Some Tips for Taking Collegiate Programming Examination (CPE)

Lab 9: Arithmetic for Skewed Binary

Rung-Bi Lin
csrlin@saturn.yzu.edu.tw

International Bachelor Program in Informatics
Yuan Ze University

11/21/2023

Collegiate Programming Exam (CPE)

- Programs are developed for some given problems online and judged automatically. The purpose is to evaluate the programming competence of students.
- Exam's are normally held in March, June, September, and December. They are free of charge. More than 2000 students from about 40 universities in Taiwan took the exam each time. Students can freely select a test site wherever is nearer to them.
- Every student must program alone on a computer which does not have networking capability. No materials except English Dictionary can be brought into the exam room.
- **Web site: https://cpe.cse.nsysu.edu.tw/**

CPE Problems

- There are seven problems with different levels of difficulty. Three of them are easy ones.
- Except the public test data sets, there are some hidden test data sets. The public data sets can be used to help debug a program.
- There are five difficulty levels denoted by the number of star symbol *. The easiest problems are denoted by one star symbol. Each exam will have at least one problem out of about 40 one-star problems being included into the problem set. Please refer to the following links for details. http://cpe.cse.nsysu.edu.tw/environment.php#starList

Coming Exam

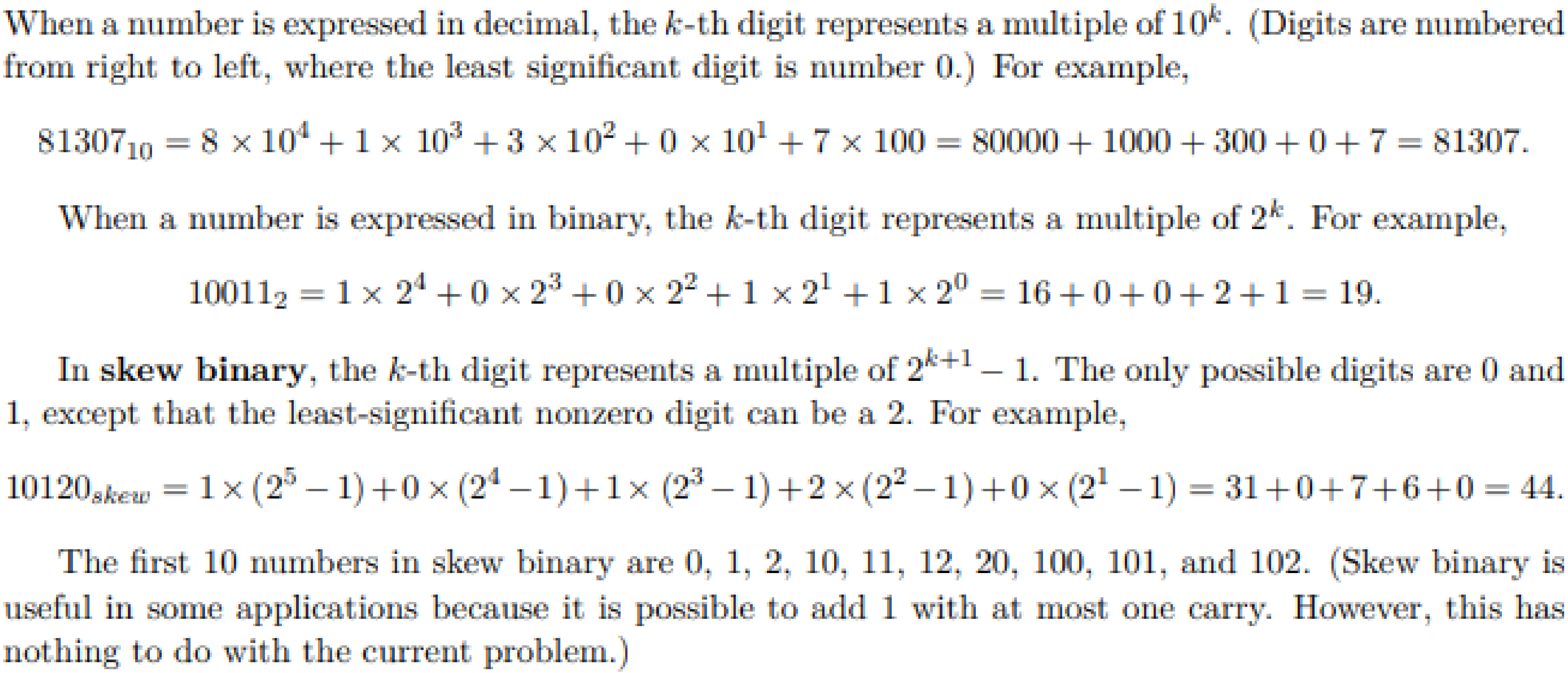
- Exam date: 2023/12/12
- Starting date and time for registration: 2023/11/28, 14:25
- Ending date and time for registration: 2023/12/09,18:00
- Exam time
  - 17:30-17:40 for entering exam room，no admission after **18:00**
  - 17:40-18:30 for practice
  - 18:40-21:40 for exam
- Place: R1008, R1201A, R1301B (select during registration)
- Programming and evaluating platform
  - CodingFrenzy http://coding-frenzy.arping.me/

Some Rules

- Must obtain an account before you can register for a test. : https://cpe.cse.nsysu.edu.tw/cpe/
- If you are absent for a test without permission, your right for next test will be suspended.
- If you cheat, the test score is zero and your right for next three tests will be suspended.

Example of a Problem

(UVA 1565)



When a number is expressed in decimal, the k-th digit represents a multiple of 10^k. (Digits are numbered from right to left, where the least significant digit is number 0.) For example,

$$81307_{10} = 8 \times 10^4 + 1 \times 10^3 + 3 \times 10^2 + 0 \times 10^1 + 7 \times 100 = 80000 + 1000 + 300 + 0 + 7 = 81307.$$

When a number is expressed in binary, the k-th digit represents a multiple of 2^k. For example,

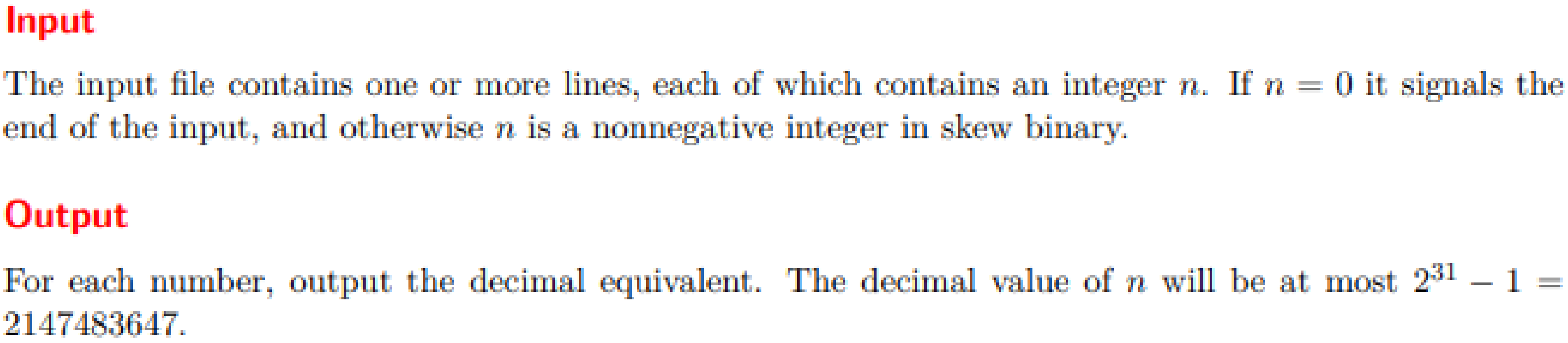
$$10011_2 = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 16 + 0 + 0 + 2 + 1 = 19.$$

In **skew binary**, the k-th digit represents a multiple of $2^{k+1} - 1$. The only possible digits are 0 and 1, except that the least-significant nonzero digit can be a 2. For example,

$$10120_{skew} = 1 \times (2^5 - 1) + 0 \times (2^4 - 1) + 1 \times (2^3 - 1) + 2 \times (2^2 - 1) + 0 \times (2^1 - 1) = 31 + 0 + 7 + 6 + 0 = 44.$$

The first 10 numbers in skew binary are 0, 1, 2, 10, 11, 12, 20, 100, 101, and 102. (Skew binary is useful in some applications because it is possible to add 1 with at most one carry. However, this has nothing to do with the current problem.)

Input & Output



Input

The input file contains one or more lines, each of which contains an integer n. If $n = 0$ it signals the end of the input, and otherwise n is a nonnegative integer in skew binary.

Output

For each number, output the decimal equivalent. The decimal value of n will be at most $2^{31} - 1 = 2147483647$.

Sample Input

```
10120
200000000000000000000000000000
10
1000000000000000000000000000000
11
100
11111000001110000101101102000
0
```

Sample Output

```
44
2147483646
3
2147483647
4
7
10411107
```

Tips for Solving This Problem

- Skew binary number is a positional number system.
- Treat the input skew binary number as a string, said **skewStr**.
  - A string is an array of characters
  - The digits of 10120 have their place values 31,15, 7, 3, and 1, respectively. The equivalent decimal number is 1*31+0*15+1*7+2*3+0*1=44.
- Starting from the least significant digit, use a for loop to go through every digit and sum the digit*placeValue.
  - Need to know where the least significant digit is stored in the character array. It is at highest position of skewStr.
  - Convert a character in skewStr into a digit by skewStr[i] –'0'.
  - Keep track of the place value at each place, which is 2^{k+1}-1 where *k* is a place number. For programming, it would be easier to keep track of 2^{k+1}. With 2^{k+1}, 2^{k+1}-1 can be obtained immediately.

Basics of Reading Test Data

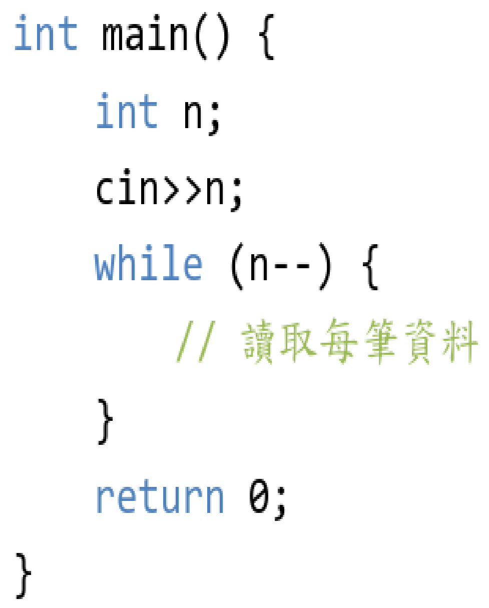
- Typically, one of the following three ways of reading test data into a program is used.
  - Clearly specifies the number of data items that will be read.
  - Read data items until end of input is met. That is, after the last item is read.
  - Read data items until a particular number or symbol is met.

Reading n Data Items

Input

```
3
10
31
50
```

Corresponding code



```
int main() {
    int n;
    cin>>n;
    while (n--) {
        // 讀取每筆資料
    }
    return 0;
}
```

```
int main() {
        Int n;
        cin >> n
        while (n > 0){
        // do something here
        n = n-1;
        }
        return 0;
}
```

```
int main() {
        Int n;
        cin >> n
        for (int i=0; i<n; i++){
        // do something here
        }
        return 0;
}
```

Problem Example of Reading *n* Data Items

- Problem Title: Summing two numbers
- Calculate the sum of two numbers on an input lines
  - Input
    - The first line gives the number of data items. Starting from the second line, every line presents one data item.
    - Every data item consists of two numbers separated by a space character.
  - Output

  Each line shows the result of adding the two numbers.

Input

```
3
10 20
33 25
41 64
```

Output

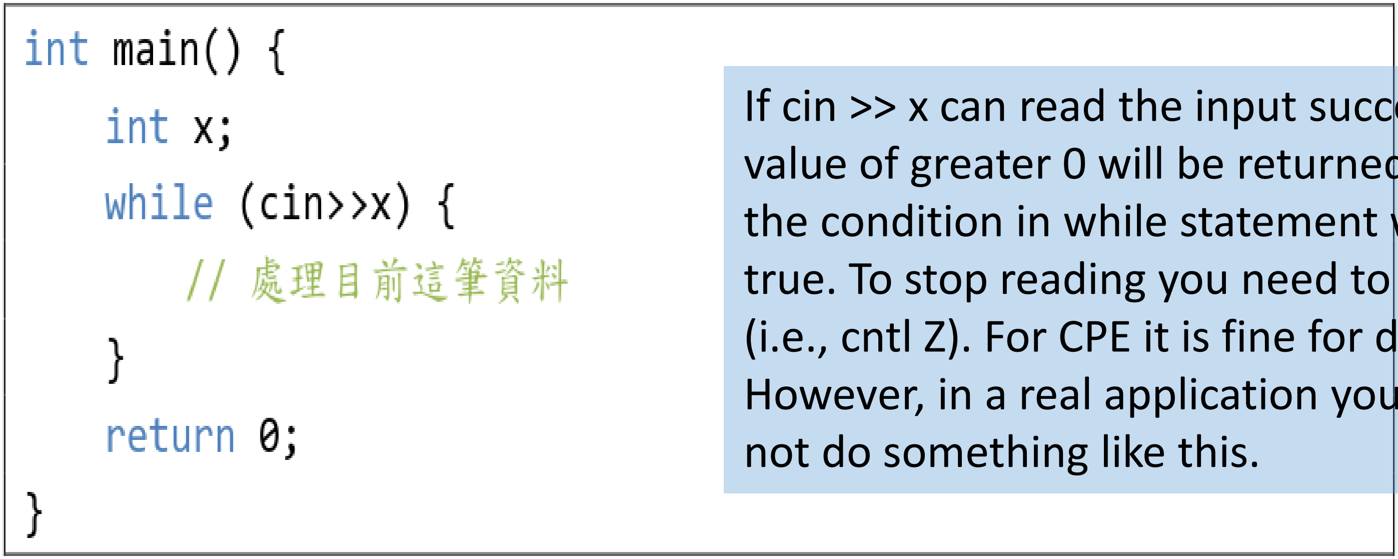
```
30
58
105
```

Reading Data till the End of Input

Input

```
10
31
50
```

Corresponding code



```
int main() {
    int x;
    while (cin>>x) {
        // 處理目前這筆資料
    }
    return 0;
}
```

If cin >> x can read the input successfully, a value of greater 0 will be returned. Hence, the condition in while statement will be true. To stop reading you need to press a ^Z (i.e., cntl Z). For CPE it is fine for doing so. However, in a real application you should not do something like this.

Problem Example of Reading Data Items till End of Input

- Problem Title: Summing two numbers
- Calculate the sum of two numbers on an input lines
  - Input
    - Start from the first line, every line presents one data item. Continue reading data items till the end of input.
    - Every data item consists of two numbers separated by a space character.
  - Output

  Each line shows the result of adding the two numbers.

Input

```
10 20
33 25
41 64
```

Output

```
30
58
105
```

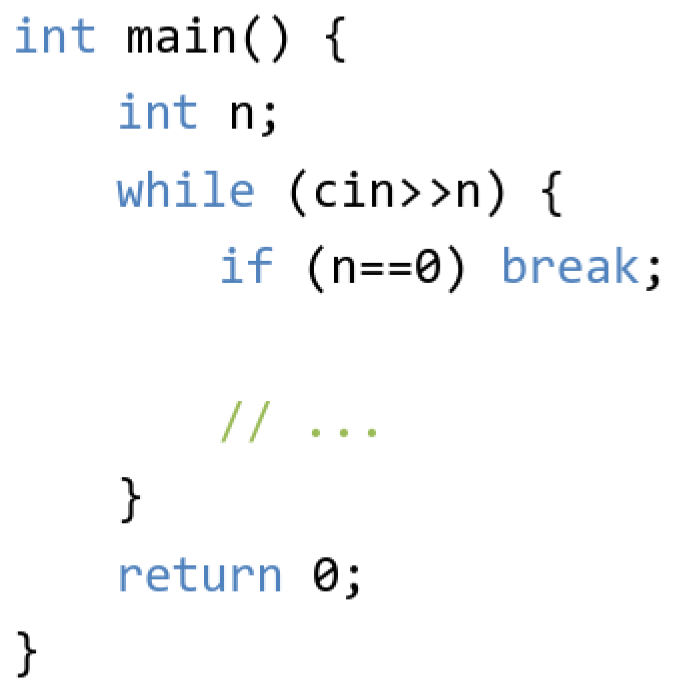
Reading Data Items till 0's Are Encountered

Input

```
10
31
50
0
```

```
int main() {
        Int n;
        cin >> n;
        while (n != 0){
        // do something here
        cin >> n;
        }
        return 0;
}
```

Corresponding Code



```
int main() {
    int n;
    while (cin>>n) {
        if (n==0) break;

        // ...
    }
    return 0;
}
```

```
int main() {
        Int n;
        while (cin >> n && n != 0){
        // do something here
        }
        return 0;
}
```

Problem Example of Reading Data Items till 0’s Are Met

- Problem Title: Summing two numbers
- Calculate the sum of two numbers on an input lines
  - Input
    - Start from the first line, every line presents one data item. Continue reading data items till two 0’s are read.
    - Every data item consists of two numbers separated by a space character.
  - Output

  Each line shows the result of adding the two numbers.

Input

```
10 20
33 25
41 64
0 1
0 0
```

Output

```
30
58
105
1
```

CPE Web Site

- Collegiate Programming Exam
  - https://cpe.cse.nsysu.edu.tw/
- It uses 「瘋狂程設」評判系統 (codingFrenzy System online judge system)
  - You have to get an account by http://coding-frenzy.arping.me/

Lab 9: Arithmetic for Skew Binary

Problem description

Add two skewed binaries and print the resulting skewed binary. You can refer to the slides given in this lecture note for the background about skewed binary.

Input

The first line of input specifies the number of test cases. Each test case takes a single line containing two skewed binaries separated by an unknown number of blank characters, where each skewed binary has no more than 62 digits.

Output

For each test case, print the resulting skewed binary on a single line. An output line should start with a ‘#’ and then is followed by three blank characters and then the resulting skewed binary. Every 10 digits should be followed by a blank character.

Sample Input & Output

Sample Input

```
10
1101 111102
101 11101120
111200 101012
11111101111111111111112 10110101001001011 2
1001110001111111001001 10111111110101011010010100010101 1112
11100000101010010100100100101010010010 1010010101001000001010010101010010101200
1101001011010000111110101001001010100100100010 0102 100110110101001010010001010001011111112
100000000000000000000000000000000000000000000000000000000000000 100000000000000000000000000000000000000000000000000000000000000
111111111111111111111111111111111111111111111111111111111111112 111111111111111111111111111111111111111111111111111111111111112
200000000000000000000000000000000000000000000000000000000000000 200000000000000000000000000000000000000000000000000000000000000
```

Sample Output

```
#   1000200
#   11110012
#   1101001
#   1000000001 1010100100 01011
#   1011111111 0110000001 0111000100 012000
#   1101110101 1100100111 1011100111 1100111120
#   1101001011 1001000110 0100111001 0011001101 0100011012
#   2000000000 0000000000 0000000000 0000000000 0000000000 0000000000 00
#   1111111111 1111111111 1111111111 1111111111 1111111111 1111111011 120
#   1120000000 0000000000 0000000000 0000000000 0000000000 0000000000 000
```

Tips for Solving This Problem

- To solve this problem, you may need to know how to convert a decimal number to a skewed binary.
- It may be easier to process the input data if the skewed binaries are input as strings.