

Федеральное государственное автономное образовательное
учреждение высшего образования
**"НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
"ВЫСШАЯ ШКОЛА ЭКОНОМИКИ"**

Факультет Компьютерных наук
Департамент программной инженерии

Пояснительная записка

к домашнему заданию по дисциплине
"Архитектура вычислительных систем"

Симоновича Иван Сергеевича

БПИ 193-2 19 Вариант

Москва 2020

Формулировка задания:

Вариант 19. У одной очень привлекательной студентки есть N поклонников. Традиционно в день св. Валентина очень привлекательная студентка проводит романтический вечер с одним из поклонников. Счастливый избранник заранее не известен. С утра очень привлекательная студентка получает N «валентинок» с различными вариантами романтического вечера. Выбрав наиболее заманчивое предложение, студентка извещает счастливчика о своем согласии, а остальных – об отказе. Требуется создать многопоточное приложение, моделирующее поведение студентки. При решении использовать парадигму «клиент-сервер» с активным ожиданием.

Решение задания:

При решении использовалась парадигма «клиент-сервер» с активным ожиданием. Согласно этой парадигме несколько клиентов и один сервер совместно с промежуточным программным обеспечением и средой взаимодействия образуют единую систему, обеспечивающую распределенные вычисления, анализ и представление данных. Если точнее, то клиентский поток запрашивает сервер и после ожидает ответа, в то время как сервер ждет запроса от клиента и после этого выполняет свою логику в зависимости от запроса и отправляет ответ на запрос клиенту. Иногда клиент может «исполнить» роль сервера, если сам будет получать запросы. Аналогично сервер будет выступать в роли клиента, если ему потребуется обращаться с запросами к другим программам.

Источники, из которых брал информацию:

<https://pro-prof.com/forums/topic/parallel-programming-paradigms> - про парадигму клиент-сервер

<https://bagdar.info/modele-klient-server-razdelenie-prilozenij-po-urovnyam-tipi-kl.html> - про парадигму клиент-сервер

http://www.cplusplus.com/reference/condition_variable/condition_variable/ - просмотр примеров

<https://habr.com/ru/post/182626/> - про работу с потоками

<https://science-pedagogy.ru/ru/article/view?id=1962> – про методы синхронизации

При запуске в программу вводится положительное число больше нуля. Верхнее ограничение - 10000, нужно для корректной работы программы и избегания чрезмерного использования памяти. В программе присутствует проверка и повтор ввода, при неверных значениях. Это положительное число n – количество клиентов сервера.

Для корректной реализации парадигмы клиент-сервер в программе были реализованы два класса Valentine, где хранятся данные об клиентах, и Server, где разработана логика сервера.

После успешного ввода создается массив из 8 фраз, указатель сервера и указатели потоков клиентов. После этого создается и запускается поток сервера, с методом serverMain класса Server.

```
thread serverThread([server] { server->serverMain(); });
```

В этом методе выведется информация, что начался новый день и сервер будет спать 3 секунды, принимая сообщения, эмулируя ожидание студентки.

В это же время в main в цикле определяются и запускаются для каждого клиента потоки содержащие метод ClientMain, куда передаются номер клиента, случайная фраза и сервер, на который он должен отправлять запросы.

```
for (int i = 0; i < n; i++)  
{  
    clientThreads[i] = new thread(clientMain, i + 1, phrases[rand() % 8],  
server);  
}
```

В этом методе клиент сообщает что начал писать письмо, после чего создается объект Valentine содержащий информацию о письме и его номере, после чего поток засыпает на случайное время, эмулируя реального клиента и его трату времени на написание этого письма.

```
this_thread::sleep_for(chrono::milliseconds(rand() % 1000));
```

Сам srand() задается в начале функции ClientMain, его сид зависит от номера каждого клиента и размера его введенного текста.

```
srand(roomNumber+text.length());
```

Если поток клиента уже поспал необходимое время, то отправляет свой запрос с ссылкой на письмо на сервер. А сам сервер при вызове запроса `enqueueRequest(Valentine* letter)` заносит это письмо в вектор `vector<Valentine*> letters` и тем самым увеличивает своё количество принятых писем.

```
server->enqueueRequest(&letter);
```

Дальше происходит **активное ожидание** потока, то есть пока сервер не определит победителя все запущенные потоки будут ожидать.

```
while (!server->isResponseReady())  
    this_thread::sleep_for(chrono::milliseconds(5));
```

Если сервер выбрал победителя (а он его всегда выбирает случайно из поступивших валентинок за время ожидания в 3 секунды: `winner = rand() % letters.size();`)

, то поток клиента продолжает выполняться, так как метод сервера `isResponseReady()` вернет true, и принимает ответ на свой запрос с сервера с помощью метода `server->getResponse(&letter)`, который возвращает является ли этот клиент победителем, и печатает информацию об этом в консоль. Сам сервер перед этим пишет количество полученных писем, а также письмо и номер комнаты победителя (номер самого клиента).

Клиенты могут отправлять запросы на сервер в случайном порядке, так как у каждого из них случайное время задержки (сна), после которого они уже и отправляют свой запрос (см. примеры тестов).

В методах `getResponse`, `enqueueRequest` и `serverMain`, вызываемых в потоках, используется мьютекс `mu` (методы `lock()`, `unlock()`), для синхронизации потоков, чтобы доступ к ресурсу получало по одному потоку.

В конце программы в цикле вызываются методы `join` для потоков клиента и потока сервера отдельно. Это нужно, чтобы потоки успели отработать до завершения программы без ошибок. В конце также удаляются сервер и потоки, это необходимо для очищения памяти.

Все подробные комментарии для каждого метода содержатся в коде программы.

Вся необходимая информация выводится в консоль. Если вывод происходит на сервере, то в начале предложения будет выведено `[SERVER]`, если вывод происходит в клиенте, то в начале выведется `[CLIENT i]`, где `i` – номер

клиента. После конца ожидания сервера (3 секунды) выводится информация о поступивших от клиентов валентинках и о выбранном студенткой победителе.

Под письмом в описании выше понимается экземпляр клиента валентинка `Valentine letter(text, roomNumber)`.

Тестирование программы:

Случаи некорректного ввода:

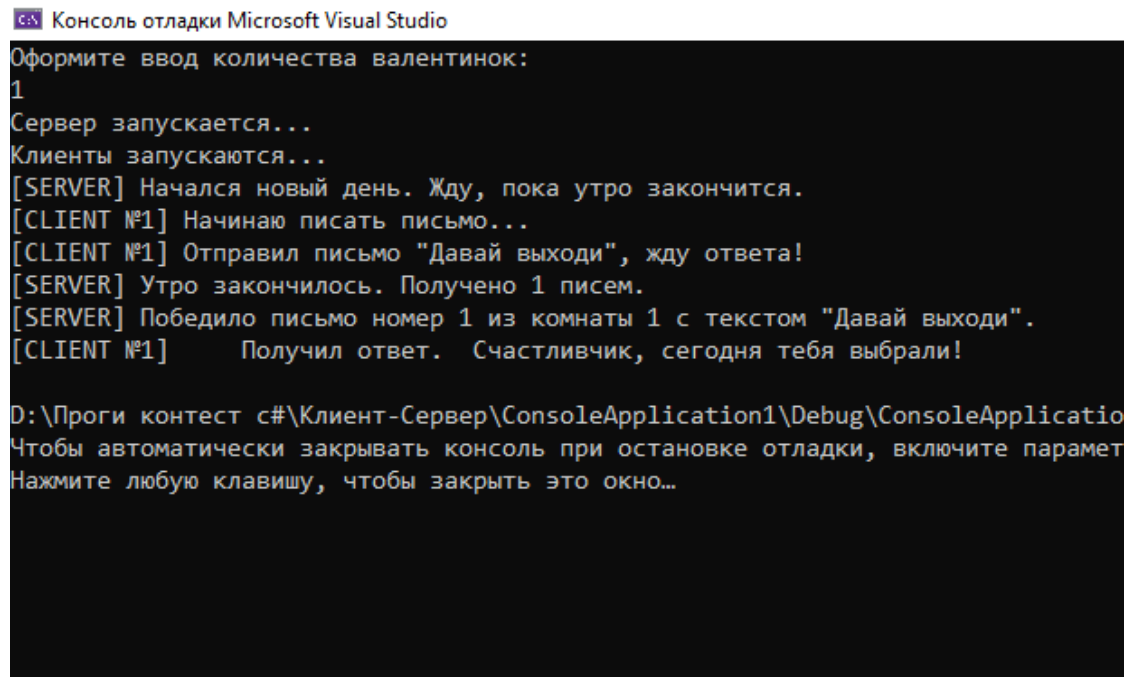
Несколько возможных случаев некорректного ввода разобраны на одном скрине. При вводе неподходящего числа программа запросит повтор ввода и продолжит выполняться после корректного ввода:

```
C:\Users\user\Desktop\HW многопоточность КОНЕЧНЫЙ\ConsoleApplication1\x64\Debug\ConsoleApplication1.exe
Оформите ввод количества валентинок:
10001
Введено отрицательное число, ноль или число >10000
Повторите ввод
-2
Введено отрицательное число, ноль или число >10000
Повторите ввод
0
Введено отрицательное число, ноль или число >10000
Повторите ввод
tdh
Ошибка ввода!
Повторите ввод
tdhhdh
Ошибка ввода!
Повторите ввод
2.2
Ошибка ввода!
Повторите ввод
-2.4
Ошибка ввода!
Повторите ввод
2
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №2] Отправил письмо "Моему любимому хейтеру", жду ответа!
[CLIENT №1] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[SERVER] Утро закончилось. Получено 2 писем.
[SERVER] Победило письмо номер 2 из комнаты 1 с текстом "Да, любил я сладко покодить, но больше всего я люблю тебя".
[CLIENT №2] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №1] Получил ответ. Счастливчик, сегодня тебя выбрали!
Для продолжения нажмите любую клавишу . . .
```

Примеры корректного ввода:

Ниже приведены примеры тестов для $n > 0$

Для $n=1$



```
Консоль отладки Microsoft Visual Studio
Оформите ввод количества валентинок:
1
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №1] Отправил письмо "Давай выходи", жду ответа!
[SERVER] Утро закончилось. Получено 1 писем.
[SERVER] Победило письмо номер 1 из комнаты 1 с текстом "Давай выходи".
[CLIENT №1] Получил ответ. Счастливчик, сегодня тебя выбрали!

D:\Проги контекст с#\Клиент-Сервер\ConsoleApplication1\Debug\ConsoleApplicatio
Чтобы автоматически закрывать консоль при остановке отладки, включите парамет
Нажмите любую клавишу, чтобы закрыть это окно...
```

Для $n=6$. Видно, что порядок отправки и выбора студента каждый раз новый. В первом случае выбрали 4-ое письмо, и 4-ым отправил 6 студент. А во втором случае студентка выбрала 3-е письмо от 5-ого клиента.

C:\Users\user\Desktop\HW многопоточность КОНЕЧНЫЙ\ConsoleApplication1\x64\Debug\ConsoleApplication1.exe

```
Оформите ввод количества валентинок:
6
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №3] Начинаю писать письмо...
[CLIENT №4] Начинаю писать письмо...
[CLIENT №6] Начинаю писать письмо...
[CLIENT №5] Начинаю писать письмо...
[CLIENT №3] Отправил письмо "Моему любимому хейтеру", жду ответа!
[CLIENT №1] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[CLIENT №2] Отправил письмо "Я никогда не объявлю тебе забастовку", жду ответа!
[CLIENT №6] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[CLIENT №4] Отправил письмо "Я никогда не объявлю тебе забастовку", жду ответа!
[CLIENT №5] Отправил письмо "Я никогда не объявлю тебе забастовку", жду ответа!
[SERVER] Утро закончилось. Получено 6 писем.
[SERVER] Победило письмо номер 4 из комнаты 6 с текстом "Сильнее пуджа я люблю только тебя".
[CLIENT №1] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №5] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №4] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №6] Получил ответ. Счастливчик, сегодня тебя выбрали!
[CLIENT №2] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №3] Получил ответ. Нет, студентка с тобой не идет!
Для продолжения нажмите любую клавишу . . .
```

```

Оформите ввод количества валентинок:
6
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №3] Начинаю писать письмо...
[CLIENT №4] Начинаю писать письмо...
[CLIENT №5] Начинаю писать письмо...
[CLIENT №6] Начинаю писать письмо...
[CLIENT №2] Отправил письмо "Любимка", жду ответа!
[CLIENT №1] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №5] Отправил письмо "Давай выходи", жду ответа!
[CLIENT №3] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №6] Отправил письмо "Давай выходи", жду ответа!
[CLIENT №4] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[SERVER] Утро закончилось. Получено 6 писем.
[SERVER] Победило письмо номер 3 из комнаты 5 с текстом "Давай выходи".
[CLIENT №4]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №2]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №3]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №1]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №6]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №5]    Получил ответ. Счастливчик, сегодня тебя выбрали!
Для продолжения нажмите любую клавишу . . .

```

Для n=8. Аналогично видно, что при разных запусках выводы значительно отличаются (порядок, победитель и его номер комнаты)

```

Оформите ввод количества валентинок:
8
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №3] Начинаю писать письмо...
[CLIENT №8] Начинаю писать письмо...
[CLIENT №5] Начинаю писать письмо...
[CLIENT №6] Начинаю писать письмо...
[CLIENT №7] Начинаю писать письмо...
[CLIENT №4] Начинаю писать письмо...
[CLIENT №7] Отправил письмо "Любимка", жду ответа!
[CLIENT №2] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №8] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №3] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[CLIENT №5] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[CLIENT №6] Отправил письмо "Дед мороз тебе свою любовь принес", жду ответа!
[CLIENT №1] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[CLIENT №4] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[SERVER] Утро закончилось. Получено 8 писем.
[SERVER] Победило письмо номер 2 из комнаты 2 с текстом "Я твоё чудовище".
[CLIENT №2]    Получил ответ. Счастливчик, сегодня тебя выбрали!
[CLIENT №5]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №1]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №4]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №6]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №8]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №3]    Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №7]    Получил ответ. Нет, студентка с тобой не идет!
Для продолжения нажмите любую клавишу . . .

```


C:\Users\user\Desktop\HW многопоточность КОНЕЧНЫЙ(ConsoleApplication1\Debug\ConsoleApplication1.exe

```
Оформите ввод количества валентинок:
8
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №7] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №3] Начинаю писать письмо...
[CLIENT №4] Начинаю писать письмо...
[CLIENT №5] Начинаю писать письмо...
[CLIENT №6] Начинаю писать письмо...
[CLIENT №1] Начинаю писать письмо...
[CLIENT №8] Начинаю писать письмо...
[CLIENT №4] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №2] Отправил письмо "Моему любимому хейтеру", жду ответа!
[CLIENT №1] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!
[CLIENT №5] Отправил письмо "Дед мороз тебе свою любовь принес", жду ответа!
[CLIENT №6] Отправил письмо "Дед мороз тебе свою любовь принес", жду ответа!
[CLIENT №8] Отправил письмо "Дед мороз тебе свою любовь принес", жду ответа!
[CLIENT №3] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[CLIENT №7] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[SERVER] Утро закончилось. Получено 8 писем.
[SERVER] Победило письмо номер 8 из комнаты 7 с текстом "Да, любил я сладко покодить, но больше всего я люблю тебя".
[CLIENT №6] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №3] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №5] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №8] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №2] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №7] Получил ответ. Счастливчик, сегодня тебя выбрали!
[CLIENT №4] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №1] Получил ответ. Нет, студентка с тобой не идет!
Для продолжения нажмите любую клавишу . . .
```

Для n=10. Аналогично

C:\Users\user\Desktop\HW многопоточность КОНЕЧНЫЙ(ConsoleApplication1\Debug\ConsoleApplication1.exe

```
Оформите ввод количества валентинок:
10
Сервер запускается...
Клиенты запускаются...
[SERVER] Начался новый день. Жду, пока утро закончится.
[CLIENT №1] Начинаю писать письмо...
[CLIENT №3] Начинаю писать письмо...
[CLIENT №2] Начинаю писать письмо...
[CLIENT №4] Начинаю писать письмо...
[CLIENT №5] Начинаю писать письмо...
[CLIENT №6] Начинаю писать письмо...
[CLIENT №7] Начинаю писать письмо...
[CLIENT №8] Начинаю писать письмо...
[CLIENT №9] Начинаю писать письмо...
[CLIENT №10] Начинаю писать письмо...
[CLIENT №1] Отправил письмо "Давай выходи", жду ответа!
[CLIENT №7] Отправил письмо "Любимка", жду ответа!
[CLIENT №8] Отправил письмо "Любимка", жду ответа!
[CLIENT №4] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №5] Отправил письмо "Я твоё чудовище", жду ответа!
[CLIENT №9] Отправил письмо "Давай выходи", жду ответа!
[CLIENT №2] Отправил письмо "Моему любимому хейтеру", жду ответа!
[CLIENT №6] Отправил письмо "Моему любимому хейтеру", жду ответа!
[CLIENT №3] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[CLIENT №10] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!
[SERVER] Утро закончилось. Получено 10 писем.
[SERVER] Победило письмо номер 10 из комнаты 10 с текстом "Да, любил я сладко покодить, но больше всего я люблю тебя".
[CLIENT №4] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №10] Получил ответ. Счастливчик, сегодня тебя выбрали!
[CLIENT №3] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №2] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №7] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №6] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №5] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №9] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №1] Получил ответ. Нет, студентка с тобой не идет!
[CLIENT №8] Получил ответ. Нет, студентка с тобой не идет!
Для продолжения нажмите любую клавишу . . .
```

C:\Users\user\Desktop\HW многопоточность КОНЕЧНЫЙ\ConsoleApplication1\x64\Debug\ConsoleApplication1.exe

Оформите ввод количества валентинок:

10

Сервер запускается...

Клиенты запускаются...

[SERVER] Начался новый день. Жду, пока утро закончится.

[CLIENT №1] Начинаю писать письмо...

[CLIENT №2] Начинаю писать письмо...

[CLIENT №3] Начинаю писать письмо...

[CLIENT №4] Начинаю писать письмо...

[CLIENT №5] Начинаю писать письмо...

[CLIENT №6] Начинаю писать письмо...

[CLIENT №7] Начинаю писать письмо...

[CLIENT №8] Начинаю писать письмо...

[CLIENT №9] Начинаю писать письмо...

[CLIENT №10] Начинаю писать письмо...

[CLIENT №2] Отправил письмо "Давай выходи", жду ответа!

[CLIENT №6] Отправил письмо "Любимка", жду ответа!

[CLIENT №8] Отправил письмо "Любимка", жду ответа!

[CLIENT №10] Отправил письмо "Любимка", жду ответа!

[CLIENT №9] Отправил письмо "Давай выходи", жду ответа!

[CLIENT №4] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!

[CLIENT №3] Отправил письмо "Я никогда не объявлю тебе забастовку", жду ответа!

[CLIENT №7] Отправил письмо "Сильнее пуджа я люблю только тебя", жду ответа!

[CLIENT №5] Отправил письмо "Я никогда не объявлю тебе забастовку", жду ответа!

[CLIENT №1] Отправил письмо "Да, любил я сладко покодить, но больше всего я люблю тебя", жду ответа!

[SERVER] Утро закончилось. Получено 10 писем.

[SERVER] Победило письмо номер 5 из комнаты 9 с текстом "Давай выходи".

[CLIENT №8] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №3] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №4] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №10] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №9] Получил ответ. Счастливчик, сегодня тебя выбрали!

[CLIENT №2] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №7] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №1] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №6] Получил ответ. Нет, студентка с тобой не идет!

[CLIENT №5] Получил ответ. Нет, студентка с тобой не идет!

Для продолжения нажмите любую клавишу . . . ■