# 《Analysis and Design of Algorithm》 — Projects and Requirements

***Requirements:***

1. Based on the following problem-list, choose one to do algorithm analysis and design.
2. You can finish it alone or form a team, **at most 3 students**, and once the team is successfully organized, no change!!! **Careful**: you need to complete the team by **June 4th, 2020**.
3. You need to submit a report and make a presentation (live or video) with PPT, upload a “studentID-groupName (studentName)-problemName.zip” including:
   1. **PPT with audio or PPT**(if you choose live presentation): online presentation or recorded video ***at most 10 minutes***.
   2. **Experiment report**: should follow the experiment report template, ***make sure the content is complete, the thinking is clear and the algorithm analysis and design steps are available (****pay attention to the 5 characteristics of algorithm****).***
4. The deadline: **June 11th, 2020. At the same time 13:30~16:00 as our examination time, we will arrange the presentation, please prepare for it.**

# Problem 1: A tomato a day

Professor Kerry loves tomatoes! The professor eats one tomato every day, because she is obsessed with the health benefits of the potent antioxidant lycopene and because she just happens to like them very much, thank you. The price of tomatoes rises and falls during the year, and when the price of tomatoes is low, the professor would naturally like to buy as many tomatoes as she can. Because tomatoes have a shelf-life of only ***d*** days, however, she must eat a tomato bought on dayon some dayin the range , or else the tomato will spoil and be wasted. Thus, although the professor can buy as many tomatoes as she wants on any given day, because she consumes only one tomato per day, she must be circumspect about purchasing too many, even if the price is low.

The professor’s obsession has led her to worry about whether she is spending too much money on tomatoes. She has obtained historical pricing data for n days, and she knows how much she actually spent on those days. The historical data consists of an array , where is the price of a tomato on day . She would like to analyze the historical data to determine what is the minimum amount she could possibly have spent in order to satisfy her tomato-a-day habit, and then she will compare that value to what she actually spent.

**Give an efﬁcient algorithm** to determine the optimal ofﬂine (20/20 hindsight) purchasing strategy on the historical data. Given , , and , your algorithm should output , where is the number of tomatoes to buy on day .

# Problem 2: Spam distribution

Professor Hormel is designing a spam distribution network. The network is represented by a rooted tree with root and nonnegative edge-weight function . Each vertex represents a server with one million email addresses, and each edge represents a communication channel that costs dollars to purchase. A server receives spam precisely if the entire path from the root to is purchased. The professor wants to send spam from the root to servers (including the root) by spending as little money as possible.

Help the professor by **designing an algorithm that ﬁnds a minimum-weight connected subtree** of with vertices including the root. (For partial credit, solve the problem when each vertex has at most 2 children in .)

# Problem 3: Planning a company party

Professor Stewart is consulting for the president of a corporation that is planning a company party. The company has a hierarchical structure; that is, the supervisor relation forms a tree rooted at the president. The personnel ofﬁce has ranked each employee with a conviviality rating, which is a real number. In order to make the party fun for all attendees, the president does not want both an employee and his or her immediate supervisor to attend.

Professor Stewart is given the tree that describes the structure of the corporation, using the left-child, right-sibling representation. Each node of the tree holds, in addition to the pointers, the name of an employee and that employee’s conviviality ranking.

**Describe an algorithm** to make up a guest list that maximizes the sum of the conviviality ratings of the guests. Analyze the running time of your algorithm.

# Problem 4: The Data Center

The world-famous architect Gary O’ Frank has been commissioned to design a new building, called the Data Center. Gary wants his top architectural protege to design a scale model of the Data Center using precision-cut sticks, but he wants to preclude the model from inadvertently containing any right angles. Gary fabricates a set of n sticks, labeled , where stick has length. Before giving the sticks to the protege, he shows them to you and asks you whether it is possible to create a right triangle using any three of the sticks.

**Give an efﬁcient algorithm** for determining whether there exist three sticks , , and such that the triangle formed from them — having sides of lengths , , and — is a right triangle (that is, ).

# Problem 5: Escape problem

An ***grid*** is an undirected graph consisting of rows and columns of vertices. We denote the vertex in the row and the column by . All vertices in a grid have exactly four neighbors, except for the boundary vertices, which are the points for which.

Given starting points in the grid, the ***escape problem*** is to determine whether or not there are ***vertex-disjoint paths*** from the starting points to any different points on the boundary. For example, the grid in Figure (a) has an escape, but the grid in Figure (b) does not.

Describe an efficient algorithm to solve the escape problem, and analyze its running time.

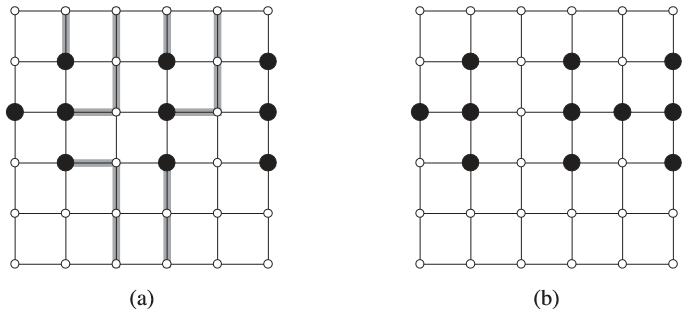


Figure: Grids for the escape problem. Starting points are black, and other grid vertices are white. **(a)** A grid with an escape, shown by shaded paths. **(b)** A grid with no escape

# Problem 6: Picking up sticks

Professor Charon has a set of sticks, which are piled up in some conﬁguration. Each stick is speciﬁed by its endpoints, and each endpoint is an ordered triple giving itscoordinates. No stick is vertical.

He wishes to pick up all the sticks, one at a time, subject to the condition that he may pick up a stick only if there is no other stick on top of it.

Describe an efﬁcient algorithm that determines whether it is possible to pick up all the sticks, and if so, provides a legal order in which to pick them up.