Игра «Змейка». Отображение движения змейки (на экране) в одном направлении.

Темы: Инструкции цикла (while, for), перечисления, директивы препроцессора, одномерные массивы.

В этом задании не потребуется обрабатывать пользовательский ввод. На этот раз нам надо будет научиться изображать движущуюся змейку. Для решения этой задачи можно использовать разные приемы, один из которых мы применим в этой работе, а другой рассмотрим в следующей.

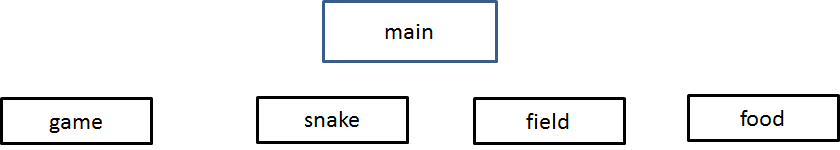
**Замечание**

Текущая версия программы будет упрощенной.

* Поле, по которому двигается змейка будет состоять только из одной строки
* Обработка нажатия клавиш производиться не будет
* Весь рабочий код пока будет располагаться в функции main.

# Декомпозиция задачи

В результате декомпозиции задачи можно выделить следующие “модули”:



* поле игры
* змейка
* еда
* игра

+

* функция main
* файл констант

**Применение модульности**

Каждый “модуль” отвечает за свой «объект», причем каждый “модуль” будет состоять из заголовочного файла + файла реализации. Не все перечисленные модули мы будем использовать в текущей работе.

**Задание 1**

Добавьте в проект файлы:

1. snake.cpp и snake.h, которые будут содержать информацию, относящуюся к понятию "Змейка"
2. game.cpp и game.h, которые будут содержать информацию, относящуюся к понятию "Игра".
3. Заголовочный файл const.h, который будет содержать общие константы.
4. Файл main.cpp будет содержать все управляющие инструкции

Замечание

Модули «Еда» и «Поле игры» нам пока не потребуются.

## Обмен данными между подпрограммами.

Для передачи информации между функциями будем использовать **глобальные** переменные (пока!).

## Глобальные данные

|  |  |  |  |
| --- | --- | --- | --- |
| Описание | переменная | Модуль | Примечания |
| Текущий размер змейки | snake\_size | snake | >0  Включает голову+размер хвоста  (если 1- то только голова) |
| Таймаут задержки между шагами игры | timeout | game | >100 (ms) |
|  |  |  |  |
| Символ для головы змейки | head\_symbol | const.h | const |
| Символ для хвоста змейки | tail\_symbol | const.h | const |
| Длина игрового поля | columns | const.h | const, >0 |

Имеет смысл все константы поместить в отдельный модуль. Это будет удобно для настройки отображения игры.

**Задание 2**

1. Определите переменные и константы:

* таймаут задержки между шагами игры
* символ головы змейки
* символ хвоста змейки
* текущий размер змейки
* длину игрового поля

Подумайте, где можно определять названные переменные и как сделать их доступными из функции main

**Задание 3**

Будем изображать движущуюся змейку (состоящую только из головы). Задайте значение длины змейки, равное 1. Пока змейка будет двигаться в цикле пока позволяет длина игрового поля.

Идея анимации состоит в том, что:

* текущее изображение объекта стирается
* изменяется позиция объекта
* объект изображается в новой позиции

Для упрощения будем использовать полную очистку консоли. Стирание экрана можно выполнить с помощью вызова функции **system (“cls”).** Эта функция объявлена в файле <windows.h>.

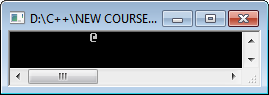
Для того, чтобы видеть изменение положения змейки через определенные интервалы времени надо вызвать функцию **Sleep ()**, а в качестве параметра задать значение таймаута в миллисекундах.

Например, **Sleep (1000);** будет задерживать выполнение программы на 1 секунду.

Реализуйте отображение передвижения змейки до тех пор, пока она не достигнет границы игрового поля.

Змейка начинает двигаться из левого верхнего угла экрана, т.е. ее начальная координата =0

Подсказка: для того, чтобы изобразить змейку в позиции **Х,** надо сначала вывести X пробелов.



**Задание 4**

В этом задании будем применять те же идеи, что им задании 2, но кроме головы надо будет изобразить хвост змейки. Будем изображать движущуюся змейку, состоящую из головы и хвоста, поэтому необходимо задать значение длины змейки, отличное от 1.

