Движение змейки в одном направлении + еда + функции без параметров (подсказки).

## Модули программы

|  |  |
| --- | --- |
| Название | Описание |
| snake | данные и функции, относящиеся к змейке (только) |
| game | данные и функции, относящиеся к игре (только) |
| const | общие данные (нужны в нескольких модулях) |
| field | данные и функции, относящиеся к полю игры (только) |
| food | данные и функции, относящиеся к еде (только) |
| main | Управляющий модуль |

## 

## Данные модуля const

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | тип |
| Длина игрового поля | columns |  |
| Максимальный размер массива змейки | L |  |
| Символ для хвоста змейки | tail\_symbol |  |
| Символ для головы змейки | head\_symbol |  |
| Символ еды | food\_symbol |  |
| Символ поля | field\_symbol |  |
| Символ границы | border\_symbol |  |

## Данные (переменные) модуля main

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | Тип |
| Текущий размер змейки |  |  |
| Массив координат змейки по горизонтали (X) |  |  |
| Массив игрового поля |  |  |
| Признак, что еда установлена |  |  |
| Координата еды по горизонтали (X) |  |  |
| Таймаут задержки между шагами игры |  |  |
| Признак продолжения игры |  |  |

## Функции модуля snake

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | Надо передать |
| Инициализация змейки | ??? init\_snake(???) | Начальную позицию змейки, массив змейки ,[возможно что-то еще] |
| Передвижение змейки | ??? move\_snake(???) | Новую позицию змейки, массив змейки,[возможно что-то еще] |

### Алгоритм функции init\_snake ( ) –без изменения

Устанавливается начальный размер змейки. Массив змейки заполняется значениями

### Алгоритм функции move\_snake ( )- без изменения

* Передвигаются элементы хвоста
* Передвигается голова

## Функции модуля field

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | Примечания |
| Инициализация поля. Заполняется массив поля, в том числе границы | ??? init\_ field (???) | Массив поля, [возможно что-то еще] |
| Печать поля | ??? print\_field(???) | Массив поля, [возможно что-то еще] |
| Очистка поля | ???clear\_field(???) |  |

### Алгоритм функции init\_field( )-без изменения

Массив игрового поля заполняется символом-очистителем (например, пробелом), а граница поля символом границы (например, #).

### Алгоритм функции print\_field( ) -без изменения

Массив игрового поля выводится на экран.

### Алгоритм функции clear\_field( ) -без изменения

Массив игрового поля заполняется символом-очистителем (например, пробелом).

## Функции модуля food

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | Примечания |
| Генерация еды | ??? generate\_food(???) | Эта функция не является необходимой, генерацию еды можно выполнить в рамках функции установки еды (в модуле game) |

### Алгоритм функции generate\_food( ) -без изменения

Генерируется случайное значение, которое используется в качестве координаты еды в поле игры. Важно, чтобы позиция еды попадала в диапазон индексов игрового поля, но в то же время не попадала на границу поля.

## Функции модуля game

|  |  |  |
| --- | --- | --- |
| Описание | Идентификатор | Примечания |
| Инициализация игры | ??? init\_game(???) | таймаут, игра, [возможно что-то еще] |
| Установка змейки | ??? set\_snake(???) | Массив поля игры, массив змейки [возможно что-то еще] |
| Очистка поля змейки | ??? clear\_snake(???) | Массив поля игры, массив змейки, [возможно что-то еще] |
| Проверка еды (что змейка съела еду) | ??? check\_eating(???) | Массив змейки, параметры еды, [возможно что-то еще] |
| Проверка змейки (что она заползла на границу или укусила себя). | ??? check\_snake(???) | Массив поля игры, массив змейки, [возможно что-то еще] |
| Установка еды | ???set\_food(???) | Массив поля игры, параметры еды, [возможно что-то еще] |
| Проверка конца игры | ??? check­\_game(???) | Здесь можно проверить, что змейке некуда ползти, а также, что змейка укусила себя |

### Алгоритм функции init\_game( ) -без изменения

Устанавливаются начальные значения параметров игры.

### Алгоритм функции set\_snake( ) -без изменения

Змейка устанавливается в игровом поле, т.е.

* символ головы змейки копируется в соответствующую позицию игрового поля
* символы хвоста также копируются в игровое поле.

### Алгоритм функции clear\_snake( ) -без изменения

Очищается позиция змейки в массиве, т.е. в соответствующей позиции поля устанавливается пробел.

### Алгоритм функции set\_food ( ) -без изменения

Еда устанавливается в игровом поле. Можно вызывать функцию generate\_food() и проверять, что в сгенерированной позиции ничего нет.

### Алгоритм функции check\_eating ( )

Выполняется проверка того, что змейка съела еду. Если координаты еды и координаты головы змейки совпадают, то

* устанавливается в false признак того, что еда есть на поле.

Замечание: признак того, что еда отсутствует на поле, можно вернуть либо через возвращаемое значение, либо через параметр

### Алгоритм функции check\_snake ( )

Выполняется проверка того, что змейка натолкнулась на препятствие.

* Если голова змейки упирается в стенку, то движение змейки идет по кольцу, т.е. она появляется с другого (противоположного) края поля.
* Если голова змейки натолкнулась на свой хвост, то игра завершается.

Замечание: признак того, что змейка укусила сама себя, можно вернуть либо через возвращаемое значение, либо через параметр

### Алгоритм функции check\_game ( )

Выполняется проверка того, что на поле есть свободное место для передвижения. Если места нет, то игра должна завершиться.

Замечание: признак того, что игра должна завершиться, можно вернуть либо через возвращаемое значение, либо через параметр.