## Authors

If you have questions for the problems below, please contact:

- Problem 1. Xuanhao Pan: [xuanhaopan@link.cuhk.edu.cn](mailto:xuanhaopan@link.cuhk.edu.cn)
- Problem 2. Tianci Hou: [tiancihou@link.cuhk.edu.cn](mailto:tiancihou@link.cuhk.edu.cn)

# CSC3100 Data Structures Fall 2023

## Programming Assignment 1

Due: Oct 6 2023 23:59:00

Assignment Link: <http://oj.cuhk.edu.cn/contest/csc310023falla1>

Access Code: 9v7Dxqet

## 1 Queue Disorder (40% of this assignment)

### 1.1 Description

In a queue, people are supposed to stand in ascending order of their heights. However, due to some confusion, the order of the queue gets disrupted. Your task is to determine the number of disorder pairs in the queue.

A disorder pair is defined as a pair of people  $(p_i, p_j)$  such that  $i < j$  and  $p_i$  is taller than  $p_j$  in the queue, which means their order is reversed.

For example, consider a queue with 5 people: [180, 160, 175, 170, 170]. In this case, there are 6 disorder pairs: (180, 160), (180, 175), (180, 170), (180, 170), (175, 170) and (175, 170). Please note that (170, 170) is not considered as a disorder pair.

Write a program to calculate the number of disorder pairs in a given queue.

### 1.2 Input

The first line of input contains an integer  $N$  ( $1 \leq N \leq 10^6$ ), representing the number of people in the queue.

The second line contains  $N$  space-separated integers  $p_1, p_2, \dots, p_N$  ( $1 \leq p_i \leq 10^9$ ), representing the heights of people in the queue.

### 1.3 Output

Output a single integer, the number of disorder pairs in the queue.

#### Sample Input 1

```
6
1 2 3 4 5 6
```

#### Sample Output 1

```
0
```

#### Sample Input 2

```
5
180 160 175 170 170
```

#### Sample Output 2

```
6
```

## Problem Scale & Subtasks

Test Case No.	Constraints
1-4	$N \leq 1000$
5-8	$N \leq 10000$
9-10	$N \leq 10^6$

## Hint

For C/C++ and Java users, an `int` type stores integers range from -2,147,483,648 to 2,147,483,647. It may be too small for this problem. You need other data types, such as `long` for C/C++ and `long` for Java. They store integers ranging from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. Use `scanf("%lld",&n)` for C, `cin>>n` for C++ and `n = scanner.nextLong()` for Java to get the input  $n$ . And the other operations for `long` and `long long` are quite same as `int`.

## 2 Star Sequence (50% of this assignment)

### 2.1 Description

Renko Usami observes space through a telescope when she notices a fantastic phenomenon – the number of stars in the fields follows a mathematical pattern.

Specifically, let's denote the number of stars in the  $N$ th field by  $f_N$ , then  $f_N$  satisfies the following expression, and  $a, b$  are given positive integers.

$$f_N = af_{N-1} + bf_{N-2} \quad (N \geq 2)$$

Now Renko Usami is curious about how many stars in the  $n$ th field, but the  $n$ th field is too far away to be observed through her cheap astronomical telescope. Since there are so many stars, she only cares about the value of the number of stars modulo  $m$ . In other words, she want to know  $f_n \bmod m$ .

Fortunately, Renko Usami is able to observe the two closest star fields to her, and the numbers of stars in fields are  $f_0$  and  $f_1$ . Unfortunately, she is going to be late again for her appointment with Merry. Can you write a program for her to calculate  $f_n \bmod m$ ?

### 2.2 Input

The only line of the input contains 6 integers  $n(1 \leq n \leq 10^{18})$ ,  $a, b, f_0, f_1(1 \leq a, b, f_0, f_1 \leq 100)$ ,  $m(1 \leq m \leq 2 \times 10^9)$ . – the meanings of these numbers are shown in the problem description.

### 2.3 Output

Output a single integer –  $f_n \bmod m$ .

#### Sample Input 1

4 1 1 1 1 1000
----------------

#### Sample Output 1

5
---

#### Sample Input 2

468908 34 29 33 30 998244353
------------------------------

#### Sample Output 2

829261643
-----------

## Problem Scale & Subtasks

For 100% of the test cases,  $1 \leq n \leq 10^{18}$ ,  $1 \leq a, b, f_0, f_1 \leq 100$ ,  $1 \leq m \leq 2 \times 10^9$ .

Test Case No.	Constraints
1-2	$n \leq 10$
3-5	$n \leq 10^6$
6-10	No additional constraints

## Hint

*Hint1:* For C/C++ and Java users, an `int` type stores integers range from -2,147,483,648 to 2,147,483,647. It may be too small for this problem. You need other data types, such as `long long` for C/C++ and `long` for Java. They store integers ranging from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807. Use `scanf("%lld",&n)` for C, `cin>>n` for C++ and `n = scanner.nextLong()` for Java to get the input  $n$ . And the other operations for `long` and `long long` are quite same as `int`.

*Hint2:* Here's a simple definition of the modulo operation. Your output should be the remainder of the Euclidean division of  $f_n$  by  $m$ , where  $f_n$  is the dividend and  $m$  is the divisor. And for the modulo operation the following equation holds:

$$(x + y) \bmod m = ((x \bmod m) + (y \bmod m)) \bmod m$$

*Hint3:* When  $a$  and  $b$  are 1,  $f_n$  is Fibonacci Sequence. Here is a way to compute the Fibonacci Sequence by using matrices:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f_1 \\ f_0 \end{bmatrix} = \begin{bmatrix} f_2 \\ f_1 \end{bmatrix}$$

The Matrix  $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$  is called transition matrix. Also, note that matrix multiplication is associative. By multiplying the transition matrix  $n - 1$  times, we can get the  $f_n$ .