Assignment for “Algorithm Design and Analysis”

Abstract:

1. The candidate must complete four tasks. The candidate is rewarded for successfully completing five tasks.
2. For each task, the submission must include five parts: i) the analysis for the proposed solution, ii) the time complexity analysis for the algorithm you presented; iii) the source code using JAVA or C++; iv) the screenshots of running your program; and v) the test result analysis.
3. The soft copy is due by midnight on **April 17, 2016** via email (**computationalintelligence@yahoo.com**). The hard copy is due by 9:55am on **April 18, 2016**.
4. The candidate cannot pass this Unit if the submissions are behind the above deadline, or there are accusations of plagiarism.

---

1. TASK ONE

“Joseph” is a famous problem. There are n people standed in a circle. They are numbered as $1, 2, \cdots, n$. Then, they number off, starting with the first person whose number is 1. And the one whose number is a multiple of m, will be knocked off the circle. This process is repeated until there is only one person left in the circle. For example, when $n = 6$ and $m = 5$, those persons knocked off the circle, are ranked with their numbers by $5, 4, 6, 2, 3$. And the last person in the circle is the one whose number is 1. Now, when $m = 2$, your task is to find out how many of n ($1 \leqslant n \leqslant 2005$), such that the person whose number is 1 is the last one left in the circle. For example, when $n = 2^m$ ($m = 0, 1, 2, 3, \cdots$), there are 11 within $[1, 2005]$.

2. TASK TWO

There are total N stations in a Metro Line. Those stations can be divided into three types: U (ground), D (underground), and C (composite). The construction cost of each station differs throughout the Metro Line. It should be noted that the adjacent station type can not be repeated. Given the construction costs of three types for each station, you are required to find out the minimum cost of constructing this Metro Line. For example,

	U	D	C
Station 1	1	9	9
Station 2	9	1	9
Station 3	9	9	1

The minimum cost is: 1+1+1=3.

Now please input N stations and the corresponding construction costs of three types for each station, and give the optimal planning by using Dynamic Programming method.

Edited April 1, 2016.

3. TASK THREE

In a Minesweeper computer game, there is a $P \times P$ field with some mines, where P is the size and $3 \leqslant P \leqslant 2016$. We do not know the location of those mines. In the field, each square contains a number (1 to 9), and each number is the total number of mines in this square and the adjacent squares. Now you are required to design an algorithm to find out how many mines in this $P \times P$ field.

4. TASK FOUR

Assume we have a sequence that contains N numbers of type long. And we know for sure that among this sequence each number does occur exactly n times except for the one number that occurs exactly m times ($0 < m < n$). How do we find that number with $O(N)$ operations and $O(1)$ additional memory?

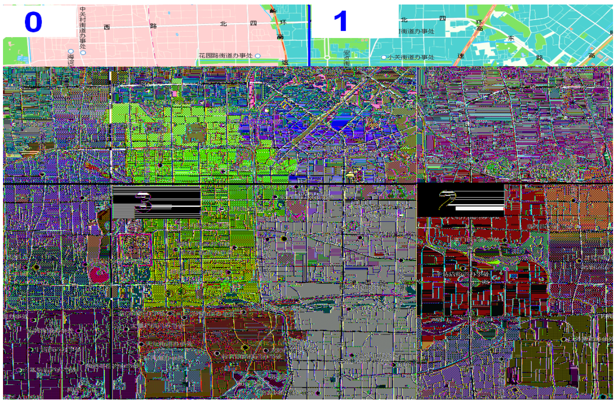
Example: We know that $n = 3$ and $m = 2$. The sequence ($N = 8$) is:

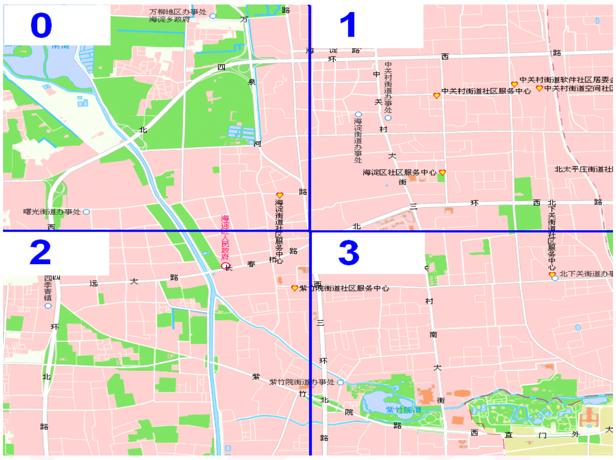
5 11 5 2 11 5 2 11

And the right answer is 2 in this particular case because it occurs only twice.

5. TASK FIVE

Mr. Algorithm decides to develop a web site, which will provide the service of transit search. But he can only get the transit data of Beijing, so his web site can only support the transit search of Beijing. We suppose Beijing is 10240 meters by 10240 meters. The coordinate of the top-left corner is $(0, 0)$. The coordinate of the bottom-right corner is $(10240, 10240)$. The X-axis is from top to bottom and the Y-axis is from left to right. At the beginning, four pictures of the size 10 cm by 10 cm make up of the whole map of Beijing. They are numbered from 0 to 3. It is





to say at the beginning the scale of the map is 1 cm:512 meters. We call the four pictures are at level 1.

When you double-click on the map using the mouse, the map will zoom in. The pictures at next level will be shown on the screen. For example, when you double-click on the above map, picture 0 will be replaced by four pictures $00, 01, 02, 03$, all of whose sizes are 10 cm by 10 cm, and the scale of the map changes to 1 cm:256 meters. Notice that, pictures $00, 01, 02, 03$ together describe the same area as picture 0. When you continue double-click, picture 01 will be replaced by pictures $010, 011, 012, 013$, and so on. Now, a position's coordinate can be given by (str, x, y). 'str' consists of 8 characters each from 0 to 3. It describes the *id* of the picture in which the position is located at x and y (0 cm$\leqslant x, y \leqslant$ 10 cm) describe the position's offset relative to the top-left corner on picture str. Notice that the X-axis is from top to bottom and the Y-axis is from left to right.

Now, the start position and end position are given as $(start, sx, sy)$, (end, ex, ey). And some information about the bus line will be also given. First, each bus stop will be described by $(name, x, y)$, which is its name and its coordinate. Second, each bus line will be described by $(name_1, name_2, name_3, \cdots, name_k)$ which are the bus stops the bus line travels through. If the distance between the start position and end position is no more than 2000 meters, the web site will suggest walking there. Otherwise, the web site will find a bus stop whose distance is no more than 1000 meters from the start position. You can take buses to a bus stop whose distance is no more than 1000 meters from the end position. Along the way, you can change buses at any bus stop. If you can take buses according the above rules, the web site will find a route with fewest number of changing buses. If you can not take buses according the above rules, the web site will suggest taking a taxi.