Remind: Every source code you provide should be compiled or be able to run at linux machine. Your code should not print any unrelated information.

Problem1.
In crypto future market, we usually trade weekly, biweekly and quarterly contract.
Here is the definition of three contracts from Huobi

Weekly contracts will be settled on imminent Friday not including today
Bi-weekly contracts will be settled on next Friday not including today
Quarterly contracts will be settled on the last Friday of March, June, September and December not including today.

The expiry date of weekly contracts should be before biweekly contract. The expiry date of biweekly contracts should be before quarterly contract.
Example:

```
Today = '20191218'
Weekly contract = '191220'
Bi-Weekly contract = '191227'
Quarterly contract = '200327'
```

```
Today = '20191201'
Weekly contract = '191206'
Bi-Weekly contract = '191213'
Quarterly contract = '191227'
```

Please write a python code to return the expiry date for the quarterly contract. Please fill the snippet in p1.py. *Hint: you could compute the expiry date of weekly/biweekly contract at first so that you could know when to switch quarterly contract.*

Problem2.
There are some processes which need to be executed. The running hours for each process could be represented by one positive integer number. If we have several servers, and the performance of each server is exactly the same. Please find the minimum hours that we should wait for all processes finished.
Given an array of *n(<=20)* positive integers, each integer represents the hour for each process should take. And *m(<=n)* is the number of servers we could use. Please fill in your snippet in p2.cpp which should be compiled with c++14.
Example:
({3, 4, 5, 6}, 3) -> ans = *7*, possibly distributed as *({3, 4}, {5}, {6})*
({3, 10, 3, 4}, 3) -> ans = *10*, possibly distributed as *({3, 3}, {4}, {10})*
({4, 5, 4, 5, 4, 3}, 2) -> ans = *13*, possibly distributed as *({5, 5, 3}, {4, 4, 4})*
({1000, 1, 2, 3}, 2) -> ans = *1000*, possibly distributed as *({1000}, {1, 2, 3})*

Problem3.
Given a binary file which consists of several size-fixed data in the little-endian order defined in the below. Please write a program with c++ to read ***output.dat*** file, and stdout in the following format.
We will redirect its stdout into another file so that we could compare your answer automatically. Please make the format correct.

```
std::cout << trade.timestamp << '\t' << trade.index << '\t' << (trade.buy ? "BUY" : "SELL") << std::endl;
```

Example:

```
1579620836000000000     1     SELL
1579620839000000000     2     BUY
1579620842000000000     3     BUY
1579620851000000000     4     BUY
1579620854000000000     5     BUY
1579620855000000000     6     BUY
```

```
struct Trade //size-fixed structure
{
    uint64_t timestamp;
    uint64_t index;
    bool buy;
    char padding[7];
};
```