

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский университет «Высшая школа экономики»  
Факультет компьютерных наук  
Образовательная программа Прикладная математика и информатика  
бакалавриат  
**01.03.02 Прикладная математика и информатика**

**О Т Ч Е Т**  
**по учебной практике**

Выполнил студент гр. БПМИ176  
Биршерт Алексей Дмитриевич  
(ФИО)

\_\_\_\_\_  
(подпись)

**Проверили:**

доцент ДБДИП Авдеев Роман Сергеевич

\_\_\_\_\_  
(оценка)

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
(дата)

**2017/2018 уч.г.**

## Оглавление

Введение .....	3
Основная часть.....	4
Заключение.....	7
Список литературы.....	8
Приложение.....	9

## Введение

В своей работе я изучаю математическую модель известную как муравей Лэнгтона. Муравей Лэнгтона – это двухмерная машина Тьюринга, изобретенная в 1986 году Кристофером Лэнгтоном. Муравей работает на бесконечном поле, каждая из клеточек которого может быть покрашена в какое-то количество цветов, когда муравей оказывается на какой-то клетке, он поворачивает налево или направо на 90 градусов в зависимости от цвета, перекрашивает текущую клетку по циклу ( $0 \rightarrow 1, 1 \rightarrow 2, 2 \rightarrow 3$  и так далее, цвета меняются как вычеты по модулю количества цветов) и отправляется дальше. Я работал с шести цветным обобщением муравья, работающим по правилу LLLRRR, то есть на первых трех цветах муравей поворачивает налево, на последних трех направо, цвета меняются от 0 до 5 по циклу. После какого-то достаточно непродолжительного движения по полю, покрашенному в 0 цвет целиком, муравей начинает строить так называемый «хайвей» - в последовательности поворотов муравья, состоящей из поворотов влево и вправо, появляется некий цикл длиной в 38 шагов. Следуя этому циклу, муравей строит бесконечную диагональную дорогу. Однако до сих пор остается неразрешенным вопрос, а на всякой ли конечной раскраске поля, то есть, когда только конечное число клеток поля перед началом движения муравья покрашено в цвет, отличающийся от 0, муравей начнет строить «хайвей».

В этой работе я стремлюсь разрешить две задачи:

1. Научиться определять, вошел ли муравей в «хайвей» на текущем шагу и если да, то с какого момента в цикле.
2. Для большого количества произвольных раскрасок поля изучить наиболее частые последовательности квазипериодического движения муравья.

## Основная часть

Для того, чтобы изучать муравья Лэнгтона, я решил написать компьютерную симуляцию муравья на языке C++17. Исходный код (представлен в файлах №1 и №2 приложения) работает по следующему принципу – надо было реализовать бесконечное поле и самого муравья. Бесконечное поле я представил в виде словаря с ключами в виде пары координат {x, y}, на значениях объект класса Node, который имеет три дочерних поля со значениями его координат и цвета, а также дочернюю функцию для смены цвета клеточки (void ch\_color()). Сам муравей представлен в виде объекта класса Ant, который имеет три дочерних поля – текущие координаты муравья и его направление (0 это вверх, 1 вправо, 2 вниз и 3 влево). Так же муравей имеет две дочерние функции - (void ch\_direction(Node& n)) для смены направления на клетке, (void new\_xy()) для смены координат муравья в соответствии с его направлением.

В самом начале я решаю задачу поиска цикла муравья, чтобы определить его длину и саму последовательность поворотов. Для этого я запускаю муравья на непокрашенном поле на 1000 шагов (визуализацию поля после 800 ходов муравья можно посмотреть в файле №3 приложения) и записываю последовательность поворотов в массив пар {"left/right", color}. Потом я рисую все поле, визуализируя лишь клетки, покрашенные в ненулевой цвет. Муравей рисует хайвей с достаточно маленького шага, я беру вторую половину последовательности поворотов и ищу в ней цикл минимальной длины. Функция, которая ищет цикл (std::vector<std::pair<std::string, int>> find\_cycle(std::vector<std::pair<std::string, int>>& path)) возвращает массив пар {"left/right", color} с найденным циклом. Очевидно, что если цикл повторяется в последовательности, то если я возьму два элемента на каком-то расстоянии друг от друга (начиная с 10), и они будут равны, то можно сверить два кусочка последовательности между ними и следующим за вторым, и если они будут равны, то мы нашли цикл, равный по длине расстоянию между этими элементами. Таким образом, я получил цикл длиной 38. Однако я задался вопросом найти первый элемент цикла для запуска с чистого поля. Для этого я спустился по последовательности поворотов муравья назад до момента, когда кусочек длиной 38 не совпал с моим циклом, потом нашел элемент, на котором эти массивы не совпали. Таким образом, элемент, следующий за первым неравным, будет первым в моем цикле поворотов, который уже можно назвать «хайвеем» с первого шага. Итоговый цикл имеет следующий вид, как показано в файле №4 приложения.

Теперь я задался задачей описать все клетки, которые муравей посетит во время своего «хайвея», чтобы получить возможность определить одно из условий, необходимых для определения вхождения в «хайвей». Я запустил муравья с 0 поля, с момента вхождения в «хайвей», записанного в первый раз, я запустил цикл, который будет записывать для каждого из 38 шагов все клетки, которые муравей посетит в будущем в относительных координатах (относительно точки старта шага) до следующего такого же шага. Таким образом я получил массив из 4 массивов (для 4 начальных направлений), каждый из которых содержит массив из 38 массивов клеток. Очевидно, что если муравей посетил какую-то клетку дважды, то логично будет проверять цвет только для первого посещения. Тогда я записал точно такой же массив, только содержащий все клетки в момент первого

посещения. Теперь я имею массив, который по текущему положению муравья и его направлению может сказать, какие клетки вокруг него должны быть на каждом из 38 шагов хайвея. Еще немаловажным будет уметь определять направление самого «хайвея» в зависимости от направления муравья в текущем шаге «хайвея». Всего «хайвей» может иметь четыре направления – влево вверх, влево вниз, вправо вверх и вправо вниз. Я знаю, как соотносится направление «хайвея» с начальным направлением муравья в нулевой момент времени, соответственно могу записать для каждого направления «хайвея» направление муравья на каждом из 38 шагов.

Теперь надо реализовать второе условие – в дальнейшей полосе хайвея не должно быть клеток, покрашенных не в 0 цвет. Для этого надо просто перебрать все клетки, которые покрашены не в 0 цвет и определить, попали ли они в полосу «хайвея». Если да, то «хайвей» не будет строиться бесконечно, значит определяем не вхождение.

Теперь я имею алгоритм (`inline int if_highway(Node curr, std::vector<std::vector<std::vector<Node>>>& future, int direction, int pos)`), который определяет, вошел муравей в «хайвей» или нет по двум вышеописанным условиям – если вошел, возвращает номер шага от 1 до 38, иначе -1.

Начинаю сбор данных для второй задачи – для каждого запуска муравья я храню последовательность движения вне квадрата, так называемые «релевантные ходы», которые могут быть как последовательность движения между выходом и входом в квадрат, так и движение после выхода из квадрата до начала «хайвея». Для этого я фиксирую момент выхода и входа в квадрат, записываю промежуточные последовательности. Для каждой полученной последовательности сразу пишу все данные про нее в текстовый файл `output.txt`. Для каждого ухода в «хайвей» там содержится следующая информация – шаг «хайвея», с которого он начался, внешний шаг, то есть просто номер шага муравья, а также координаты муравья в этот момент – пара `x` и `y` и направление «хайвея» от 1 до 4 – 1 соответствует северо-восточному, далее по часовой стрелке поворот на 90 градусов. Следующие три строки содержат по 3 числа – раскраска квадрата 3 на 3 вокруг точки старта муравья. Далее идет количество строк в последовательности релевантных ходов, количество прерываний в последовательности и сами релевантные ходы в виде пары координат `x`, `y` и цвета точки.

Запустив цикл, пускающий муравья со случайной раскраски на 10.000 раз, каждый запуск на 4.000 шагов (после серии пробных запусков муравья на длительные периоды, когда я сохранял все раскраски, вне зависимости от того, начал муравей строить на них «хайвей» или нет, я проследил, что происходит с муравьем на значительно большем количестве шагов, на раскраске, когда на 4.000 он не начал строить «хайвей». Для таких условно «плохих» раскрасок муравей так и не начинал строить «хайвей» даже на 100.000 шагу. Таким образом, решив пожертвовать возможными случаями начала построения «хайвея» на поистине огромных числах ради производительности, я отказался от идеи запускать муравья каждый раз больше чем на 4000 шагов, тем более, что самым длительным пребыванием в хаотическом движении было около 2.400 шагов до «хайвея») я получил 5059 случаев вхождения в «хайвей» из 10000. Статистика ухода с конкретных шагов «хайвея» и

координат вокруг квадрата собрана в файле №5 из приложения. Обработав релевантные ходы, вывожу три самые часто встречающиеся последовательности – 38 раз с шага 8, 26 раз с 1 шага и 14 раз с 8. Каждая из них предстает как изначальная раскраска с данными о хайвее, потом последовательность входов и выходов из квадрата, где сам квадрат закрашен 7 цветом, на изображении последнего выхода из квадрата точка входа в хайвей отмечена 9 цветом (файл №6 из приложения).

## Заключение

В течение этой практики я приобрел достаточное количество знаний в различных сферах жизни – дискретной математике, программировании на C++, оптимизации и алгоритмов, и структур данных. Также, я стал ближе на шаг, пусть и маленький, к великим математикам, ведь только изучая что-то можно это что-то постигнуть. В данной работе высказывалась гипотеза, что муравей Лэнгтона начинает строить «хайвей» для любой конечной раскраски, таким образом я не опровергнул ее, но собрал какое-то количество данных, которые в будущем помогут кому-то, а возможно и мне дальше заниматься этой задачей. В ходе этой работы были собраны данные про квазипериодические движения муравья Лэнгтона на случайных раскрасках и смоделирован переборный алгоритм, определяющий вхождение муравья в «хайвей».

## Список литературы

1. Муравей Лэнгтона URL:  
[https://ru.wikipedia.org/wiki/Муравей\\_Лэнгтона](https://ru.wikipedia.org/wiki/Муравей_Лэнгтона)



## Приложение

Список файлов:

1. <https://github.com/birshert/hse/blob/Practice/part-1.cpp>
2. <https://github.com/birshert/hse/blob/Practice/part-2.cpp>
3. <https://github.com/birshert/hse/blob/Practice/0-colored%20field%20start.JPG>
4. <https://github.com/birshert/hse/blob/Practice/cycle.pdf>
5. <https://github.com/birshert/hse/blob/Practice/analysis.pdf>
6. [https://github.com/birshert/hse/blob/Practice/3\\_relevant.pdf](https://github.com/birshert/hse/blob/Practice/3_relevant.pdf)