

## 16. Очередь с защитой от ошибок

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Научитесь пользоваться стандартной структурой данных `queue` для целых чисел. Напишите программу, содержащую описание очереди и моделирующую работу очереди, реализовав все указанные здесь методы.

Программа считывает последовательность команд и в зависимости от команды выполняет ту или иную операцию. После выполнения каждой команды программа должна вывести одну строчку.

Возможные команды для программы:

`push n`

Добавить в очередь число `n` (значение `n` задается после команды). Программа должна вывести `ok`.

`pop`

Удалить из очереди первый элемент. Программа должна вывести его значение.

`front`

Программа должна вывести значение первого элемента, не удаляя его из очереди.

`size`

Программа должна вывести количество элементов в очереди.

`clear`

Программа должна очистить очередь и вывести `ok`.

`exit`

Программа должна вывести `bye` и завершить работу.

Перед исполнением операций `front` и `pop` программа должна проверять, содержится ли в очереди хотя бы один элемент. Если во входных данных встречается операция `front` или `pop`, и при этом очередь пуста, то программа должна вместо числового значения вывести строку `error`.

### Формат ввода

Вводятся команды управления очередью, по одной на строке

### Формат вывода

Требуется вывести протокол работы очереди, по одному сообщению на строке

#### Пример 1

Ввод <input type="text"/>	Вывод <input type="text"/>
<code>push 1</code>	<code>ok</code>
<code>front</code>	<code>1</code>
<code>exit</code>	<code>bye</code>

#### Пример 2

Ввод <input type="text"/>	Вывод <input type="text"/>
<code>size</code>	<code>0</code>
<code>push 1</code>	<code>ok</code>
<code>size</code>	<code>1</code>
<code>push 2</code>	<code>ok</code>
<code>size</code>	<code>2</code>
<code>push 3</code>	<code>ok</code>
<code>size</code>	<code>3</code>
<code>exit</code>	<code>bye</code>

## Пример 3

Ввод

Вывод

push 3	ok
push 14	ok
size	2
clear	ok
push 1	ok
front	1
push 2	ok
front	1
pop	1
size	1
pop	2
size	0
exit	bye

Язык

GNU GCC 12.2 C++20

Набрать здесь

Отправить файл

```
1 #include <iostream>
2 #include <queue>
3
4 int main() {
5     std::string s;
6     std::queue<int> q;
7     while(std::cin >> s) {
8         if (int tmp; s == "push") {
9             std::cin >> tmp;
10            q.push(tmp);
11            std::cout << "ok" << "\n";
12        } else if (s == "pop") {
13            if (q.empty()) {
14                std::cout << "error" << "\n";
15                continue;
16            }
17            std::cout << q.front() << "\n";
18            q.pop();
19        } else if (s == "front") {
20            if (q.empty()) {
21                std::cout << "error" << "\n";
22                continue;
23            }
24            std::cout << q.front() << "\n";
25        } else if (s == "size") {
26            std::cout << q.size() << "\n";
27        } else if (s == "clear") {
28            q = std::queue<int>();
29            std::cout << "ok" << "\n";
30        } else {
31            std::cout << "bye";
32            break;
33        }
34    }
35    return 0;
36 }
```

Отправить

Предыдущая

Следующая