CSC3100 Data Structures Fall 2022

Programming Assignment I

Due: Nov 4 2022

1 02 Representaion (20% of this assignment)

1.1 Problem Description

Each positive integer n corresponds to a unique binary representation (**specifying that binary numbers are written from right to left, in order of most significant digit to least significant digit**).

For example: $10 = 8 + 2 = 2(3) + 2(1)$

For ease of writing, "a(b)" is used in this question to denote "a to the power of b", i.e., a^b.

Specifying the 02 representation of a number can be obtained by the following procedure:

1. substitute a number that is not 0 or 2 with its binary representation, additionally, using 2(0) instead of 1.
2. check whether the result contains only 0 and 2, and if it contains numbers other than 0 and 2, repeat the step 1 for these numbers

It can be proved that the representation is *unique* after the above steps.

Take 137 as an example:

- First round: $137 = 2(7) + 2(3) + 2(0)$, 7 and 3 do not meet the requirements
- Second round:
  - Substitute 7 and 3 with their binary expressions: $7 = 2(2) + 2 + 2(0)$, $3 = 2 + 2(0)$
  - Thus, $137 = 2(2(2) + 2 + 2(0)) + 2(2 + 2(0)) + 2(0)$

So the "02 representation" of 137 is $2(2(2) + 2 + 2(0)) + 2(2 + 2(0)) + 2(0)$, remember that binary numbers are written from right to left, in order of most significant digit to least significant digit

Your task is to write a program that reads a positive integer n and outputs the 02 representation of the given number

1.2 Problem Description

An integer n, the number to be represented in 02 form (ensure that $1 \leq n \leq 10^9$).

Hint: For C/C++ and Java users, an `int` type stores integers range from -2,147,483,648 to 2,147,483,647. So it is possible to store n with an `int` type.

1.3 Output

One line, the 02 representation of n, (**no spaces between characters**).

Trailing spaces and newlines after the last character are ok.

Sample Input I

```
7
```

Sample Output I

```
2(2)+2+2(0)
```

Sample Input II

```
137
```

Sample Output II

```
2(2(2)+2+2(0))+2(2+2(0))+2(0)
```

Problem Scale & Subtasks

For 30% of the testcases $1 \leq n \leq 200$

For 100% of the testcases $1 \leq n \leq 10^9$

Test Case No.	Constraints
1	$n = 7$
2	$n \leq 200$
3	$n = 137$
4-10	No additional constraints

2 Crossing (20% of this assignment)

2.1 Description

A terminal is a row of n equal segments numbered 1 to n in order. There are two parallel terminals, one above the other.

You are given an array a of length n. For all $i = 1, 2, , n$, there should be a straight wire from some point on segment i of the top terminal to some point on segment a_i of the bottom terminal. You can't select the endpoints of a segment. For example, the following pictures show two possible wirings if $n = 7$ and $a = [4, 1, 4, 6, 7, 7, 5]$.

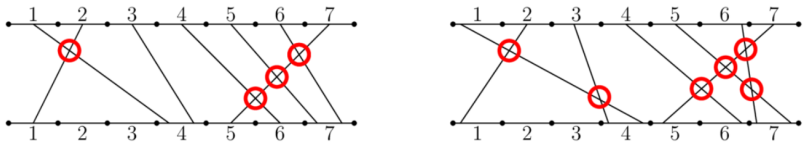


Figure 1: A crossing occurs when two wires share a point in common. In the picture above, crossings are circled in red.

What is the maximum number of crossings there can be if you place the wires optimally?

2.2 Input

The first line contains an integer n ($1 \leq n \leq 10^5$), representing length of the array.

Then follows 1 line with n numbers, representing the array. It is guaranteed ($1 \leq a_i \leq n$)

2.3 Output

Output one integer representing the maximum number of crosses.

Hint (For C/C++ and Java users): The result may exceed the range of `int` type, you are recommended to use `long long` (in C/C++) or `long` (in Java) to store the result.

Sample Input I

```
7
4 1 4 6 7 7 5
```

Sample Output I

```
6
```

Sample Input II

```
3
2 2 2
```

Sample Output II

```
3
```

Problem Scale & Subtasks

For 30% of the testing data, $n \leq 1,000$

For 100% of the testing data, $1 \leq n \leq 100,000$

Test Case No.	Constraints
1-4	$n \leq 1000$
5-8	a is a a permutation, i.e., a contains all integers from 1 to n exactly once.
9-10	No additional constraints

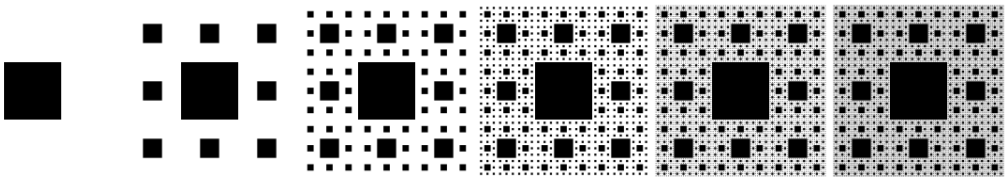
3 Sierpiński carpet (20% of this assignment)

3.1 Description

You are required to write a program to print out the Sierpiński carpet of size $3^k \times 3^k$.

Construction of Sierpiński carpet (quoted from wikipeadia):

The construction of the Sierpiński carpet begins with a square. The square is cut into 9 congruent subsquares in a 3-by-3 grid, and the central subsquare is removed. The same procedure is then applied recursively to the remaining 8 subsquares, ad infinitum.



3.2 Input

A positive integer k, ensure that $1 \leq k \leq 7$

3.3 Output

Sierpiński carpet of size $3^k \times 3^k$. ' ' (`SPACE`) indicates the area is removed, and '`#`' indicates the area remains.

Trailing spaces and newlines after the last character are ok.

Sample Input I

```
1
```

Sample Output I

```
###
# #
###
```

Sample Input II

```
2
```

Sample Output II

```
#########
# ## ## #
#########
###   ###
# #   # #
###   ###
#########
# ## ## #
#########
```

Problem Scale & Subtasks

For 100% of the test cases, $1 \leq k \leq 7$

Test Case No.	Constraints
1-3	$k \leq 3$
4-7	No additional constraints

4 Array Maintenance (30% of this assignment)

4.1 Description

You are required to maintenance an array, which can do the following operations.

1. insert an element with value x after position k. (After this operation, element with value x will be the $k+1$-th element in the array and $k+1$-th element will be moved to position $k+2$ and so on).
2. delete the k-th element in the array. (After this operation, the k-th element will be removed and $k+1$-th element will be moved to position k and so on).
3. Calculate the sum from the l-th element to the r-th element.

In this problem, it is guaranteed that all the number k is randomly generated with equal possibility from all legal values.

(Hint: For a tree structure with n vertex whose root is vertex 1, if vertex i's parent is randomly chosen form $[1, i-1]$, then the expected height of the tree is $O(\log n)$).

4.2 Input

The first line contains an integer n ($1 \leq n \leq 2 \times 10^5$), representing the number of operations.

Then follows n lines, with each line contains several integers. The first integer is the type of operation.

- If it is 1, then follows two integers k ($0 \leq k \leq$ len(array)), x ($1 \leq x \leq 10^9$).
- If it is 2, then follows one integer k ($1 \leq k \leq$ len(array)).
- If it is 3, then follows two integers l, r ($1 \leq l \leq r \leq$ len(array)).

Note: In operation of type 1, if $k = 0$, the new element x is inserted at the very beginning of the array, i.e., after the insertion, x should be the first element of the array.

4.3 Output

For each operation with type 3 output one integer in one line representing the answer of this operation.

Hint (For C/C++ and Java users): The result may exceed the range of `int` type, you are recommended to use `long long` (in C/C++) or `long` (in Java) to store the result.

Sample Input I

```
6
1 0 1
1 0 2
1 1 4
3 1 1
3 2 2
3 1 3
```

Sample Output I

```
2
4
7
```

Sample Input II

```
10
1 0 2
1 1 5
1 1 4
1 1 1
3 1 3
2 3
3 1 3
1 2 9
2 1
3 1 3
```

Sample Output II

```
7
8
15
```

Problem Scale & Subtasks

For 100% of the testing data, $1 \leq n \leq 2 \times 10^5$

Test Case No.	Constraints
1-3	$n \leq 2,000$
4-6	there's no operation with type 2.
7-10	No additional constraints

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 runnable 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.

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

Report (10%)

You also need to write a report to explain the following:

- What are the possible solutions for the proble?
- 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 I	Sample Output I
1 2	3

Problem Scale & Subtasks

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

Solutions

Java

```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

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

C

```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++

```cpp
#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 & report to Black Board.

C.1 Online Judge

Once you have completed one problem, you can submit your code on the page on the Online Judge platform (cuhkszoj.com, campus only) in order to gain marks for the code part. You can submit your solution of one problem for **no more than 30 times**. After you have submitted your program, OJ will test your program on all test cases and give you grade. The grade of your lastest submission will be regarded to as the final grade of the corresponding problem. Each problem are 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 on the online judge system. OJ will show the time and memory limits for different languages on the corresponding problem page.

OJ access code: `CSC3100assignment2`, you are using this code whenever you are asked to do so.

If you have other questions about the online judge system, please refer to OJ wiki (campus network only). And if this cannot help you well, 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 `A2_<Student ID>.zip` :

```
A2_<Student ID>.zip:
    A2_P1_<Student ID>.java/py/c/cpp/cc
    A2_P2_<Student ID>.java/py/c/cpp/cc
    A2_P3_<Student ID>.java/py/c/cpp/cc
    A2_P4_<Student ID>.java/py/c/cpp/cc
    A2_Report_<Student ID>.pdf
```

For example, suppose your ID is 123456789, and your problem 1 is written in `Python`, problem 2 is written in `Java`, problem 3 is written in `C` and problem 4 in `C++`, then the following contents should be included in your submitted `A2_123456789.zip`:

```
A2_P1_123456789.py
A2_P2_123456789.java
A2_P3_123456789.c
A2_P4_123456789.cpp
A2_Report_123456789.pdf
```

Note

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

- Problem 1. Derong Jin: 120090562@link.cuhk.edu.cn
- Problem 2. Qingshuo Guo: 121090151@link.cuhk.edu.cn
- Problem 3. Derong Jin: 120090562@link.cuhk.edu.cn
- Problem 4. Qingshuo Guo: 121090151@link.cuhk.edu.cn
- OJ issues: Tianci Hou: 121090184@link.cuhk.edu.cn