Interactive, output only & Communication Task

Alex Poon 08/04/2017

Agenda

- Interactive Task
- Output only Task
- Communication Task
- Examples

Batch Task

- Most of the problems you have encountered are batch task
- We can read all input data before process in batch task
- But there is some non-batch task
- For example, interactive task, output only task, communication task

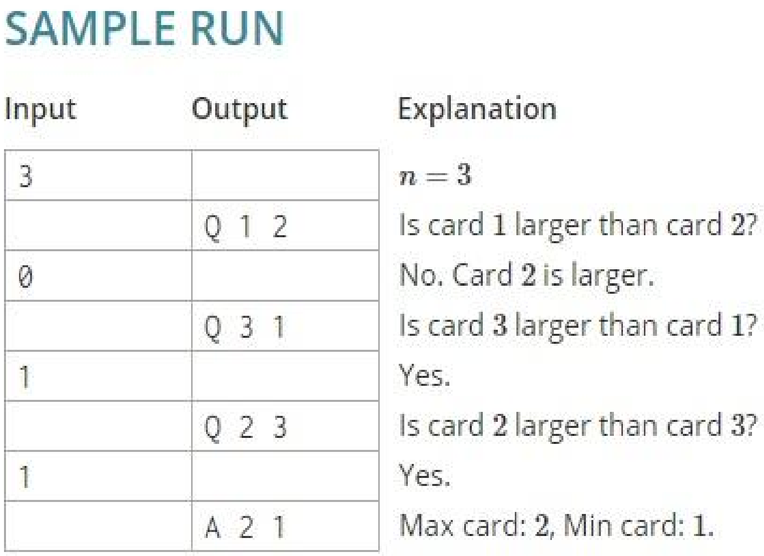
Interactive Task

- Your program will interact with a judging program
- More precisely, your program will ask problems(queries)
- The judging program will answer your queries
- You need to solve the problems by the answer of queries
- Usually, there are constraints limiting how many times you can query

Example - Comparing Game

- HKOJ M1431
- Given N, there are N unknown integers A[1] … A[n]
- You can ask the judging program : is A[i] greater than A[j] ?
- The judging program will answer : Yes or No
- You need to find i, j where A[i], A[j] is the maximum & minimum among all N number
- You can query at most 1.5N times

Comparing Game



SAMPLE RUN

Input	Output	Explanation
3		$n = 3$
	Q 1 2	Is card 1 larger than card 2?
0		No. Card 2 is larger.
	Q 3 1	Is card 3 larger than card 1?
1		Yes.
	Q 2 3	Is card 2 larger than card 3?
1		Yes.
	A 2 1	Max card: 2, Min card: 1.

The problem guide you how to interact with the judge program

In this problem, you should output "Q x y" when make the following query is card x larger than card y

You should flush the buffer before reading the answer of the query

Sample Partial Solution - Pascal

```
readln(n);
For i := 1 to n do begin
  largest := True;
  For j := 1 to n do
  if i <> j then begin
    writeln(‘Q ’, i, ‘ ’, j);
    flush(output);
    readln(ans);
    if ans = 0 then largest := 0;
  end;
  if largest = 1 then writeln(‘A ’, i);
end;
```

Explanation

// Output your query

// flush the buffer

// read the answer given by the judging program

// Output your answer

Sample Partial Solution - C++

```
scanf(“%d”, &n);
for(int i = 1; i <= n; i++) {
    bool largest = 1;
    for(int j = 1; j <= n; j++)
    if (i != j) {
        printf(“Q %d %d\n”, i, j);
        fflush(stdout);
        scanf(“%d”, &ans);
        if (ans == 0) largest = 0;
    }
    if (largest == 1) printf(“A %d\n”, i);
}
```

Explanation

// Output your query
// flush the buffer
// read the answer given by the judging program

// Output your answer

Interactive Task

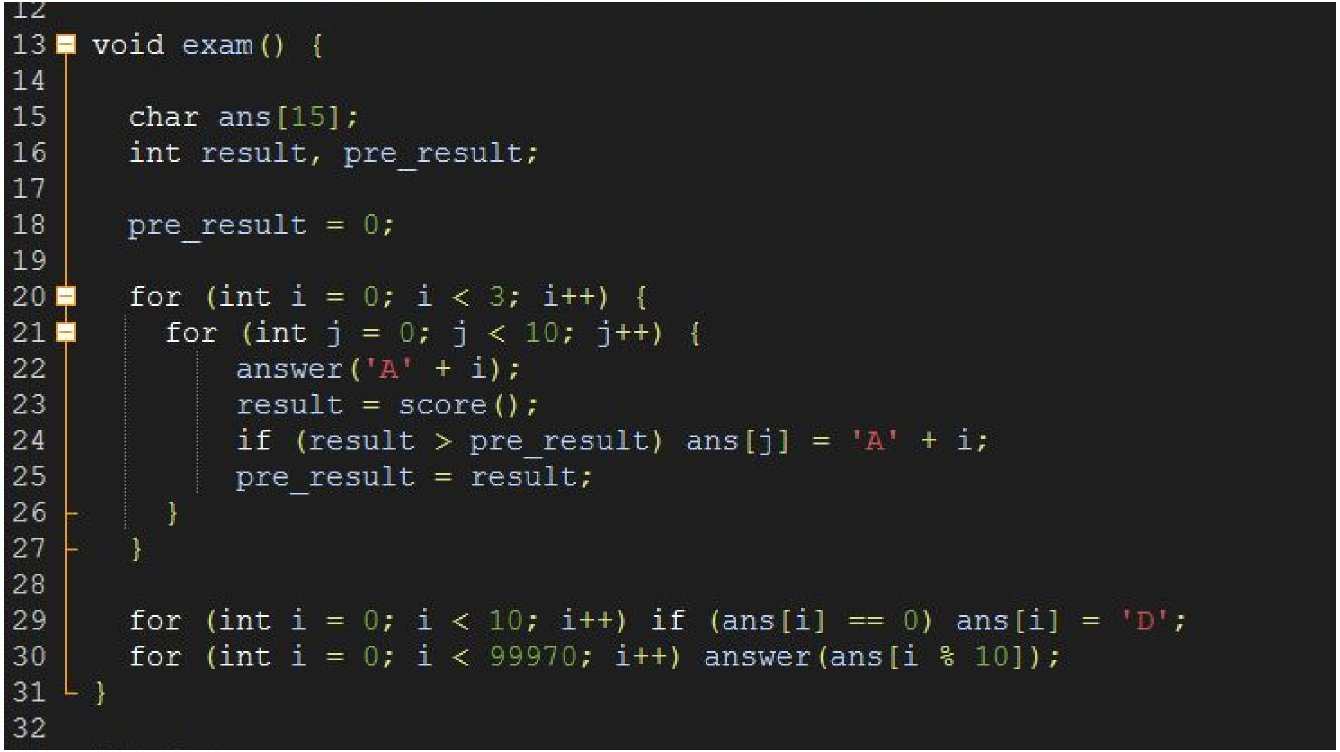
- Other than interaction through standard I/O (Codeforces use)
- Another type of interaction : through library (IOI/NOI/HKOI use)
- The library includes some functions for you to call for query & answer
- Usually, the problem statement will state how to use the functions for query & answer

Example: Model Answer II (T163)

- For interactive task that use library as interface
- A template with several function (score(), answer() in this example) will be given
- Explanation of function, sample run will also be stated clearly

- Link to problem statement: https://judge.hkoi.org/task/T163

Example: Model Answer II (T163)



```
void exam() {

    char ans[15];
    int result, pre_result;

    pre_result = 0;

    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 10; j++) {
            answer('A' + i);
            result = score();
            if (result > pre_result) ans[j] = 'A' + i;
            pre_result = result;
        }
    }

    for (int i = 0; i < 10; i++) if (ans[i] == 0) ans[i] = 'D';
    for (int i = 0; i < 99970; i++) answer(ans[i % 10]);
}
```

Tips: Interactive Task

- Actually, no tips for interactive task as the scope of interactive task is wide
- T163 (Ad-hoc, optimization)
- I1321 (Binary search)
- I1222 (Greedy)
- I1413 (Graph)

- The most important tips: Interactive task may not be difficult!!!

Output Only Task

- All input data is open
- You only need to submit the output file of your program (.out / .txt instead of .cpp / .pas)
- i.e. No time limit, memory limit
- Usually, there are no fast & optimal solution
- But require a near-optimal solution
- The marks you get base on the accuracy of your program
- There is usually a formula for calculating your score

Example - Maze

- IOI 2010 Day Q3

- Given a N * M grid, some of them a occupied
- Construct a simple path (pass through K cells) in the grid s.t.
- 1. All selected cells (except the two end) of the path connect to exactly 2 selected cells only
- 2. The end of the path connect to exactly 1 selected cell
- 3. Exactly one of the two end should touch the boundary of the grid
- 4. Other cell should not touch the boundary of the grid

- Link to problem: https://judge.hkoi.org/task/I1023

Example - Maze

Input 1
(Not sample input, it is real dot test case 1)

```
##X#######
###X######
####X##X##
##########
##XXXX####
##########
```

Output 1 (Example only)
(Not sample input, it is real dot test case 1)

```
##X#######
###X##...#
#...X#.X.#
#.#....#.#
#.XXXX##.#
########.#
```

Length = 17
Score = 10^(17/20) = 7.07 for this case

Tips for output only task

- Output only task usually does not have optimal and fast solution
- You should balance the accuracy / speed of your program
- Study the scoring formula carefully
- Study the input data carefully !!! -> it may contain some special pattern!!!
- For small case, do it by hand!!
- Usually easy to get partial score (even 10 points is very important in HKOI - TFT)

- Often used approach:
  1. Greedy
  2. Exhaustion + branch and bound
  3. Randomize algorithm

Communication Task

- You need to write 2 programs (or 2 modes)
- Inputs of the 2 programs are different
- Usually, program A will get more useful input data
- In program A, you need to encrypt the useful data to a 01 string
- Then the 01 string will be inputted to program B
- Program B most solve the original problem 01

- Score depends on the length of the 01 string you send

Communication Task

- Flow of communication task
- Run program A
- The output of program A = Input of program B
- Run program B
- Grade your program by output of program A/B

Dividing the cities (HKOI 2014)

- Given N nodes and M edges and 10 colours
- Construct a way to colour the nodes s.t. no two nodes connected by an edge has same colour
- N <= 3000, M <= 10000
- Link to problem : https://judge.hkoi.org/task/S141

Dividing the cities

Program A:

- Read the node and edge and a valid way to colour the node
- Output a 01 string with length L

Program B:

- Read the node and edge and the 01 string
- Output a valid way to colour the node

Mark depends on L

Dividing the cities

Trivial Solution:

Program A: Encode the whole answer provide to an 01 string
Program B: Decode the 01 string back to the whole answer

e.g. The answer provide is 1 2 3 1 9

We may use 4 bits to encode 1 answer (2^4 = 16 > 10)
1 -> 0001, 2 -> 0010, 3 -> 0011, 9 -> 1001.
The 01 string become : 00010010001100011001
In practical, this solution use 4 * N bits and get 25 marks

Dividing the cities

- We should reduce length of the 01 string in order to get higher marks

- We have 2 direction to reduce L
    - Reduce the bits we need to encode the same number of data
    - Reduce the number of data we need to encode

Dividing the cities

- Optimization 1

- We use 4 bits (can represent 16 comb.) to encode 1 answer(10 comb.)
- It is waste
- Better to use 10 bits (can represent 1024 comb.) to encode 3 answer (1000 comb.)

e.g. Answer is 10 3 5 -> we first convert it to 0 base -> 9 2 4 -> 924

- Use 10 bits to represent 924 by change it to binary
- Can get 42 marks

Dividing the cities

- Optimization 2

- Note that if a node has < 10 neighbours
- No matter what the colour of its neighbours is
- There must exist a colour s.t. no collision occur

Therefore, if a node has < 10 neighbours, we don’t need to pass its answer from prog A to prog B

Dividing the cities

- There are at most 10000 edges
- Which mean there are at most 10000 * 2 / 10 = 2000 nodes has >= 10 neighbour
- Combine 2 optimizations, we use only 2000 / 3 * 10 = 6667 bits
- We can get 70 marks

Communication Task

- Usually, we need some observation to reduce number of data we need to send
- But we have some standard way to reduce the bits we need to encode same amount of data

Method 1 - Compress multiple data to one

- Just like the example above
- Use 4 bits to encode 1 colour -> Use 10 bits to encode 3 colours

Method 2 - Reduce leading zero

- e.g. We need to encode A[1..N] where A[i] <= 2^20
- We need 20 bits to encode 1 data

Improvement

- e.g. if the number < 1024, use 10 bits only, else use 20 bits
- then how to encrype 1 → 00000 00001 → fewer bits =]

- Wait, when I am decrypting (in Program B), how do I know I use 10/20bits to encrype this numbers ?

Method 2 - Reduce leading zero

- Add a signal
- e.g. If the first bit is 0, then the following 10 bits are in 1 group
- If the first bit is 1, then the following 20 bits are in 1 group

- How to encrype 1 → 0 00000 00001
- How to encrype 1000000 → 1 11110 10000 10010 00000

- The main point is to where is the cut off … < 1024 ?? < 4096 ??
- And how many cut off you should add

Tips for Communication Task

- Important technique, decimal <-> binary (bases converting)
- No other special tips again as the scope of communication task is wide
- Sometimes (should be Usually) we need adhoc observations
- Easy to get partial score but difficult to full

Practise

- The best way to perform better in non-batch task is to practise!!!

M1431, T054, T113, T124, T134, T144, S141, I1321, I1011, I1021, I1023, ……

1. 調皮的小孩 : download.noi.cn/T/noi/noi2002A.pdf
2. Towns : http://olympiads.kz/ioi2015/day2/towns-en.pdf
3. Parrots : http://www.ioi2011.or.th/hsc/tasks/EN/parrots.pdf
4. Rail: http://www.ioinformatics.org/locations/ioi14/contest/day1/rail/rail.pdf

Example 1 - Celebrity (HKOJ 01084)

- Link to problem statement: https://judge.hkoi.org/task/01084

- There are N people and exactly one of them is celebrity
- Everyone knows the celebrity and the celebrity knows none of them
- You can use Q(i, j) to ask the system is people[i] knows people[j]?
- Your task is to find who is the celebrity
- You should query no more than 1.5N times

- Idea?

Example 1 - Celebrity (HKOJ 01084)

- Observe that if x knows y, then x must not be the celebrity
- If x does not know y, then y must not be the celebrity

- We can eliminate one potential celebrity by one query
- So, we need N - 1 query only

- Interactive task is EASY!!!!

Example 2 - Model Answer II (HKOI TFT 2016)

- Link to problem statement: https://judge.hkoi.org/task/T163

- Let brainstorm together~

- Strategy 1:
- For each position, we make 3 guesses (A, B, C)
- After each guesses, we call the score function to check whether it is correct
- If all 3 guesses wrong, we know the answer is ‘D’
- So, we can get the correct sequence within 30 call of score() function
- We can get at least 99970 correct answers and we need 30 call of score() function
- 20 marks

- Any better idea?

Example 2 - Model Answer II (HKOI TFT 2016)

- Observe that the formula gives more penalty to an additional call of score function
- Try to reduce calling score function, although you may get fewer correct guess

- Strategy 2
- We will call score function for 3 times
- We guesses ‘A’ for 1 times in pos 1, 2 times in pos 2, 4 times in pos 3 … 2^(i-1) in pos i
- Then we call score function
- The binary of the score reflects which position we guessed correctly
- e.g. score = 19, we know we guess position 1, 2, 5 correctly as (1 + 2 + 16 = 2^0+2^1+2^4 = 19)
- Do the same for guessing ‘B’ and ‘C’
- Score : ~40

- Think other solution yourselves~

Summary

- Interactive and communication task often appear in contest (e.g. HKOI - TFT, NOI, IOI)

- You should know how to write a compile-able / run-able program for these tasks

- The scope of these tasks are wide, no special training is needed

- You are recommended to do 1 - 2 tasks (no need to get full marks but to get partial score) before HKOI-TFT as based on historical data, Interactive / communication tasks often appear