



## Введение в программирование на языке с Практическая работа - L2 Проект 2023-2024 гг.



# Пчелы против шершней

## Оглавление

1	Цель проекта	1
2	Расположение и уровни игры	4
3	Благоустройство	9
4	Работа и визуализация	9
5	Последние советы	10

## 1 Цель проекта

Цель проекта - смоделировать взаимодействие между пчелами и шершнями (очень) простым способом. Проект основан на использовании списков цепочек и университетской графической библиотеки `libmclv`. Он может быть реализован в парах (или в исключительных случаях самостоятельно, в этом случае обратитесь к преподавателям) и разработан в последовательных шагах.

В игре есть две стороны - пчелы и шершни. Игра ведется на прямоугольной сетке. Существует пять видов пчел:

- Ульи, которые не оседают. Они могут производить маток, воинов, рабочих и эскадрильи воинов,
- королевы, которые перемещаются, нападают или строят улей,
- воины, которые наступают и атакуют,
- отряды воинов, которые выдвигаются и атакуют,
- рабочие, которые оседают или собирают пыльцу.

Пыльца - это ресурс пчел для их развития. Она и с п о л ь з у е т с я всеми ульями. Шершни делятся на три категории:

- Гнезда, которые не оседают. Они могут производить на свет королей и шершней,
- королевы, которые перемещаются, нападают или строят гнездо,
- шершни, которые перемещаются и могут нападать.

Ресурсы шершней - это пчелы, которых они создали. Мы будем называть этот ресурс пчелиным источником. Он является общим для всех осиных гнезд.

Мы называем единицу ульем, гнездом, королевой, рабочим, воином, эскадрилей или шершнем.

### 1.1 Настройка моделирования

Моделирование будет проводиться на прямоугольной сетке из 18 строк и 12 столбцов, и мы будем обозначать  $(i, j)$  клетку, расположенную в  $i$ -й строке и  $j$ -м столбце, так что  $(0, 0)$

находится слева сверху. Считается, что у каждой клетки есть 8 соседей. Края доски не могут пересекаться. Изначально имеется один улей, одна королева, один рабочий и один воин с t пчелами; с t шершнями - одно гнездо, одна королева и два шершня. Улей и гнездо расположены в противоположных углах, в точках (0, 0) и (18, 12). Мы

сможет инициализировать 10 единиц пыльцы для пчел и 10 пчелиных ресурсов для шершней.

В начале каждого раунда каждая единица помещается на клетку. Насекомые могут двигаться и сражаться, когда они находятся на одной клетке, после того как каждый из них переместился. Ход разворачивается следующим образом:

В конце раунда случайным образом определяется, кто из пчел или шершней будет действовать первым.

Сначала первый лагерь отдает приказы на изготовление ульев или гнезд, затем - на насекомых. Затем наступает очередь второго лагеря.

Только после этого битвы разрешаются (о разрешении битв см. ниже).

Затем ресурсы обеих сторон обновляются. Рабочие, которые пережили этот раунд без поражений, добавляют по 1 единице пыльцы, а пчелы, побежденные в бою, отдают шершням очки ресурсов пчел.

Поворот окончен.

## 1.2 Ульи и гнезда

Насекомые производятся в ульях пчел и гнездах шершней. Стоимость производства пчел измеряется в единицах пыльцы (собираемой рабочими), а шершней - в количестве пчел (получаемых в бою). Королева пчел или шершней может создать улей или гнездо за 1 ход за 10 единиц пыльцы или 10 пчел соответственно.

Ульи и гнезда могут производить насекомых. Выбрав производство улья или гнезда, его нельзя изменить до тех пор, пока оно не будет завершено. Здесь указаны время в количестве ходов и стоимость производства для каждой стороны. Эти значения должны быть легко изменяемы.

Пчелы 	Ко 	Время
Queen	7	8
Книга	3	2
Герри ре	5	4
Эскадрилья	6	6

Шершни 	Ко 	Время
Queen	8	8
Шершень	3	5

После завершения работы улей или гнездо ожидает нового производственного заказа.

У пчел и шершней есть система привязанности к улью и гнезду. Все члены улья или гнезда привязаны к нему. Более того, когда королева пчел или шершней строит улей или гнездо, она прикрепляется к нему сама.

## 1.3 Выбор действий

Чтобы легко определить единицу, ожидающую заказа, она помечается д р у г и м цветом.

### 1.3.1 Квинс

Королевы пчел и шершней имеют возможность

1. Переместитесь на соседнюю площадь
2. Создайте улей или гнездо (один на квадрат, 1 ход)
3. Пропустите поворот

Королева может основать только один новый улей или гнездо.

### 1.3.2 Военные корабли, эскадрильи и шершни

Эти три класса насекомых способны

1. Переместитесь на соседнюю площадь

2. Пропустите поворот

### 1.3.3 Работы

У работника есть возможность

1. Переместиться на соседнюю площадь
2. Сбор пыльцы (4 раунда)

Пыльца может быть собрана на любой клетке сетки, не занятой отрядом шершней. Чтобы собрать урожай, рабочий остается неподвижным в течение 4 ходов, после чего погибает (отряд уничтожается, не возвращая шершням ни одного пчелиного ресурса). Во время каждого раунда сбора урожая и обновления ресурсов собирается 1 единица пыльцы, которая добавляется к количеству ресурсов пчел.

### 1.4 Бой

Если в конце хода два противоборствующих отряда оказываются на одной клетке, происходит бой.

Бой - это...

Конец поединка определяется случайным броском 60-гранной игральной кости (например, функция `int MLV_integer_random (int begin, int end)`). К их силе применяется коэффициент. По ссылке

Например, если сражаются пчела-воин и шершень, мы сравниваем `drawDe() x FGUERRIERE` и `drawDe() x FFRELON`, и побеждает тот, чей показатель выше.

В конце боя действуют следующие правила:

Когда насекомое проигрывает бой, оно погибает.

Когда королева шершней или шершень побеждают улей, улей превращается в гнездо шершней, а все связанные пчелы уничтожаются. Шершень, победивший улей, оказывается в этом новом гнезде.

Все пчелы, убитые во время боя или захвата улья, приносят шершням очки пчелиных ресурсов. Количество пчелиных ресурсов, полученных в зависимости от того, кто был побежден или уничтожен, соответствует количеству единиц пыльцы, необходимых для их первоначального производства, т.е.

Пчелиная форель	Отчет о пчелиных ресурсах
Queen	7
Книга	3
Герри ре	5
Эскадрилья	6

Если несколько отрядов из одного лагеря находятся на одной площади, процесс повторяется до тех пор, пока один из лагерей не покинет площадь. Отряды, занимающие одну и ту же площадь, получают приказ вмешаться первыми:

эскадрильи, воины, королевы и пчелы;

шершни, затем королевы с шершнями.

Если улей или гнездо сохранились, они подвергаются нападению только в том случае, если рядом нет насекомого из их лагеря, которое могло бы их уничтожить.

### 1.5 Конец игры

Игра закончится, когда все ульи или осиные гнезда будут уничтожены. Вы также можете выйти из игры.

### 1.6 Сохранение и загрузка

В начале хода, после случайной жеребьевки, определяющей, кто начинает ход, сторона, играющая первой, может выбрать сохранение игры в файл. Формат сохранения описан в

разделе 2.4 ниже. Он фотографирует игру на момент сохранения, записывая, какая сторона будет действовать, ее ресурсы и юниты, а также ресурсы и юниты противника.

Также должна быть возможность выбрать сохранение сохраненной части в файл. В этом случае пользователь вводит имя файла резервной копии.

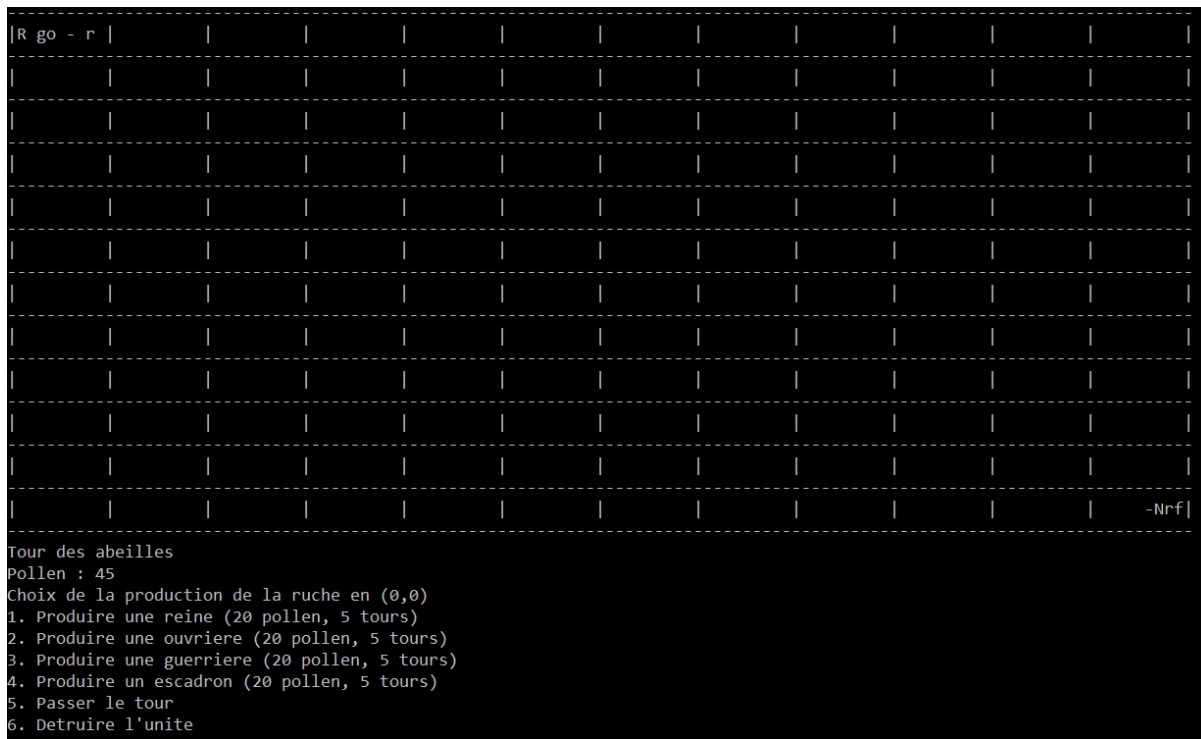
## 2 Расположение и уровни игры

Проект `nalis` использует графический интерфейс, получаемый через `libMLV`. Менее продвинутый уровень проекта может обойтись терминальным интерфейсом.

Игра ведется по очереди. В конце своего хода игрок должен указать номер своего хода, чтобы вернуть руку сопернику.

## 2.1 Интерфейс терминала

Чтобы быстрее настроить первые шаги моделирования, вы можете начать с сохранения интерфейса через libMLV и использования терминала. Например (с более мелкой сеткой для улучшения видимости) :



Единицы измерения, которые можно найти в коробке, обозначаются следующим кодом: Rroge-Nrf. Первые 5 символов соответствуют пчелам: R - если присутствует улей, r - если присутствует королева пчел, o, g, e - если присутствует рабочий, воин или эскадрилья. Последние 4 буквы обозначают шершней: n - если присутствует гнездо, r - если присутствует королева шершней и f - если присутствует шершень. Если шершня нет, не пишите букву и оставьте пробел. В частности, в одной коробке не может быть улья и гнезда. Например, R go - r означает, что на квадрате есть улей, воин, рабочий и королева шершней.

Чтобы разместить насекомых, игрокам предлагается указать направление соседнего квадрата, на который они хотят переместиться: N (север), NE (северо-восток), E (восток), SE (юго-восток), S (юг), SW (юго-запад), W (запад), NW (северо-запад).

## 2.2 Графический интерфейс

Графический интерфейс должен соответствовать следующим требованиям:

одна часть представляет игровую сетку с видимым рисунком сетки

Каждый квадрат имеет размер 60 на 60 пикселей и разделен на 7 зон следующим образом

О	Королева пчел		Королева шершней	
	Гуерриер	Улей или гнездо	Шершень	
	Гуерриер	Эскадриль		

Улей/гнездо и насекомые могут быть представлены квадратами, окрашенными в разные цвета (или даже с помощью изображений из руководства MLV\_image.h);

Вторая часть образует строку меню, содержащую кнопки для управления различными приказами, отправляемыми рабочим, воинам, эскадрильям, шершням, королевам, ульям и гнездам; она также содержит раздел для отображения координат обрабатываемой единицы и действий, которые она выполняет в данный момент;

Последняя зона содержит кнопку сохранения, кнопку загрузки, кнопку выхода из игры, кнопку окончания хода, а также номер хода и клан текущего игрока.

Чтобы поместить насекомое на соседнюю клетку, щелкните по ней во время хода насекомого.

## 2.3 Структуры данных и представления

Ульи, гнезда и насекомые представлены одной и той же структурой Unite. В зависимости от используемых шагов реализации некоторые поля могут не использоваться. Используются следующие символические константы и структуры:

```
// Размеры сетки в количестве ячеек (начало в левом верхнем углу): #define
COLUMNS 12
#define LINES 18
```

```
// Обе стороны:
#define BEE 'A' #define
WILDLIFE 'F'
```

```
// Типы устройств :
#define REINE 'r'
#define OUVRIERE 'o'
#define ESCADRON 'e'
#define GUERRIERE 'g'
#define FRELON 'f'
#define RUCHE 'R'
#define NID 'N'
// Для сбора пыльцы #define
RECOLTE 'p'
```

```
// Время производства: #define TREINEA 8
#define TREINEF 8
#define TOUVRIERE 2
#define TGUERRIERE 4
#define TESCADRON 6
#define TFRELON 5
#define TRECOLTE 4
```



```

// Производственные затраты :
#define CREINEA 7
#define CREINEF 8
#define COUVRIERE 3
#define CGUERRIERE 5
#define CESCADRON 6
#define CFRELON 3
#define CRUCHE 10
#define CNID 10

// Сила единиц #define
FREINE 6
#define FOUVRIERE 1
#define FGUERRIERE 5
#define FESCADRON 12
#define FFRELON 8

// Структура Unite :
typedef struct unite {
    лагерь; // BEE или FRELON
    тип char; // HATCH, NEST, QUEEN, WORKER, WARRIOR, ESCADRON или FRELON
    int force; // сила блока
    int posx, posy; // текущая позиция на сетке int
    destx, desty; // пункт назначения
    (отрицательный, если он неподвижен)
    char production; // производство улья или гнезда и RECOLTE для сбора пыльцы int time;
    // общее количество оборотов для данного производства
    int remaining turns; // оставшиеся обороты для данного производства
    struct unit *usuiv, *uprec; // список единиц, связанных с ульем или гнездом
    struct unit *colsuiv, *colprec; // список других ульев или гнезд (колоний) в том же лагере struct
    unit *vsuiv, *vprec; // список других юнитов на той же площади
} Объединение, *Улиста;

// Структура дела :
typedef struct {
    Единица * колония; // Если на площади UList есть улей
    или гнездо; // другие обитатели площади
} Case;

// Структура сетки :
typedef struct {
    Кейс плато[X][Y]; пчела-
    улиста, шершень;
    int tour; // Номер тура
    int resourcesBee, resourcesFrelon;
} Grid;

```

Помните, что насекомые, производимые ульем или гнездом, прикреплены к нему. Кроме того, королева пчел или шершней, которая основывает улей или гнездо, является его частью.

## 2.4 Формат резервного копирования

Файлы резервных копий **отформатированы** следующим образом. Первая строка содержит символ `char` и два целых числа, разделенных пробелом: буква ('A' или 'F'), указывающая сторону, с которой предстоит играть, и ее ресурсы (в пыльцевых единицах для пчел и в пчелах для шершней), затем ресурсы другой стороны.

Каждая из следующих строк содержит характеристики одного блока (за исключением, конечно, значений указателей). Таким образом, строка состоит из 2 `char`, за которыми следуют 2 `int`, затем 1 `char` и в n 1 `int`, разделенных одним пробелом.

Такое форматирование должно позволить изменять тестовые файлы.

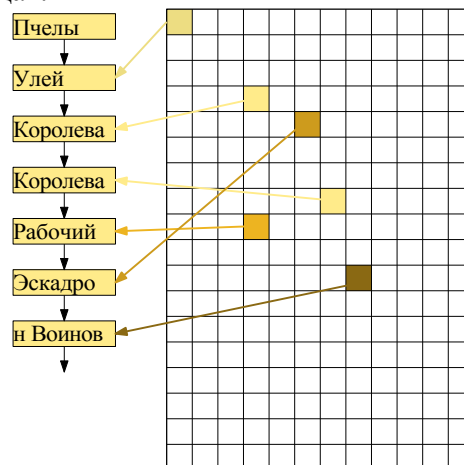
## 2.5 Этапы развития

Студентам предлагается разрабатывать проект поэтапно, по мере возрастания сложности, что соответствует увеличению максимальных оценок.

### 2.5.1 Стадия "Пчелиная колония"

Первая задача - управлять размещением пчел и производством одного улья. Вы должны уметь выбирать производство улья, размещать или уничтожать насекомых и собирать деньги для рабочих. Каждый ход вы должны находить упорядоченный список (королевы, эскадрильи, воины, затем рабочие для пчел) всех присутствующих юнитов.

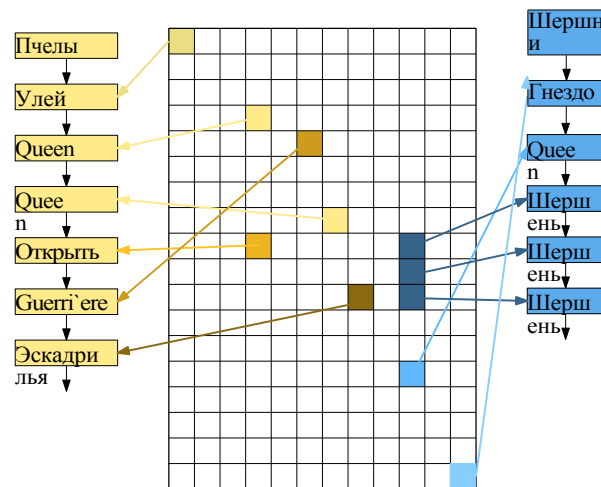
Система оценок следующая.



В этот м о м е н т две единицы не могут быть размещены на одной клетке. Насекомое, произведенное ульем, помещается на соседнюю свободную клетку. Квадрат указывает на единицу, находящуюся на этой клетке.

### 2.5.2 Сцена "Гнездо шершня"

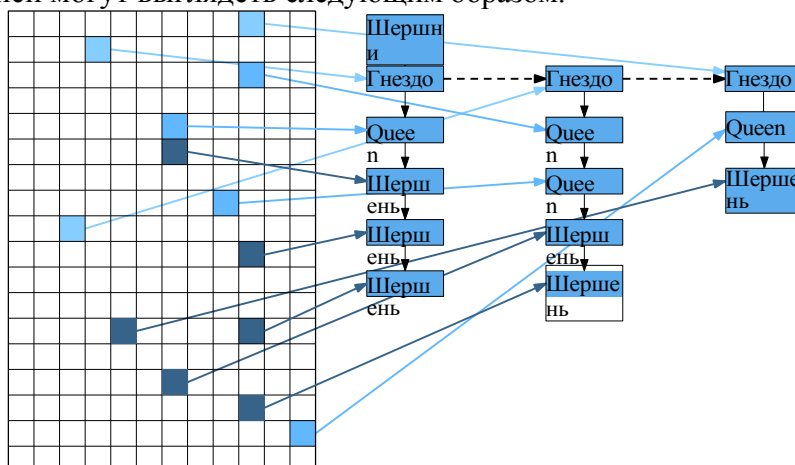
Я работаю так же, как и в шаге "Пчелиная колония", но добавляю осиное гнездо с его королевами и шершнями. Схема подсчета очков выглядит следующим образом.



### 2.5.3 Стадия "Колонии и многочисленные гнезда"

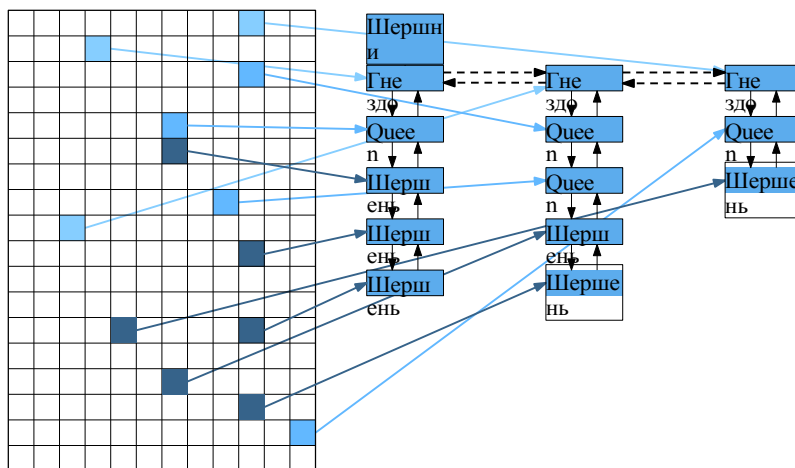
Мы добавляем возможность для каждой стороны создавать новые колонии, что означает, что королевы пчел могут создавать ульи, а королевы шершней - гнезда. Новые ульи или гнезда добавляются в связанный список ульев или гнезд с помощью указателей \*colsuiv. Для этих первых трех шагов вы можете использовать только списки, которые просто связаны указателями \*usuiv и \*colsuiv.

Пятна шершней могут выглядеть следующим образом.



### 2.5.4 Боевая сцена

В дополнение к крау "Колонии и несколько гнезд" мы добавили управление боем между двумя сторонами, причем на одной клетке всегда будет максимум по одному юниту от каждой стороны. На этом уровне игрок будет использовать двойные цепи для соединения ульев и гнезд (\*usuiv, \*uprec), а также для перемещения из одного улья или гнезда в другое (\*colsuiv, \*colprec).



### 2.5.5 Шаг "Резервное копирование"

Мы также добавили управление сохранением и загрузкой игры.

### 2.5.6 Графический интерфейс" этап

Использование графического интерфейса является обязательным для этой ленты.

### 2.5.7 Заключительный этап

Теперь в одной ячейке можно разместить несколько единиц. Для этого используйте двухцепочечный список юнитов в одной ячейке (\*vsuiv, \*vprec). Юниты в цепочке могут быть упорядочены в соответствии с порядком их участия в бою (см. раздел 1.4).

## 3 Am liorations

Улучшения возможны, но они будут учтены только в том случае, если были предприняты все предыдущие шаги. Например

Добавьте систему хит-пойнтов и очков атаки для каждого юнита.

Разрешите рабочим перемещаться на 2 клетки, если они не примыкают к отряду в лагере шершней

Соотнесите силу улья с количеством воинов и эскадрилий в нем. Соотнесите силу гнезда с количеством шершней в нем.

Соотнесите время производства в улье с количеством рабочих.

## 4 Работа и

Проект будет выполняться в парах. Отчет будет включать в себя :

Программы на языке C, которые можно компилировать в университете;  
отчет компании Crivant:

как работает игра;

во-вторых, используемые методы и выбор макета.

Цель отчета - дать возможность любому пользователю использовать программу, а более опытному программисту - с легкостью заставить ваш код развиваться.

Защита состоится в январе 2024 года. В рамках защиты ваша программа будет скомпилирована, а код продемонстрирован на университетской машине. После этого каждому члену пары будет задан ряд вопросов о вашей работе.

## 5 Последние советы

Программы на языке C должны быть понятными и хорошо структурированными. Не начинайте разработку новой ленты, пока предыдущая не будет полностью функциональной.

Сохраняйте функциональные версии шаг за шагом, чтобы можно было начать заново, когда новые функции окажутся слишком сложными для реализации. Каждую версию можно протестировать в день презентации.