**Министерство науки и высшего образования Российской Федерации**
**Федеральное государственное бюджетное образовательное учреждение высшего образования**
**«Московский государственный технический университет имени Н.Э. Баумана**
**(национальный исследовательский университет)»**
**(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

# Отчет по лабораторной работе №6 по курсу "Анализ алгоритмов"

**Тема** Муравьиный алгоритм

**Студент** Гурова Н.А.

**Группа** ИУ7-54Б

**Оценка (баллы)**

**Преподаватель** Волкова Л.Л., Строганов Ю.В.

Москва — 2022 г.

# Оглавление

# Введение

Современные средства навигации, организация логистики, конвейерного производства, анализ эффективности финансовых инструментов строятся на алгоритмах решения задачи поиска оптимального решения по выбранному параметру в сложной системе. Данную задачу высокой вычислительной сложности называют задачей коммивояжера [1].

Задачи высокой вычислительной сложности могут быть решены при помощи полного перебора вариантов и эвристических алгоритмов [2]. Смысл понятия "эвристический алгоритм" состоит в том, что в этом случае алгоритм не вытекает из строгих положений теории, а в значительной степени основан на интуиции и опыте. Такие методы могут давать удовлетворительные результаты при вероятностных параметрах. Алгоритмы, основанные на использовании эвристических алгоритмов, не всегда приводят к оптимальным решениям. Однако для их применения на практике достаточно, чтобы ошибка прогнозирования не превышала допустимого значения, а этого можно добиться, например, подбором более информативных параметров.

Целью данной лабораторной работы является реализация муравьиного алгоритма и приобретение навыков параметризации методов на примере реализованного алгоритма, примененного к задаче коммивояжера.

Для достижения данной цели необходимо решить следующие задачи.

1. Изучить алгоритм полного перебора для решения задачи коммивояжера.

2. Реализовать алгоритм полного перебора для решения задачи коммивояжера.

3. Изучить муравьиный алгоритм для решения задачи коммивояжера.

4. Реализовать муравьиный алгоритм для решения задачи коммивояжера.

5. Провести параметризацию муравьиного алгоритма на трех классах данных.

6. Провести сравнительный анализ скорости работы реализованных алгоритмов.

7. Подготовить отчет о выполненной лабораторной работе.

# 1 Аналитическая часть

В данном разделе была рассмотрена задача коммивояжера и были описаны алгоритмы её решения.

## 1.1 Задача коммивояжера

Цель задачи коммивояжера [3] заключается в нахождении самого выгодного маршрута (кратчайшего, самого быстрого, наиболее дешевого), проходящего через все заданные точки (пункты, города) по одному разу.

Условия задачи должны содержать критерий выгодности маршрута (должен ли он быть максимально коротким, быстрым, дешевым или все вместе), а также исходные данные в виде матрицы затрат (расстояния, стоимости, времени) при перемещении между рассматриваемыми пунктами.

### Математическая модель

Исходные условия можно представить в виде взвешенного графа - конечного множества вершин и множества ребер, соединяющих вершины. Вершины символизируют города, ребра - пути между городами, вес ребра - стоимость пути.

Рассмотрим взевешенный граф на рисунке 1.1. Самый выгодный маршрут для данного графа равен 14 ($1 \to 3 \to 2 \to 4 \to 1$).

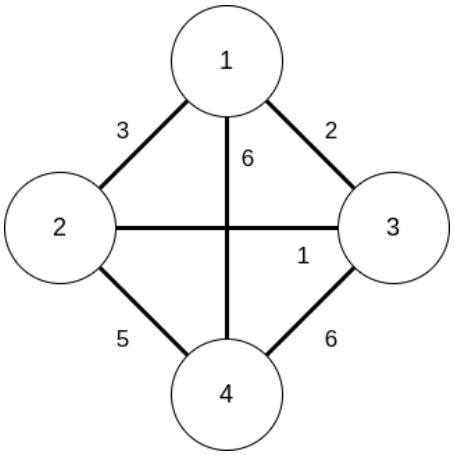


Рисунок 1.1 – Взвешенный граф

По симметричности задача коммивояжера бывает:

- симметричная - все пары ребер, соединяющие одни и те же вершины, имеют одинаковый вес (граф неориентированный);
- ассиметричная - вес пар ребер соединяющих одни и те же города, может различаться (ориентированный граф).

По замкнутости маршрута задача бывает:

- замкнутая - нахождение кратчайшего пути, проходящего через все вершины по одному разу с последующим возвратом в точку старта;
- незамкнутая — нахождение кратчайшего пути, проходящего через все вершины по одному разу и без обязательного возврата в исходную точку.

Далее будут рассмотрены алгоритмы решения симметричной замкнутой задачи коммивояжера.

## 1.2 Алгоритм полного перебора для решения задачи коммивояжера

Рассмотрим $n$ городов и матрицу расстояний между ними. Найдем самый короткий маршрут посещения всех городов ровно по одному разу, без возвращения в первый город:

- число вариантов выбрать первый город равно $n$;
- число вариантов выбрать второй город равно $n - 1$;
- с каждым выбором следующего города число вариантов уменьшается на 1;
- число всех вариантов маршрутра равно $n!$;
- минимальный по сумме значений матрицы расстояний вариант маршрута - искомый.

В связи со сложностью $n!$ полный перебор вариантов занимает существенное время, а при большом количестве городов становится технически невозможным.

## 1.3 Муравьиный алгоритм для решения задачи коммивояжера

Идея муравьиного алгоритма [4] — моделирование поведения муравьев, связанное с их способностью быстро находить кратчайший путь и адаптироваться к изменяющимся условиям, находя новый кратчайший путь.

Муравьи действуют согласно правилам:

- муравей запоминает посещенные города, причем каждый город может быть посещен только один раз. Обозначим через $J_{i,k}$ список городов, которые посетил муравей $k$, находящийся в городе $i$;
- муравей обладает видимостью $\eta_{ij}$ - эвристическим желанием посетить город $j$, если муравей находится в городе i, причем

$$\eta_{ij} = 1/D_{ij}, \tag{1.1}$$

  где $D_{ij}$ — стоимость пути из города $i$ в город $j$;
- муравей может улавливать след феромона - специального химического вещества. Число феромона на пути из города $i$ в город $j$ - $\tau_{ij}$.

Муравей выполняет следующую последовательность действий, пока не посетит все города:

- выбирает следующий город назначения, основываясь на вероятностно-пропорциональном правиле (1.2), в котором учитываются видимость и число феромона.

$$P_{ij,k} = \begin{cases} \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum_{l=1}^{m}\tau_{il}^{\alpha}\eta_{il}^{\beta}}, \text{если город j необходимо посетить;} \\ 0, \text{иначе,} \end{cases} \tag{1.2}$$

где $\alpha$ - параметр влияния феромона, $\beta$ - параметр влияния видимости пути, $\tau_{ij}$ - число феромона на ребре $(ij)$, $\eta_{ij}$ - эвристическое желание посетить город $j$, если муравей находится в городе $i$. Выбор города является вероятностным, данное правило определяет ширину зоны города $j$, в общую зону всех городов $J_{i,k}$ бросается случайное число, которое и определяет выбор муравья;

- муравей проходит путь $(ij)$ и оставляет на нем феромон.

Информация о числе феромона на пути используется другими муравьями для выбора пути. Те муравьи, которые случайно выберут кратчайший путь, будут быстрее его проходить, и за несколько передвижений он будет более обогащен феромоном. Следующие муравьи будут предпочитать именно этот путь, продолжая обогащать его феромоном.

После прохождения маршрутов всеми муравьями значение феромона на путях обновляется в соответствии со следующим правилом (1.3).

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \Delta\tau_{ij}, \tag{1.3}$$

где $\rho$ - коэффициент испарения. Чтобы найденное локальное решение не было единственным, моделируется испарение феромона.

При этом

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \tau_{ij,k}, \tag{1.4}$$

где $m$ - число муравьев,

$$\Delta\tau_{ij,k} = \begin{cases} Q/L_k, \text{если k-ый муравей прошел путь (i,j);} \\ 0, \text{иначе.} \end{cases} \tag{1.5}$$

# Вывод

Была изучена задача поиска оптимального маршрута, проходящего через все заданные вершины по одному разу. Были рассмотрены подходы к решению замкнутой симметричной задачи коммивояжера.

# 2 Конструкторская часть

В этом разделе были представлены требования к вводу и программе, а также схемы алгоритма полного перебора и муравьиного алгоритма.

## 2.1 Требования к вводу

На вход программе должна подаваться матрица стоимостей, которая задает взвешенный неориентированный граф.

## 2.2 Требования к программе

Выходные данные программы – оптимальный маршрут, проходящий через все заданные вершины по одному разу с последующим возвратом в исходную точку, и его стоимость. Программа должна работать в рамках следующих ограничений:

- стоимости путей должны быть целыми числами;
- число городов должно быть больше 1;
- число дней должно быть больше 0;
- параметры муравьиного алгоритма должны быть вещественными числами, большими 0;
- матрица должна задавать неориентированный граф;
- должно быть выдано сообщение об ошибке при некорректном вводе параметров.

Пользователь должен иметь возможность выбора метода решения - полным перебором или муравьиным алгоритмом, и вывода результата на экран. Кроме того должна быть возможность проведения параметризации

муравьиного алгоритма. Также должны быть реализованы сравнение алгоритмов по времени работы с выводом результатов на экран и получение графического представления результатов сравнения. Данные действия пользователь должен выполнять при помощи меню.

## 2.3 Разработка алгоритмов

На рисунке 2.1 представлена схема алгоритма полного перебора путей, а на рисунках 2.2–2.3 схема муравьиного алгоритма поиска путей. Также на рисунках 2.4–2.6 представлены схемы вспомогательных функций для муравьиного алгоритма.

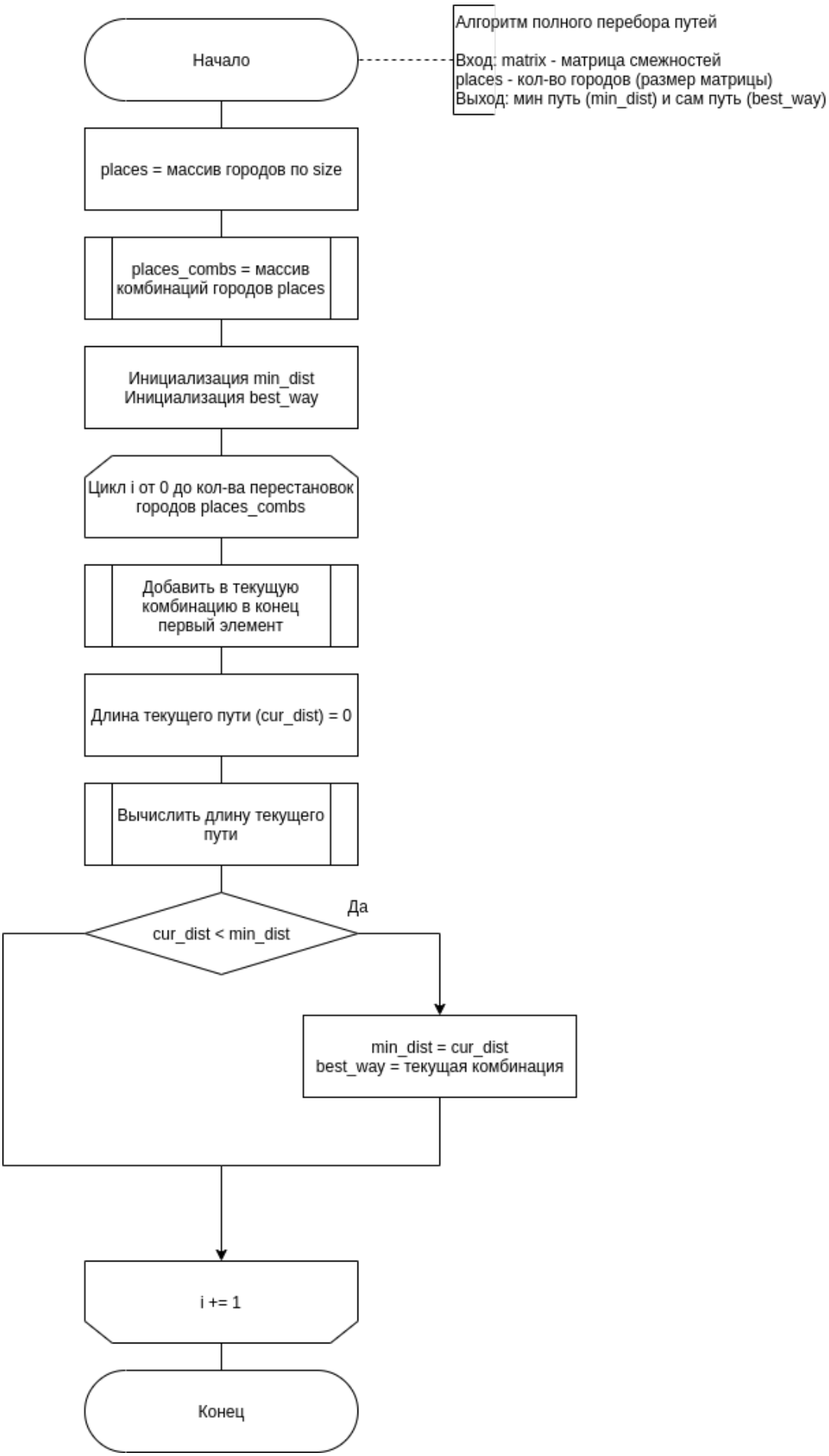


Рисунок 2.1 – Схема алгоритма полного перебора путей

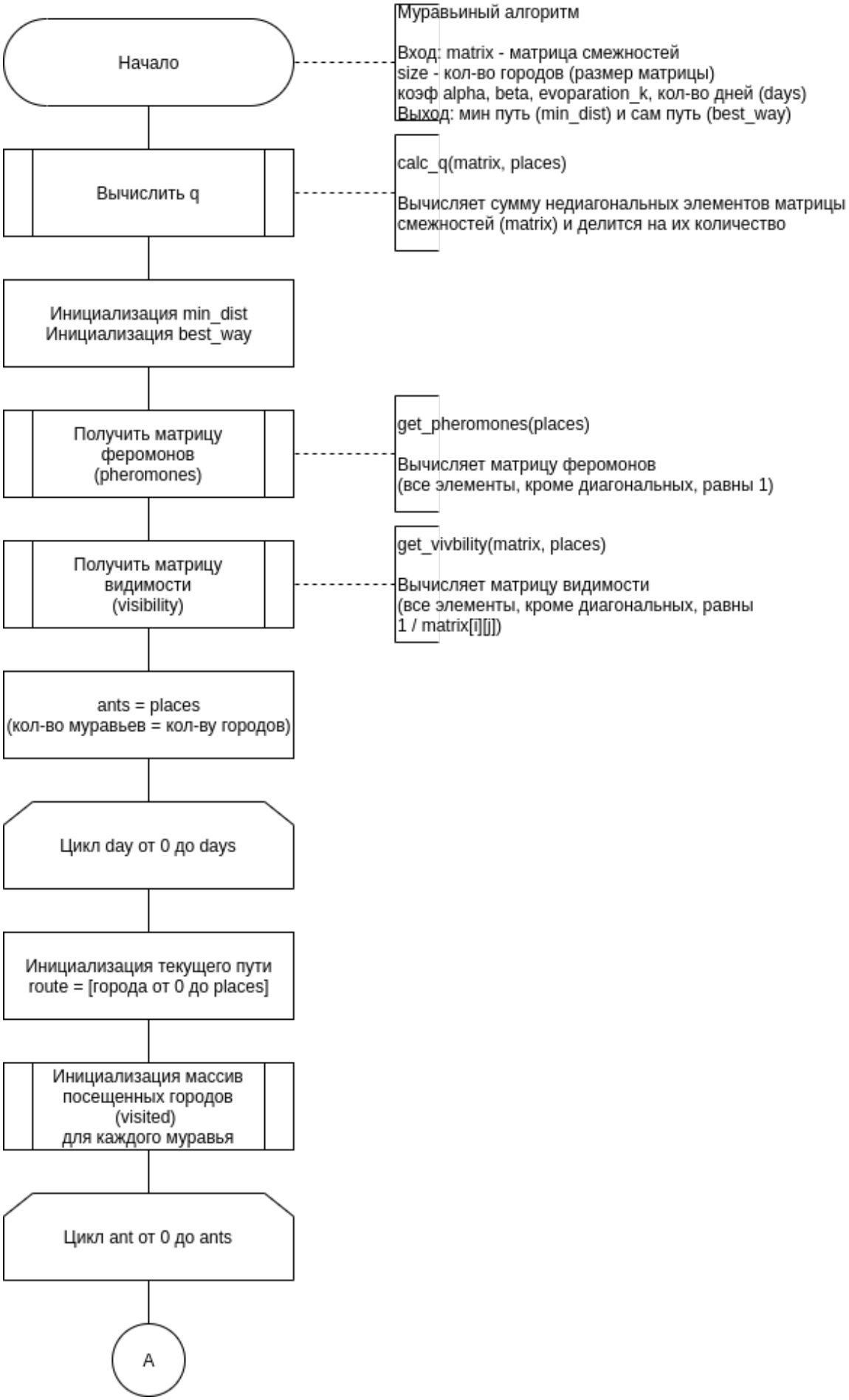


Рисунок 2.2 – Схема муравьиного алгоритма (часть 1)

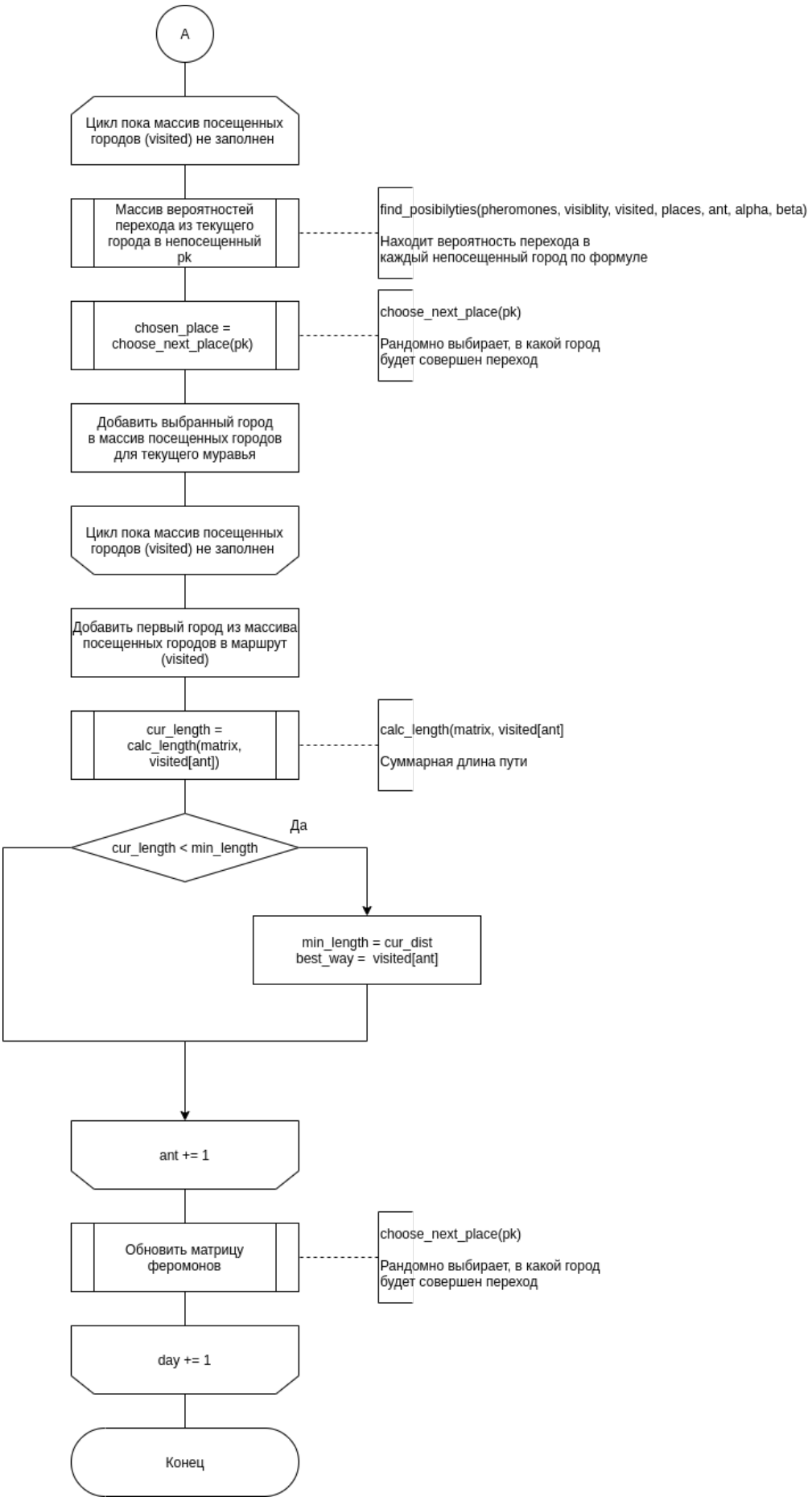


Рисунок 2.3 – Схема муравьиного алгоритма (часть 2)

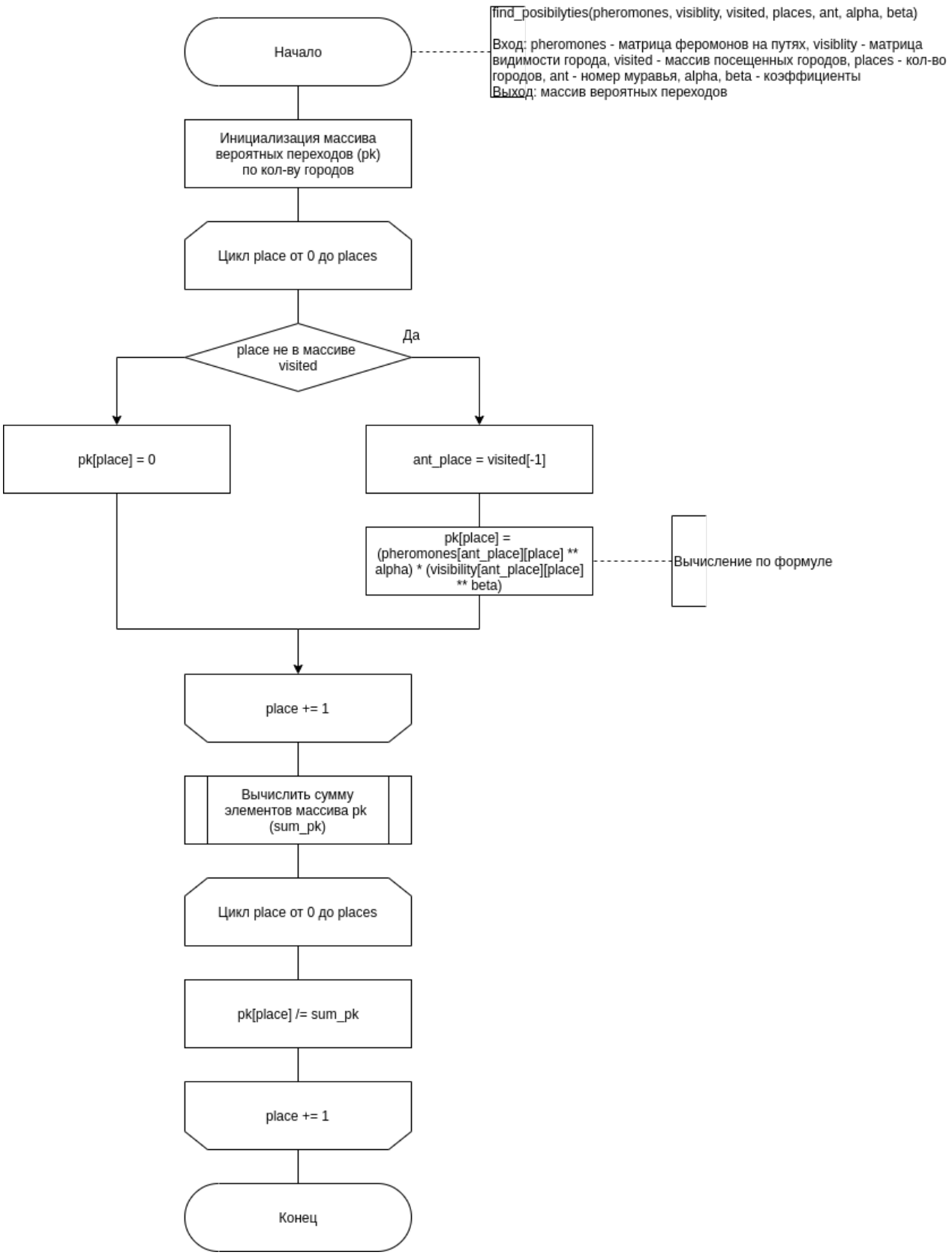


Рисунок 2.4 – Схема алгоритма нахождения массива вероятностных переходов в непосещенные города

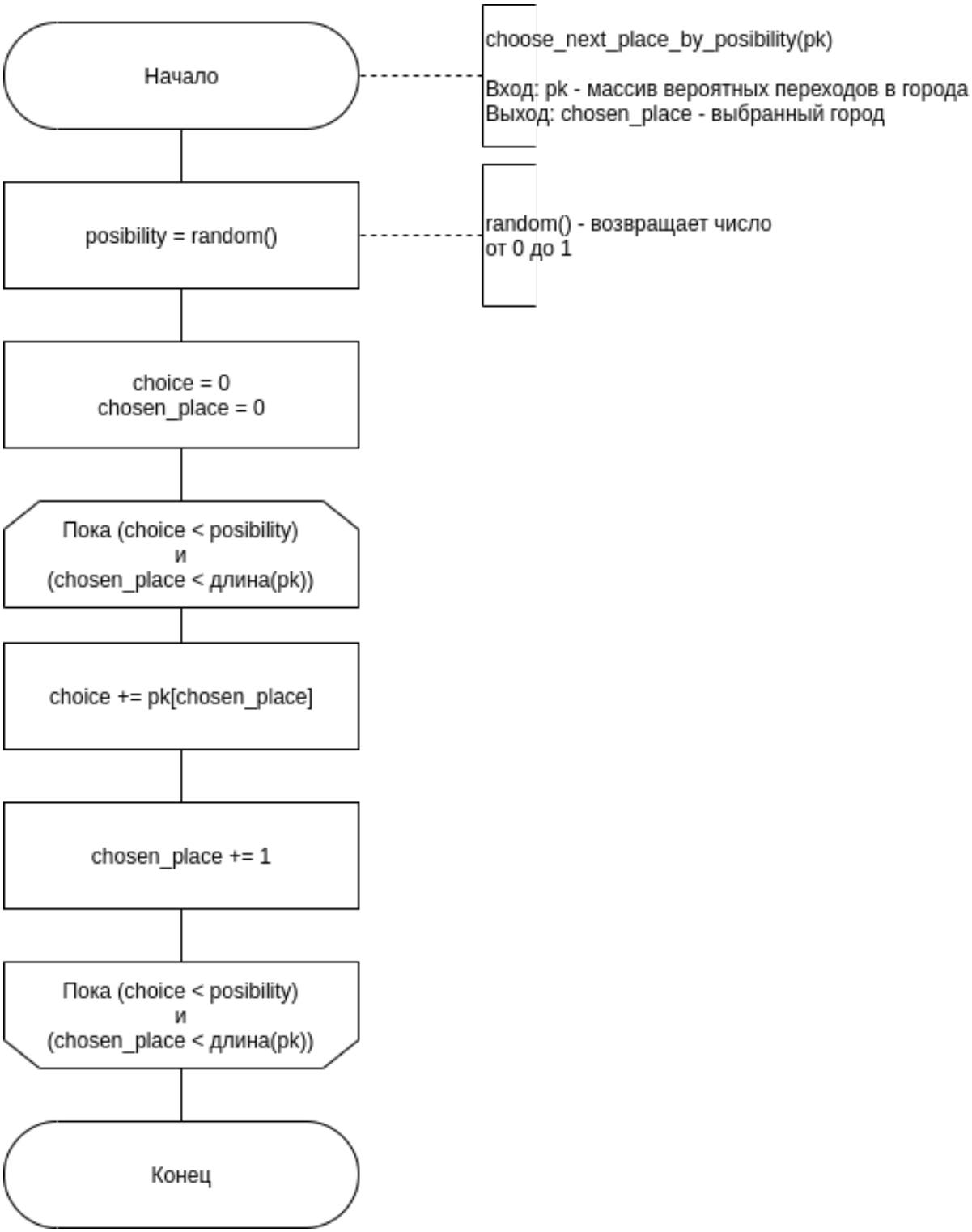


Рисунок 2.5 – Схема алгоритма нахождения следующего города на основании рандома

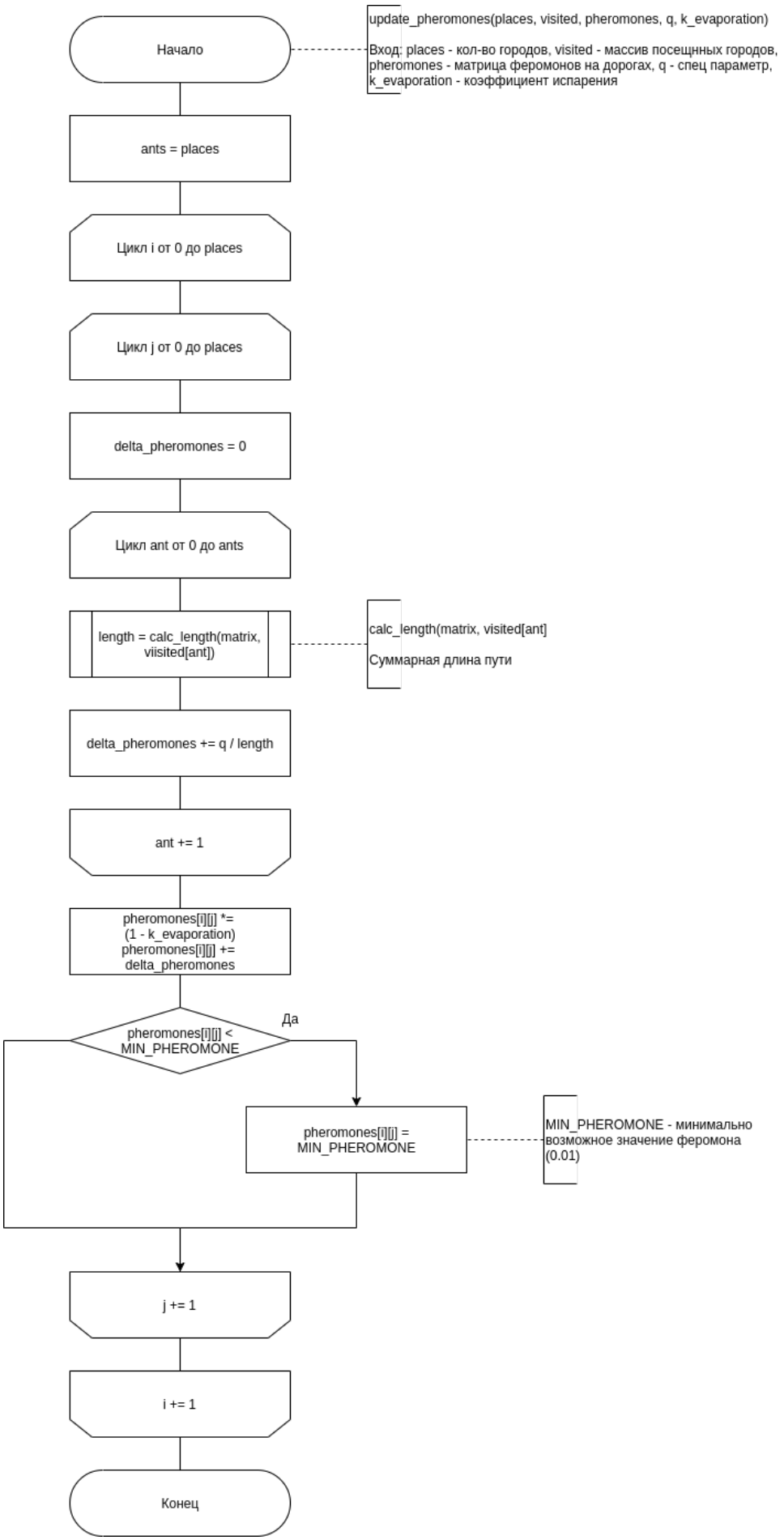


Рисунок 2.6 – Схема алгоритма обновления матрицы феромонов

# Вывод

В данном разделе были построены схемы алгоритмов, рассматриваемых в лабораторной работе, были описаны требования ко вводу и программе.

# 3 Технологическая часть

В данном разделе были приведены средства реализации и листинги кода.

## 3.1 Средства реализации

В качестве языка программирования для реализации данной лабораторной работы был выбран язык программирования C++ [5]. Данный язык имеет все небходимые инструменты для решения поставленной задачи.

## 3.2 Сведения о модулях программы

Программа состоит из четырех модулей:

1. main.cpp - главный файл программы, в котором располагается точка входа;
2. algorithms.cpp - хранит реализацию алгоритмов;
3. unit_tests.cpp - хранит реализацию тестирующей системы и тесты;
4. tools.cpp - хранит реализацию вспомогательных функций.

## 3.3 Реализация алгоритмов

В листинге 3.1 представлен алгоритм полного перебора.

Листинг 3.1 – Реализация алгоритма полного перебора

```
1 sarray get_shortest_path(sarray *cities, int matrix[N][N])
2 {
3     sarray result[MAXRECURSION];
4     sarray res_;
5
6     int min_index = 0;
7     int min_cost;
```

```
8       int cur_cost;
9
10      int len_routes = 0;
11
12      del_element(cities, 0);
13      add_element(&res_, 0);
14
15      get_routes(cities, &res_, result, &len_routes);
16      min_cost = get_cost(result[min_index], matrix);
17
18      for (int i = 1; i < len_routes; i++)
19      {
20          cur_cost = get_cost(result[i], matrix);
21          if (cur_cost < min_cost)
22          {
23              min_cost = cur_cost;
24              min_index = i;
25          }
26      }
27      return result[min_index];
28  }
```

В листинге 3.2 представлен алгоритм нахождения всех перестановок в графе.

Листинг 3.2 – Реализация алгоритма нахождения всех перестановок в графе

```
1  void get_routes(sarray *cities, sarray *res_,
2  sarray result[MAXRECURSION], int *len)
3  {
4      int element;
5      sarray buf;
6
7      if (!cities->len)
8      {
9          add_element(res_, get_element(*res_, 0));
10         result[*len] = *res_;
11         (*len)++;
12         del_element(res_, res_->len - 1);
13     }
14
15     for (int i = 0; i < cities->len - 1; i++)
```

```
16      {
17          element = get_element(*cities, i);
18          add_element(res_, element);
19          buf = copy_array(*cities);
20          del_element(&buf, i);
21          get_routes(&buf, res_, result, len);
22          del_element(res_, res_->len - 1);
23      }
24  }
```

В листинге 3.3 представлена реализация муравьиного алгоритма.

Листинг 3.3 – Реализация муравьиного алгоритма

```
1  sarray ant_alg(int matrix[N][N], int len, sarray cities, int tmax,
2  float p, float alpha, float beta)
3  {
4      int q = calc_q(matrix, len);
5      sarray best_way = copy_array(cities);
6      add_element(&best_way, get_element(best_way, 0));
7
8      int best_cost = get_cost(best_way, matrix);
9      int cur_cost = 0;
10     float matrix_p[N][N];
11
12     fill_matrix(matrix_p, len, MINPHENOM);
13     sant ants[MAXCOUNTOFANT];
14
15     for (int t = 0; t < tmax; t++)
16     {
17         generate_ants_array(ants, len);
18         for (int i = 0; i < len - 1; i++)
19         {
20             ants_choose_way(ants, matrix_p, matrix, len, alpha,
                   beta);
21         }
22
23         for (int i = 0; i < len; i++)
24         {
25             add_element(&ants[i].way, get_element(ants[i].way,
                   0));
26         }
27
```

```
28          for (int i = 0; i < len; i++)
29          {
30              cur_cost = get_cost(ants[i].way, matrix);
31
32              if (cur_cost < best_cost)
33              {
34                  best_cost = cur_cost;
35
36                  best_way = copy_array(ants[i].way);
37              }
38          }
39
40          delete_phenom(matrix_p, len, p);
41          add_phenom(matrix, matrix_p, len, q, ants);
42          phenom_correct(matrix_p, len);
43      }
44      return best_way;
45  }
```

В листинге 3.4 представлена реализация алгоритма выбора оптимального пути для муравья.

Листинг 3.4 – Реализация алгоритма выбора оптимального пути для муравья

```
1  void next_city(sant *ants, float matrix_p[N][N],
2  int matrix[N][N], float alpha, float beta)
3  {
4      float number = 0;
5      float denominator = 0;
6      float tao, rev_cost;
7      int cost;
8      int cur_city = ants->way.array[ants->way.len - 1];
9
10     for (int i = 0; i < ants->quere.len; i++)
11     {
12         cost = matrix[cur_city][ants->quere.array[i]];
13         tao = matrix_p[cur_city][ants->quere.array[i]];
14
15         if (!cost)
16             continue;
17
18         rev_cost = 1.0 / cost;
```

```
19          denominator += powf(tao, alpha) + powf(rev_cost, beta);
20      }
21
22      float p_array[N] = {0};
23      float sum = 0;
24
25      for (int i = 0; i < ants->quere.len; i++)
26      {
27          cost = matrix[cur_city][ants->quere.array[i]];
28          tao = matrix_p[cur_city][ants->quere.array[i]];
29
30          rev_cost = 1.0 / cost;
31
32          p_array[i] = (powf(tao, alpha) + powf(rev_cost, beta)) /
                denominator;
33      }
34
35      float x = (float)rand() / RAND_MAX;
36      int index = 0;
37
38      while (x >= 0)
39      {
40          x -= p_array[index];
41          index++;
42      }
43
44      add_element(&ants->way, get_element(ants->quere, index - 1));
45      del_element(&ants->quere, index - 1);
46  }
```

В листинге 3.5 представлена реализация вспомогательных функций.

Листинг 3.5 – Вспомогательные функции

```
1  void add_phenom(int matrix[N][N], float matrix_p[N][N], int len,
2  int q, sant ants[MAXCOUNTOFANT])
3  {
4      int fcity, scity;
5      int cur_cost;
6      float dtao = 0;
7
8      for (int i = 0; i < len; i++)
9      {
```

```
10          cur_cost = get_cost(ants[i].way, matrix);
11          dtao += (float)q / cur_cost;
12      }
13
14      for (int i = 0; i < len; i++)
15      {
16          for (int j = 0; j < ants[i].way.len - 1; j++)
17          {
18              fcity = ants[i].way.array[j];
19              scity = ants[i].way.array[j + 1];
20
21              matrix_p[fcity][scity] = matrix_p[fcity][scity] +
                    dtao;
22              matrix_p[scity][fcity] = matrix_p[scity][fcity] +
                    dtao;
23          }
24      }
25  }
26
27  void delete_phenom(float matrix_p[N][N], int len, float p)
28  {
29      float qq = 1 - p;
30
31      for (int i = 0; i < len; i++)
32      {
33          for (int j = 0; j < i; j++)
34          {
35              matrix_p[i][j] = matrix_p[i][j] * qq;
36              matrix_p[j][i] = matrix_p[j][i] * qq;
37          }
38      }
39  }
40
41  void phenom_correct(float matrix_p[N][N], int len)
42  {
43      for (int i = 0; i < len; i++)
44      {
45          for (int j = 0; j < i; j++)
46          {
47              if (matrix_p[i][j] <= 0.1)
48              {
```

```
49                    matrix_p[i][j] = matrix_p[j][i] = 0.1;
50                }
51            }
52        }
53    }
```

## 3.4 Тестирование

В таблице 3.1 приведены тесты для функции, реализующей алгоритм для решения задачи коммивояжера (алгоритм полного перебора и муравьиный алгоритм). Тесты пройдены успешно.

При тестировании муравьиного алгоритма тесты прошли успешно, так как матрицы смежности маленькие, и поэтому муравьиный алгоритм работает без видимых ошибок, то есть выдает такие же результаты как и алгоритм полного перебора.

Ошибки в результатах муравьиного алгоритма чаще всего возникают при больших размерах матриц (начиная от 9x9). Погрешности бывают чаще всего как разность между наименьшим путем и предпоследним (тот который мог бы быть наименьшим).

Таблица 3.1 – Функциональные тесты

| Матрица смежности | Ожидаемый результат | Действительный результат |
|---|---|---|
| $\begin{pmatrix} 0 & 1 & 10 & 7 \\ 1 & 0 & 1 & 2 \\ 10 & 1 & 0 & 1 \\ 7 & 2 & 1 & 0 \end{pmatrix}$ | 10, [0, 1, 2, 3, 0] | 10, [0, 1, 2, 3, 0] |
| $\begin{pmatrix} 0 & 3 & 5 & 7 \\ 7 & 0 & 1 & 2 \\ 5 & 1 & 0 & 1 \\ 7 & 2 & 1 & 0 \end{pmatrix}$ | 11, [2, 3, 1, 0, 2] | 11, [2, 3, 1, 0, 2] |
| $\begin{pmatrix} 0 & 3 & 5 \\ 3 & 0 & 1 \\ 5 & 1 & 0 \end{pmatrix}$ | 9, [0, 1, 2, 0] | 9, [0, 1, 2, 0] |

# Вывод

В данном разделе были представлены реализации алгоритма полного перебора и муравьиного алгоритма. Алгоритмы были протестированы.

# 4 Исследовательская часть

В данном разделе были произведен сравнительный анализ алгоритмов.

## 4.1 Технические характеристики

Технические характеристики устройства, на котором выполнялось тестирование:

- Операционная система macOS Monterey 12.5.1
- Память 16 Гб.
- Процессор 2,3 ГГц 4-ядерный процессор Intel Core i5.

Во время тестирования устройство было подключено к сети электропитания, нагружено приложениями окружения и самой системой тестирования.

## 4.2 Время выполнения алгоритмов

Для замера процессорного времени использовалась функция *std::chrono::system_ clock::now(...)* из библиотеки *chrono* [?] на C++. Функция возвращает процессорное время типа float в секундах.

Контрольная точка возвращаемого значения не определена, поэтому допустима только разница между результатами последовательных вызовов.

Замеры времени для каждой длины входного массива полигонов проводились 1000 раз. В качестве результата взято среднее время работы алгоритма на данной длине. При каждом запуске алгоритма, на вход подавались случайно сгенерированные массивы полигонов.

Результаты замеров приведены в таблице 4.1.

Таблица 4.1 – Результаты замеров времени

| Размер | Полный перебор | Муравьиный алгоритм |
|---|---|---|
| 2 | 0.000130 | 0.019932 |
| 3 | 0.000138 | 0.031615 |
| 4 | 0.000104 | 0.044361 |
| 5 | 0.000420 | 0.089291 |
| 6 | 0.002390 | 0.152131 |
| 7 | 0.019703 | 0.254059 |
| 8 | 0.162850 | 0.398472 |
| 9 | 1.637611 | 0.594024 |
| 10 | 18.207853 | 0.857666 |

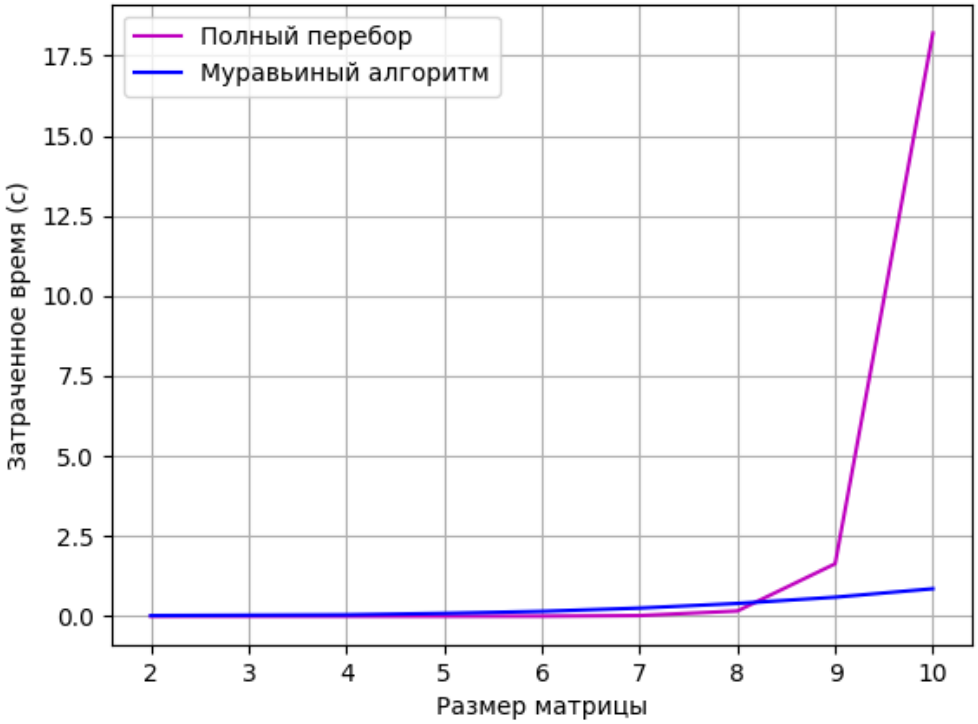


Рисунок 4.1 – Сравнение по времени алгоритмов полного перебора путей и муравьиного на разных размерах матриц

## 4.3 Параметризация

Целью проведения параметризации является определение таких комбинаций параметров, при которых муравьиный алгоритм даёт наилучшие результаты.

В результате автоматической параметризации будет получена таблицы со следующими столбцами:

- коэффициент видимости $\alpha$ - изменяющийся параметр;
- коэффициент испарения феромона $\rho$ - изменяющийся параметр;
- число дней $days$ - изменяющийся параметр;
- эталонный результат $ideal$;
- разность полученного при данных параметрах значения и эталонного $mistake$.

### 4.3.1 Класс данных 1

Класс данных 1 представляет собой матрицу стоимостей в диапозоне от 1 до 5 для 8 городов:

$$K_1 = \begin{pmatrix} 0 & 4 & 3 & 2 & 4 & 5 & 5 & 3 \\ 4 & 0 & 4 & 5 & 5 & 1 & 4 & 5 \\ 3 & 4 & 0 & 5 & 4 & 1 & 2 & 1 \\ 2 & 5 & 5 & 0 & 1 & 3 & 1 & 5 \\ 4 & 5 & 4 & 1 & 0 & 2 & 5 & 4 \\ 5 & 1 & 1 & 3 & 2 & 0 & 4 & 5 \\ 5 & 4 & 2 & 1 & 5 & 4 & 0 & 2 \\ 3 & 5 & 1 & 5 & 4 & 5 & 2 & 0 \end{pmatrix} \tag{4.1}$$

Для данного класса данных приведена таблица с выборкой параметров, которые наилучшим образом решают поставленную задачу.

Таблица 4.2 – Параметры для класса данных 1

| $\alpha$ | $\rho$ | Days | Result | Mistake |
|---|---|---|---|---|
| 0.1 | 0.9 | 100 | 15 | 0 |
| 0.1 | 0.9 | 200 | 15 | 0 |
| 0.1 | 0.9 | 300 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.1 | 0.9 | 400 | 15 | 0 |
| 0.1 | 0.9 | 500 | 15 | 0 |
| 0.2 | 0.8 | 100 | 15 | 0 |
| 0.2 | 0.8 | 200 | 15 | 0 |
| 0.2 | 0.8 | 300 | 15 | 0 |
| 0.2 | 0.8 | 400 | 15 | 0 |
| 0.2 | 0.8 | 500 | 15 | 0 |
| 0.3 | 0.7 | 100 | 15 | 0 |
| 0.3 | 0.7 | 200 | 15 | 0 |
| 0.3 | 0.7 | 300 | 15 | 0 |
| 0.3 | 0.7 | 400 | 15 | 0 |
| 0.3 | 0.7 | 500 | 15 | 0 |
| 0.4 | 0.6 | 100 | 15 | 0 |
| 0.4 | 0.6 | 200 | 15 | 0 |
| 0.4 | 0.6 | 300 | 15 | 0 |
| 0.4 | 0.6 | 400 | 15 | 0 |
| 0.4 | 0.6 | 500 | 15 | 0 |
| 0.5 | 0.5 | 100 | 15 | 0 |
| 0.5 | 0.5 | 200 | 15 | 0 |
| 0.5 | 0.5 | 300 | 15 | 0 |
| 0.5 | 0.5 | 400 | 15 | 0 |
| 0.5 | 0.5 | 500 | 15 | 0 |
| 0.6 | 0.4 | 100 | 15 | 0 |
| 0.6 | 0.4 | 200 | 15 | 0 |
| 0.6 | 0.4 | 300 | 15 | 0 |
| 0.6 | 0.4 | 400 | 15 | 0 |
| 0.6 | 0.4 | 500 | 15 | 0 |
| 0.7 | 0.3 | 100 | 15 | 0 |
| 0.7 | 0.3 | 200 | 15 | 0 |
| 0.7 | 0.3 | 300 | 15 | 0 |
| 0.7 | 0.3 | 400 | 15 | 0 |
| 0.7 | 0.3 | 500 | 15 | 0 |
| 0.8 | 0.2 | 100 | 15 | 0 |
| 0.8 | 0.2 | 200 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.8 | 0.2 | 300 | 15 | 0 |
| 0.8 | 0.2 | 400 | 15 | 0 |
| 0.8 | 0.2 | 500 | 15 | 0 |
| 0.9 | 0.1 | 100 | 15 | 0 |
| 0.9 | 0.1 | 200 | 15 | 0 |
| 0.9 | 0.1 | 300 | 15 | 0 |
| 0.9 | 0.1 | 400 | 15 | 0 |
| 0.9 | 0.1 | 500 | 15 | 0 |

### 4.3.2 Класс данных 2

Класс данных 2 представляет собой матрицу стоимостей в диапозоне от 5000 до 9000 для 8 городов:

$$K_1 = \begin{pmatrix} 0 & 5466 & 8308 & 8068 & 7284 & 5635 & 6055 & 8129 \\ 5466 & 0 & 8205 & 7384 & 6794 & 6048 & 6174 & 6306 \\ 8308 & 8205 & 0 & 5485 & 7872 & 7981 & 7868 & 6912 \\ 8068 & 7384 & 5485 & 0 & 7002 & 6683 & 7544 & 8278 \\ 7284 & 6794 & 7872 & 7002 & 0 & 5159 & 8240 & 5663 \\ 5635 & 6048 & 7981 & 6683 & 5159 & 0 & 8801 & 8844 \\ 6055 & 6174 & 7868 & 7544 & 8240 & 8801 & 0 & 5493 \\ 8129 & 6306 & 6912 & 8278 & 5663 & 8844 & 5493 & 0 \end{pmatrix} \tag{4.2}$$

Для данного класса данных приведена таблица с выборкой параметров, которые наилучшим образом решают поставленную задачу.

Таблица 4.3 – Параметры для класса данных 2

| $\alpha$ | $\rho$ | days | ideal | mistake |
|---|---|---|---|---|
| 0.1 | 0.7 | 300 | 47326 | 0 |
| 0.1 | 0.7 | 400 | 47326 | 0 |
| 0.1 | 0.7 | 500 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.2 | 0.5 | 300 | 47326 | 0 |
| 0.2 | 0.5 | 400 | 47326 | 0 |
| 0.2 | 0.5 | 500 | 47326 | 0 |
| 0.3 | 0.3 | 300 | 47326 | 0 |
| 0.3 | 0.3 | 400 | 47326 | 0 |
| 0.3 | 0.3 | 500 | 47326 | 0 |
| 0.4 | 0.1 | 300 | 47326 | 0 |
| 0.4 | 0.1 | 400 | 47326 | 0 |
| 0.4 | 0.1 | 500 | 47326 | 0 |
| 0.5 | 0.6 | 300 | 47326 | 0 |
| 0.5 | 0.6 | 400 | 47326 | 0 |
| 0.5 | 0.6 | 500 | 47326 | 0 |
| 0.6 | 0.8 | 300 | 47326 | 0 |
| 0.6 | 0.8 | 400 | 47326 | 0 |
| 0.6 | 0.8 | 500 | 47326 | 0 |
| 0.7 | 0.2 | 300 | 47326 | 0 |
| 0.7 | 0.2 | 400 | 47326 | 0 |
| 0.7 | 0.2 | 500 | 47326 | 0 |
| 0.8 | 0.6 | 300 | 47326 | 0 |
| 0.8 | 0.6 | 400 | 47326 | 0 |
| 0.8 | 0.6 | 500 | 47326 | 0 |
| 0.9 | 0.9 | 300 | 47326 | 0 |
| 0.9 | 0.9 | 400 | 47326 | 0 |
| 0.9 | 0.9 | 500 | 47326 | 0 |

# Вывод

В результате эксперимента было получено, что для 2 городов полный перебор работает быстрее муравьиного алгоритма в 32 раза. Начиная с 8 городов, муравьиный алгоритм работает быстрее полного перебора: в 75 раз быстрее для 10 городов. Таким образом, муравьиный алгоритм необходимо

использовать при большом числе городов - от 8 и более.

Также в результате проведения параметризации было установлено, что для первого класса данных лучшие результаты муравьиный алгоритм дает на следующих значениях параметров:

- $\alpha = 0.1$, $\rho = 0.1$-$0.5$, $0.7$-$0.9$;
- $\alpha = 0.2$, $\rho = 0.1$-$0.7$, $0.9$;
- $\alpha = 0.3$, $\rho = 0.2$-$0.6$, $0.9$;
- $\alpha = 0.4$, $\rho = 0.5$-$0.9$;
- $\alpha = 0.5$, $\rho = 0.1$, $0.5$-$0.9$;
- $\alpha = 0.7$, $\rho = 0.4$-$0.8$.

Можно сделать вывод о том, что данные значения параметров следует использовать для матриц стоимостей в диапазоне от 1 до 5.

Для второго класса данных лучшие результаты муравьиный алгоритм дает на следующих значениях параметров:

- $\alpha = 0.1$, $\rho = 0.1$, $0.4$, $0.7$;
- $\alpha = 0.2$, $\rho = 0.3$, $0.7$, $0.9$;
- $\alpha = 0.3$, $\rho = 0.2$, $0.6$, $0.9$;
- $\alpha = 0.4$, $\rho = 0.7$, $0.8$;
- $\alpha = 0.5$, $\rho = 0.2$, $0.6$-$0.9$;
- $\alpha = 0.5$, $\rho = 0.3$, $0.7$-$0.9$;
- $\alpha = 0.6$, $\rho = 0.2$, $0.6$-$0.9$;
- $\alpha = 0.8$, $\rho = 0.1$, $0.6$;
- $\alpha = 0.9$, $\rho = 0.9$.

Можно сделать вывод о том, что данные значения параметров следует использовать для матриц стоимостей в диапазоне от 5000 до 9000.

Из результатов параметризации видно, что погрешность результата уменьшается при большом числе дней и меньшем значении коэффициента видимости.

# Заключение

В результате исследования было получено, что эвристический метод на базе муравьиного алгоритма следует использовать при большом количество городов - от 8 и более, так как для 2 городов мураьвиный алгоритм работает медленнее полного перебора в 32 раза, а для 10 городов - быстрее в 75 раз.

Лучшие значения муравьиный алгоритм показывает при меньших значениях коэффициента видимости и при большом числе дней.

Цель, поставленная перед началом работы, была достигнута. В ходе лабораторной работы были решены следующие задачи:

1. Изучены основы алгоритма полного перебора.
2. Применены изученные основы алгоритма полного перебора для реализации.
3. Изучены основы муравьиного алгоритма.
4. Применены изученные основы муравьиного алгоритма для реализации.
5. Проведена параметризация муравьиного алгоритма.
6. Проведен сравнительный анализ времени работы реализованных алгоритмов.
7. Подготовлен отчет о лабораторной работе.

Поставленная цель достигнута.

# Список использованных источников

[1] Что такое «задача коммивояжёра» [Электронный ресурс]. Режим доступа: `https://thecode.media/komm/` (дата обращения: 5.12.2021).

[2] Дубровин ВИ, Субботин СА. Эвристический алгоритм классификации и его нейросетевая интерпретация // Радіоелектроніка, інформатика, управління. 2000. № 1 (3). С. 72–76.

[3] Задача коммивояжера - метод ветвей и границ. Режим доступа: `http://galyautdinov.ru/post/zadacha-kommivoyazhera` (дата обращения: 5.12.2021).

[4] М.В. Ульянов Ресурсно-эффективные компютерные алгоритмы. Разработка и анализ. 2007.

[5] Рэнди Дэвис Стефан. C++ для чайников. Для чайников. Вильямс, 2018. с. 400.

# Приложение 1

Таблица 4.4 – Параметризация для класса данных 1

| $\alpha$ | $\rho$ | days | ideal | mistake |
|---|---|---|---|---|
| 0.1 | 0.1 | 25 | 15 | 0 |
| 0.1 | 0.1 | 50 | 15 | 0 |
| 0.1 | 0.1 | 100 | 15 | 0 |
| 0.1 | 0.1 | 200 | 15 | 0 |
| 0.1 | 0.1 | 300 | 15 | 0 |
| 0.1 | 0.1 | 400 | 15 | 0 |
| 0.1 | 0.1 | 500 | 15 | 0 |
| 0.1 | 0.2 | 10 | 15 | 0 |
| 0.1 | 0.2 | 25 | 15 | 0 |
| 0.1 | 0.2 | 50 | 15 | 0 |
| 0.1 | 0.2 | 100 | 15 | 0 |
| 0.1 | 0.2 | 200 | 15 | 0 |
| 0.1 | 0.2 | 300 | 15 | 0 |
| 0.1 | 0.2 | 400 | 15 | 0 |
| 0.1 | 0.2 | 500 | 15 | 0 |
| 0.1 | 0.3 | 10 | 15 | 0 |
| 0.1 | 0.3 | 25 | 15 | 0 |
| 0.1 | 0.3 | 50 | 15 | 0 |
| 0.1 | 0.3 | 100 | 15 | 0 |
| 0.1 | 0.3 | 200 | 15 | 0 |
| 0.1 | 0.3 | 300 | 15 | 0 |
| 0.1 | 0.3 | 400 | 15 | 0 |
| 0.1 | 0.3 | 500 | 15 | 0 |
| 0.1 | 0.4 | 10 | 15 | 0 |
| 0.1 | 0.4 | 25 | 15 | 0 |
| 0.1 | 0.4 | 50 | 15 | 0 |
| 0.1 | 0.4 | 100 | 15 | 0 |
| 0.1 | 0.4 | 200 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.1 | 0.4 | 300 | 15 | 0 |
| 0.1 | 0.4 | 400 | 15 | 0 |
| 0.1 | 0.4 | 500 | 15 | 0 |
| 0.1 | 0.5 | 10 | 15 | 0 |
| 0.1 | 0.5 | 25 | 15 | 0 |
| 0.1 | 0.5 | 50 | 15 | 0 |
| 0.1 | 0.5 | 100 | 15 | 0 |
| 0.1 | 0.5 | 200 | 15 | 0 |
| 0.1 | 0.5 | 300 | 15 | 0 |
| 0.1 | 0.5 | 400 | 15 | 0 |
| 0.1 | 0.5 | 500 | 15 | 0 |
| 0.1 | 0.6 | 10 | 15 | 2 |
| 0.1 | 0.6 | 25 | 15 | 0 |
| 0.1 | 0.6 | 50 | 15 | 0 |
| 0.1 | 0.6 | 100 | 15 | 0 |
| 0.1 | 0.6 | 200 | 15 | 0 |
| 0.1 | 0.6 | 300 | 15 | 0 |
| 0.1 | 0.6 | 400 | 15 | 0 |
| 0.1 | 0.6 | 500 | 15 | 0 |
| 0.1 | 0.7 | 10 | 15 | 0 |
| 0.1 | 0.7 | 25 | 15 | 0 |
| 0.1 | 0.7 | 50 | 15 | 0 |
| 0.1 | 0.7 | 100 | 15 | 0 |
| 0.1 | 0.7 | 200 | 15 | 0 |
| 0.1 | 0.7 | 300 | 15 | 0 |
| 0.1 | 0.7 | 400 | 15 | 0 |
| 0.1 | 0.7 | 500 | 15 | 0 |
| 0.1 | 0.8 | 10 | 15 | 0 |
| 0.1 | 0.8 | 25 | 15 | 0 |
| 0.1 | 0.8 | 50 | 15 | 0 |
| 0.1 | 0.8 | 100 | 15 | 0 |
| 0.1 | 0.8 | 200 | 15 | 0 |
| 0.1 | 0.8 | 300 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.1 | 0.8 | 400 | 15 | 0 |
| 0.1 | 0.8 | 500 | 15 | 0 |
| 0.1 | 0.9 | 10 | 15 | 0 |
| 0.1 | 0.9 | 25 | 15 | 0 |
| 0.1 | 0.9 | 50 | 15 | 0 |
| 0.1 | 0.9 | 100 | 15 | 0 |
| 0.1 | 0.9 | 200 | 15 | 0 |
| 0.1 | 0.9 | 300 | 15 | 0 |
| 0.1 | 0.9 | 400 | 15 | 0 |
| 0.1 | 0.9 | 500 | 15 | 0 |
| 0.2 | 0.1 | 10 | 15 | 0 |
| 0.2 | 0.1 | 25 | 15 | 0 |
| 0.2 | 0.1 | 50 | 15 | 0 |
| 0.2 | 0.1 | 100 | 15 | 0 |
| 0.2 | 0.1 | 200 | 15 | 0 |
| 0.2 | 0.1 | 300 | 15 | 0 |
| 0.2 | 0.1 | 400 | 15 | 0 |
| 0.2 | 0.1 | 500 | 15 | 0 |
| 0.2 | 0.2 | 10 | 15 | 0 |
| 0.2 | 0.2 | 25 | 15 | 0 |
| 0.2 | 0.2 | 50 | 15 | 0 |
| 0.2 | 0.2 | 100 | 15 | 0 |
| 0.2 | 0.2 | 200 | 15 | 0 |
| 0.2 | 0.2 | 300 | 15 | 0 |
| 0.2 | 0.2 | 400 | 15 | 0 |
| 0.2 | 0.2 | 500 | 15 | 0 |
| 0.2 | 0.3 | 10 | 15 | 0 |
| 0.2 | 0.3 | 25 | 15 | 0 |
| 0.2 | 0.3 | 50 | 15 | 0 |
| 0.2 | 0.3 | 100 | 15 | 0 |
| 0.2 | 0.3 | 200 | 15 | 0 |
| 0.2 | 0.3 | 300 | 15 | 0 |
| 0.2 | 0.3 | 400 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.2 | 0.3 | 500 | 15 | 0 |
| 0.2 | 0.4 | 10 | 15 | 0 |
| 0.2 | 0.4 | 25 | 15 | 0 |
| 0.2 | 0.4 | 50 | 15 | 0 |
| 0.2 | 0.4 | 100 | 15 | 0 |
| 0.2 | 0.4 | 200 | 15 | 0 |
| 0.2 | 0.4 | 300 | 15 | 0 |
| 0.2 | 0.4 | 400 | 15 | 0 |
| 0.2 | 0.4 | 500 | 15 | 0 |
| 0.2 | 0.5 | 10 | 15 | 0 |
| 0.2 | 0.5 | 25 | 15 | 0 |
| 0.2 | 0.5 | 50 | 15 | 0 |
| 0.2 | 0.5 | 100 | 15 | 0 |
| 0.2 | 0.5 | 200 | 15 | 0 |
| 0.2 | 0.5 | 300 | 15 | 0 |
| 0.2 | 0.5 | 400 | 15 | 0 |
| 0.2 | 0.5 | 500 | 15 | 0 |
| 0.2 | 0.6 | 10 | 15 | 0 |
| 0.2 | 0.6 | 25 | 15 | 0 |
| 0.2 | 0.6 | 50 | 15 | 0 |
| 0.2 | 0.6 | 100 | 15 | 0 |
| 0.2 | 0.6 | 200 | 15 | 0 |
| 0.2 | 0.6 | 300 | 15 | 0 |
| 0.2 | 0.6 | 400 | 15 | 0 |
| 0.2 | 0.6 | 500 | 15 | 0 |
| 0.2 | 0.7 | 10 | 15 | 0 |
| 0.2 | 0.7 | 25 | 15 | 0 |
| 0.2 | 0.7 | 50 | 15 | 0 |
| 0.2 | 0.7 | 100 | 15 | 0 |
| 0.2 | 0.7 | 200 | 15 | 0 |
| 0.2 | 0.7 | 300 | 15 | 0 |
| 0.2 | 0.7 | 400 | 15 | 0 |
| 0.2 | 0.7 | 500 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.2 | 0.8 | 10 | 15 | 2 |
| 0.2 | 0.8 | 25 | 15 | 0 |
| 0.2 | 0.8 | 50 | 15 | 0 |
| 0.2 | 0.8 | 100 | 15 | 0 |
| 0.2 | 0.8 | 200 | 15 | 0 |
| 0.2 | 0.8 | 300 | 15 | 0 |
| 0.2 | 0.8 | 400 | 15 | 0 |
| 0.2 | 0.8 | 500 | 15 | 0 |
| 0.2 | 0.9 | 10 | 15 | 0 |
| 0.2 | 0.9 | 25 | 15 | 0 |
| 0.2 | 0.9 | 50 | 15 | 0 |
| 0.2 | 0.9 | 100 | 15 | 0 |
| 0.2 | 0.9 | 200 | 15 | 0 |
| 0.2 | 0.9 | 300 | 15 | 0 |
| 0.2 | 0.9 | 400 | 15 | 0 |
| 0.2 | 0.9 | 500 | 15 | 0 |
| 0.3 | 0.1 | 10 | 15 | 1 |
| 0.3 | 0.1 | 25 | 15 | 0 |
| 0.3 | 0.1 | 50 | 15 | 0 |
| 0.3 | 0.1 | 100 | 15 | 0 |
| 0.3 | 0.1 | 200 | 15 | 0 |
| 0.3 | 0.1 | 300 | 15 | 0 |
| 0.3 | 0.1 | 400 | 15 | 0 |
| 0.3 | 0.1 | 500 | 15 | 0 |
| 0.3 | 0.2 | 10 | 15 | 0 |
| 0.3 | 0.2 | 25 | 15 | 0 |
| 0.3 | 0.2 | 50 | 15 | 0 |
| 0.3 | 0.2 | 100 | 15 | 0 |
| 0.3 | 0.2 | 200 | 15 | 0 |
| 0.3 | 0.2 | 300 | 15 | 0 |
| 0.3 | 0.2 | 400 | 15 | 0 |
| 0.3 | 0.2 | 500 | 15 | 0 |
| 0.3 | 0.3 | 10 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.3 | 0.3 | 25 | 15 | 0 |
| 0.3 | 0.3 | 50 | 15 | 0 |
| 0.3 | 0.3 | 100 | 15 | 0 |
| 0.3 | 0.3 | 200 | 15 | 0 |
| 0.3 | 0.3 | 300 | 15 | 0 |
| 0.3 | 0.3 | 400 | 15 | 0 |
| 0.3 | 0.3 | 500 | 15 | 0 |
| 0.3 | 0.4 | 10 | 15 | 0 |
| 0.3 | 0.4 | 25 | 15 | 0 |
| 0.3 | 0.4 | 50 | 15 | 0 |
| 0.3 | 0.4 | 100 | 15 | 0 |
| 0.3 | 0.4 | 200 | 15 | 0 |
| 0.3 | 0.4 | 300 | 15 | 0 |
| 0.3 | 0.4 | 400 | 15 | 0 |
| 0.3 | 0.4 | 500 | 15 | 0 |
| 0.3 | 0.5 | 10 | 15 | 0 |
| 0.3 | 0.5 | 25 | 15 | 0 |
| 0.3 | 0.5 | 50 | 15 | 0 |
| 0.3 | 0.5 | 100 | 15 | 0 |
| 0.3 | 0.5 | 200 | 15 | 0 |
| 0.3 | 0.5 | 300 | 15 | 0 |
| 0.3 | 0.5 | 400 | 15 | 0 |
| 0.3 | 0.5 | 500 | 15 | 0 |
| 0.3 | 0.6 | 10 | 15 | 0 |
| 0.3 | 0.6 | 25 | 15 | 0 |
| 0.3 | 0.6 | 50 | 15 | 0 |
| 0.3 | 0.6 | 100 | 15 | 0 |
| 0.3 | 0.6 | 200 | 15 | 0 |
| 0.3 | 0.6 | 300 | 15 | 0 |
| 0.3 | 0.6 | 400 | 15 | 0 |
| 0.3 | 0.6 | 500 | 15 | 0 |
| 0.3 | 0.7 | 10 | 15 | 2 |
| 0.3 | 0.7 | 25 | 15 | 2 |

| | | | | |
|---|---|---|---|---|
| 0.3 | 0.7 | 50 | 15 | 0 |
| 0.3 | 0.7 | 100 | 15 | 0 |
| 0.3 | 0.7 | 200 | 15 | 0 |
| 0.3 | 0.7 | 300 | 15 | 0 |
| 0.3 | 0.7 | 400 | 15 | 0 |
| 0.3 | 0.7 | 500 | 15 | 0 |
| 0.3 | 0.8 | 10 | 15 | 2 |
| 0.3 | 0.8 | 25 | 15 | 0 |
| 0.3 | 0.8 | 50 | 15 | 0 |
| 0.3 | 0.8 | 100 | 15 | 0 |
| 0.3 | 0.8 | 200 | 15 | 0 |
| 0.3 | 0.8 | 300 | 15 | 0 |
| 0.3 | 0.8 | 400 | 15 | 0 |
| 0.3 | 0.8 | 500 | 15 | 0 |
| 0.3 | 0.9 | 10 | 15 | 0 |
| 0.3 | 0.9 | 25 | 15 | 0 |
| 0.3 | 0.9 | 50 | 15 | 0 |
| 0.3 | 0.9 | 100 | 15 | 0 |
| 0.3 | 0.9 | 200 | 15 | 0 |
| 0.3 | 0.9 | 300 | 15 | 0 |
| 0.3 | 0.9 | 400 | 15 | 0 |
| 0.3 | 0.9 | 500 | 15 | 0 |
| 0.4 | 0.1 | 10 | 15 | 3 |
| 0.4 | 0.1 | 25 | 15 | 0 |
| 0.4 | 0.1 | 50 | 15 | 0 |
| 0.4 | 0.1 | 100 | 15 | 0 |
| 0.4 | 0.1 | 200 | 15 | 0 |
| 0.4 | 0.1 | 300 | 15 | 0 |
| 0.4 | 0.1 | 400 | 15 | 0 |
| 0.4 | 0.1 | 500 | 15 | 0 |
| 0.4 | 0.2 | 10 | 15 | 2 |
| 0.4 | 0.2 | 25 | 15 | 0 |
| 0.4 | 0.2 | 50 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.4 | 0.2 | 100 | 15 | 0 |
| 0.4 | 0.2 | 200 | 15 | 0 |
| 0.4 | 0.2 | 300 | 15 | 0 |
| 0.4 | 0.2 | 400 | 15 | 0 |
| 0.4 | 0.2 | 500 | 15 | 0 |
| 0.4 | 0.3 | 10 | 15 | 2 |
| 0.4 | 0.3 | 25 | 15 | 0 |
| 0.4 | 0.3 | 50 | 15 | 0 |
| 0.4 | 0.3 | 100 | 15 | 0 |
| 0.4 | 0.3 | 200 | 15 | 0 |
| 0.4 | 0.3 | 300 | 15 | 0 |
| 0.4 | 0.3 | 400 | 15 | 0 |
| 0.4 | 0.3 | 500 | 15 | 0 |
| 0.4 | 0.4 | 10 | 15 | 0 |
| 0.4 | 0.4 | 25 | 15 | 0 |
| 0.4 | 0.4 | 50 | 15 | 0 |
| 0.4 | 0.4 | 100 | 15 | 0 |
| 0.4 | 0.4 | 200 | 15 | 0 |
| 0.4 | 0.4 | 300 | 15 | 0 |
| 0.4 | 0.4 | 400 | 15 | 0 |
| 0.4 | 0.4 | 500 | 15 | 0 |
| 0.4 | 0.5 | 10 | 15 | 0 |
| 0.4 | 0.5 | 25 | 15 | 0 |
| 0.4 | 0.5 | 50 | 15 | 0 |
| 0.4 | 0.5 | 100 | 15 | 0 |
| 0.4 | 0.5 | 200 | 15 | 0 |
| 0.4 | 0.5 | 300 | 15 | 0 |
| 0.4 | 0.5 | 400 | 15 | 0 |
| 0.4 | 0.5 | 500 | 15 | 0 |
| 0.4 | 0.6 | 10 | 15 | 0 |
| 0.4 | 0.6 | 25 | 15 | 0 |
| 0.4 | 0.6 | 50 | 15 | 0 |
| 0.4 | 0.6 | 100 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.4 | 0.6 | 200 | 15 | 0 |
| 0.4 | 0.6 | 300 | 15 | 0 |
| 0.4 | 0.6 | 400 | 15 | 0 |
| 0.4 | 0.6 | 500 | 15 | 0 |
| 0.4 | 0.7 | 10 | 15 | 2 |
| 0.4 | 0.7 | 25 | 15 | 0 |
| 0.4 | 0.7 | 50 | 15 | 0 |
| 0.4 | 0.7 | 100 | 15 | 0 |
| 0.4 | 0.7 | 200 | 15 | 0 |
| 0.4 | 0.7 | 300 | 15 | 0 |
| 0.4 | 0.7 | 400 | 15 | 0 |
| 0.4 | 0.7 | 500 | 15 | 0 |
| 0.4 | 0.8 | 10 | 15 | 0 |
| 0.4 | 0.8 | 25 | 15 | 0 |
| 0.4 | 0.8 | 50 | 15 | 0 |
| 0.4 | 0.8 | 100 | 15 | 0 |
| 0.4 | 0.8 | 200 | 15 | 0 |
| 0.4 | 0.8 | 300 | 15 | 0 |
| 0.4 | 0.8 | 400 | 15 | 0 |
| 0.4 | 0.8 | 500 | 15 | 0 |
| 0.4 | 0.9 | 10 | 15 | 0 |
| 0.4 | 0.9 | 25 | 15 | 0 |
| 0.4 | 0.9 | 50 | 15 | 0 |
| 0.4 | 0.9 | 100 | 15 | 0 |
| 0.4 | 0.9 | 200 | 15 | 0 |
| 0.4 | 0.9 | 300 | 15 | 0 |
| 0.4 | 0.9 | 400 | 15 | 0 |
| 0.4 | 0.9 | 500 | 15 | 0 |
| 0.5 | 0.1 | 10 | 15 | 0 |
| 0.5 | 0.1 | 25 | 15 | 0 |
| 0.5 | 0.1 | 50 | 15 | 0 |
| 0.5 | 0.1 | 100 | 15 | 0 |
| 0.5 | 0.1 | 200 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.1 | 300 | 15 | 0 |
| 0.5 | 0.1 | 400 | 15 | 0 |
| 0.5 | 0.1 | 500 | 15 | 0 |
| 0.5 | 0.2 | 10 | 15 | 1 |
| 0.5 | 0.2 | 25 | 15 | 0 |
| 0.5 | 0.2 | 50 | 15 | 0 |
| 0.5 | 0.2 | 100 | 15 | 0 |
| 0.5 | 0.2 | 200 | 15 | 0 |
| 0.5 | 0.2 | 300 | 15 | 0 |
| 0.5 | 0.2 | 400 | 15 | 0 |
| 0.5 | 0.2 | 500 | 15 | 0 |
| 0.5 | 0.3 | 10 | 15 | 2 |
| 0.5 | 0.3 | 25 | 15 | 2 |
| 0.5 | 0.3 | 50 | 15 | 0 |
| 0.5 | 0.3 | 100 | 15 | 0 |
| 0.5 | 0.3 | 200 | 15 | 0 |
| 0.5 | 0.3 | 300 | 15 | 0 |
| 0.5 | 0.3 | 400 | 15 | 0 |
| 0.5 | 0.3 | 500 | 15 | 0 |
| 0.5 | 0.4 | 10 | 15 | 2 |
| 0.5 | 0.4 | 25 | 15 | 0 |
| 0.5 | 0.4 | 50 | 15 | 0 |
| 0.5 | 0.4 | 100 | 15 | 0 |
| 0.5 | 0.4 | 200 | 15 | 0 |
| 0.5 | 0.4 | 300 | 15 | 0 |
| 0.5 | 0.4 | 400 | 15 | 0 |
| 0.5 | 0.4 | 500 | 15 | 0 |
| 0.5 | 0.5 | 10 | 15 | 0 |
| 0.5 | 0.5 | 25 | 15 | 0 |
| 0.5 | 0.5 | 50 | 15 | 0 |
| 0.5 | 0.5 | 100 | 15 | 0 |
| 0.5 | 0.5 | 200 | 15 | 0 |
| 0.5 | 0.5 | 300 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.5 | 400 | 15 | 0 |
| 0.5 | 0.5 | 500 | 15 | 0 |
| 0.5 | 0.6 | 10 | 15 | 0 |
| 0.5 | 0.6 | 25 | 15 | 0 |
| 0.5 | 0.6 | 50 | 15 | 0 |
| 0.5 | 0.6 | 100 | 15 | 0 |
| 0.5 | 0.6 | 200 | 15 | 0 |
| 0.5 | 0.6 | 300 | 15 | 0 |
| 0.5 | 0.6 | 400 | 15 | 0 |
| 0.5 | 0.6 | 500 | 15 | 0 |
| 0.5 | 0.7 | 10 | 15 | 0 |
| 0.5 | 0.7 | 25 | 15 | 0 |
| 0.5 | 0.7 | 50 | 15 | 0 |
| 0.5 | 0.7 | 100 | 15 | 0 |
| 0.5 | 0.7 | 200 | 15 | 0 |
| 0.5 | 0.7 | 300 | 15 | 0 |
| 0.5 | 0.7 | 400 | 15 | 0 |
| 0.5 | 0.7 | 500 | 15 | 0 |
| 0.5 | 0.8 | 10 | 15 | 0 |
| 0.5 | 0.8 | 25 | 15 | 0 |
| 0.5 | 0.8 | 50 | 15 | 0 |
| 0.5 | 0.8 | 100 | 15 | 0 |
| 0.5 | 0.8 | 200 | 15 | 0 |
| 0.5 | 0.8 | 300 | 15 | 0 |
| 0.5 | 0.8 | 400 | 15 | 0 |
| 0.5 | 0.8 | 500 | 15 | 0 |
| 0.5 | 0.9 | 10 | 15 | 0 |
| 0.5 | 0.9 | 25 | 15 | 0 |
| 0.5 | 0.9 | 50 | 15 | 0 |
| 0.5 | 0.9 | 100 | 15 | 0 |
| 0.5 | 0.9 | 200 | 15 | 0 |
| 0.5 | 0.9 | 300 | 15 | 0 |
| 0.5 | 0.9 | 400 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.9 | 500 | 15 | 0 |
| 0.6 | 0.1 | 10 | 15 | 3 |
| 0.6 | 0.1 | 25 | 15 | 0 |
| 0.6 | 0.1 | 50 | 15 | 0 |
| 0.6 | 0.1 | 100 | 15 | 0 |
| 0.6 | 0.1 | 200 | 15 | 0 |
| 0.6 | 0.1 | 300 | 15 | 0 |
| 0.6 | 0.1 | 400 | 15 | 0 |
| 0.6 | 0.1 | 500 | 15 | 0 |
| 0.6 | 0.2 | 10 | 15 | 1 |
| 0.6 | 0.2 | 25 | 15 | 0 |
| 0.6 | 0.2 | 50 | 15 | 1 |
| 0.6 | 0.2 | 100 | 15 | 0 |
| 0.6 | 0.2 | 200 | 15 | 0 |
| 0.6 | 0.2 | 300 | 15 | 0 |
| 0.6 | 0.2 | 400 | 15 | 0 |
| 0.6 | 0.2 | 500 | 15 | 0 |
| 0.6 | 0.3 | 10 | 15 | 3 |
| 0.6 | 0.3 | 25 | 15 | 0 |
| 0.6 | 0.3 | 50 | 15 | 0 |
| 0.6 | 0.3 | 100 | 15 | 0 |
| 0.6 | 0.3 | 200 | 15 | 0 |
| 0.6 | 0.3 | 300 | 15 | 0 |
| 0.6 | 0.3 | 400 | 15 | 0 |
| 0.6 | 0.3 | 500 | 15 | 0 |
| 0.6 | 0.4 | 10 | 15 | 0 |
| 0.6 | 0.4 | 25 | 15 | 0 |
| 0.6 | 0.4 | 50 | 15 | 0 |
| 0.6 | 0.4 | 100 | 15 | 0 |
| 0.6 | 0.4 | 200 | 15 | 0 |
| 0.6 | 0.4 | 300 | 15 | 0 |
| 0.6 | 0.4 | 400 | 15 | 0 |
| 0.6 | 0.4 | 500 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.6 | 0.5 | 10 | 15 | 1 |
| 0.6 | 0.5 | 25 | 15 | 0 |
| 0.6 | 0.5 | 50 | 15 | 0 |
| 0.6 | 0.5 | 100 | 15 | 0 |
| 0.6 | 0.5 | 200 | 15 | 0 |
| 0.6 | 0.5 | 300 | 15 | 0 |
| 0.6 | 0.5 | 400 | 15 | 0 |
| 0.6 | 0.5 | 500 | 15 | 0 |
| 0.6 | 0.6 | 10 | 15 | 1 |
| 0.6 | 0.6 | 25 | 15 | 0 |
| 0.6 | 0.6 | 50 | 15 | 0 |
| 0.6 | 0.6 | 100 | 15 | 0 |
| 0.6 | 0.6 | 200 | 15 | 0 |
| 0.6 | 0.6 | 300 | 15 | 0 |
| 0.6 | 0.6 | 400 | 15 | 0 |
| 0.6 | 0.6 | 500 | 15 | 0 |
| 0.6 | 0.7 | 10 | 15 | 0 |
| 0.6 | 0.7 | 25 | 15 | 0 |
| 0.6 | 0.7 | 50 | 15 | 0 |
| 0.6 | 0.7 | 100 | 15 | 0 |
| 0.6 | 0.7 | 200 | 15 | 0 |
| 0.6 | 0.7 | 300 | 15 | 0 |
| 0.6 | 0.7 | 400 | 15 | 0 |
| 0.6 | 0.7 | 500 | 15 | 0 |
| 0.6 | 0.8 | 10 | 15 | 1 |
| 0.6 | 0.8 | 25 | 15 | 0 |
| 0.6 | 0.8 | 50 | 15 | 0 |
| 0.6 | 0.8 | 100 | 15 | 0 |
| 0.6 | 0.8 | 200 | 15 | 0 |
| 0.6 | 0.8 | 300 | 15 | 0 |
| 0.6 | 0.8 | 400 | 15 | 0 |
| 0.6 | 0.8 | 500 | 15 | 0 |
| 0.6 | 0.9 | 10 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.6 | 0.9 | 25 | 15 | 0 |
| 0.6 | 0.9 | 50 | 15 | 0 |
| 0.6 | 0.9 | 100 | 15 | 0 |
| 0.6 | 0.9 | 200 | 15 | 0 |
| 0.6 | 0.9 | 300 | 15 | 0 |
| 0.6 | 0.9 | 400 | 15 | 0 |
| 0.6 | 0.9 | 500 | 15 | 0 |
| 0.7 | 0.1 | 10 | 15 | 2 |
| 0.7 | 0.1 | 25 | 15 | 0 |
| 0.7 | 0.1 | 50 | 15 | 0 |
| 0.7 | 0.1 | 100 | 15 | 0 |
| 0.7 | 0.1 | 200 | 15 | 0 |
| 0.7 | 0.1 | 300 | 15 | 0 |
| 0.7 | 0.1 | 400 | 15 | 0 |
| 0.7 | 0.1 | 500 | 15 | 0 |
| 0.7 | 0.2 | 10 | 15 | 2 |
| 0.7 | 0.2 | 25 | 15 | 0 |
| 0.7 | 0.2 | 50 | 15 | 0 |
| 0.7 | 0.2 | 100 | 15 | 0 |
| 0.7 | 0.2 | 200 | 15 | 0 |
| 0.7 | 0.2 | 300 | 15 | 0 |
| 0.7 | 0.2 | 400 | 15 | 0 |
| 0.7 | 0.2 | 500 | 15 | 0 |
| 0.7 | 0.3 | 10 | 15 | 3 |
| 0.7 | 0.3 | 25 | 15 | 1 |
| 0.7 | 0.3 | 50 | 15 | 0 |
| 0.7 | 0.3 | 100 | 15 | 0 |
| 0.7 | 0.3 | 200 | 15 | 0 |
| 0.7 | 0.3 | 300 | 15 | 0 |
| 0.7 | 0.3 | 400 | 15 | 0 |
| 0.7 | 0.3 | 500 | 15 | 0 |
| 0.7 | 0.4 | 10 | 15 | 0 |
| 0.7 | 0.4 | 25 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.7 | 0.4 | 50 | 15 | 0 |
| 0.7 | 0.4 | 100 | 15 | 0 |
| 0.7 | 0.4 | 200 | 15 | 0 |
| 0.7 | 0.4 | 300 | 15 | 0 |
| 0.7 | 0.4 | 400 | 15 | 0 |
| 0.7 | 0.4 | 500 | 15 | 0 |
| 0.7 | 0.5 | 10 | 15 | 0 |
| 0.7 | 0.5 | 25 | 15 | 0 |
| 0.7 | 0.5 | 50 | 15 | 0 |
| 0.7 | 0.5 | 100 | 15 | 0 |
| 0.7 | 0.5 | 200 | 15 | 0 |
| 0.7 | 0.5 | 300 | 15 | 0 |
| 0.7 | 0.5 | 400 | 15 | 0 |
| 0.7 | 0.5 | 500 | 15 | 0 |
| 0.7 | 0.6 | 10 | 15 | 0 |
| 0.7 | 0.6 | 25 | 15 | 0 |
| 0.7 | 0.6 | 50 | 15 | 0 |
| 0.7 | 0.6 | 100 | 15 | 0 |
| 0.7 | 0.6 | 200 | 15 | 0 |
| 0.7 | 0.6 | 300 | 15 | 0 |
| 0.7 | 0.6 | 400 | 15 | 0 |
| 0.7 | 0.6 | 500 | 15 | 0 |
| 0.7 | 0.7 | 10 | 15 | 0 |
| 0.7 | 0.7 | 25 | 15 | 0 |
| 0.7 | 0.7 | 50 | 15 | 0 |
| 0.7 | 0.7 | 100 | 15 | 0 |
| 0.7 | 0.7 | 200 | 15 | 0 |
| 0.7 | 0.7 | 300 | 15 | 0 |
| 0.7 | 0.7 | 400 | 15 | 0 |
| 0.7 | 0.7 | 500 | 15 | 0 |
| 0.7 | 0.8 | 10 | 15 | 0 |
| 0.7 | 0.8 | 25 | 15 | 0 |
| 0.7 | 0.8 | 50 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.7 | 0.8 | 100 | 15 | 0 |
| 0.7 | 0.8 | 200 | 15 | 0 |
| 0.7 | 0.8 | 300 | 15 | 0 |
| 0.7 | 0.8 | 400 | 15 | 0 |
| 0.7 | 0.8 | 500 | 15 | 0 |
| 0.7 | 0.9 | 10 | 15 | 2 |
| 0.7 | 0.9 | 25 | 15 | 0 |
| 0.7 | 0.9 | 50 | 15 | 0 |
| 0.7 | 0.9 | 100 | 15 | 0 |
| 0.7 | 0.9 | 200 | 15 | 0 |
| 0.7 | 0.9 | 300 | 15 | 0 |
| 0.7 | 0.9 | 400 | 15 | 0 |
| 0.7 | 0.9 | 500 | 15 | 0 |
| 0.8 | 0.1 | 10 | 15 | 4 |
| 0.8 | 0.1 | 25 | 15 | 2 |
| 0.8 | 0.1 | 50 | 15 | 0 |
| 0.8 | 0.1 | 100 | 15 | 0 |
| 0.8 | 0.1 | 200 | 15 | 0 |
| 0.8 | 0.1 | 300 | 15 | 0 |
| 0.8 | 0.1 | 400 | 15 | 0 |
| 0.8 | 0.1 | 500 | 15 | 0 |
| 0.8 | 0.2 | 10 | 15 | 1 |
| 0.8 | 0.2 | 25 | 15 | 0 |
| 0.8 | 0.2 | 50 | 15 | 0 |
| 0.8 | 0.2 | 100 | 15 | 0 |
| 0.8 | 0.2 | 200 | 15 | 0 |
| 0.8 | 0.2 | 300 | 15 | 0 |
| 0.8 | 0.2 | 400 | 15 | 0 |
| 0.8 | 0.2 | 500 | 15 | 0 |
| 0.8 | 0.3 | 10 | 15 | 2 |
| 0.8 | 0.3 | 25 | 15 | 0 |
| 0.8 | 0.3 | 50 | 15 | 0 |
| 0.8 | 0.3 | 100 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.8 | 0.3 | 200 | 15 | 0 |
| 0.8 | 0.3 | 300 | 15 | 0 |
| 0.8 | 0.3 | 400 | 15 | 0 |
| 0.8 | 0.3 | 500 | 15 | 0 |
| 0.8 | 0.4 | 10 | 15 | 3 |
| 0.8 | 0.4 | 25 | 15 | 1 |
| 0.8 | 0.4 | 50 | 15 | 0 |
| 0.8 | 0.4 | 100 | 15 | 0 |
| 0.8 | 0.4 | 200 | 15 | 0 |
| 0.8 | 0.4 | 300 | 15 | 0 |
| 0.8 | 0.4 | 400 | 15 | 0 |
| 0.8 | 0.4 | 500 | 15 | 0 |
| 0.8 | 0.5 | 10 | 15 | 4 |
| 0.8 | 0.5 | 25 | 15 | 1 |
| 0.8 | 0.5 | 50 | 15 | 0 |
| 0.8 | 0.5 | 100 | 15 | 0 |
| 0.8 | 0.5 | 200 | 15 | 0 |
| 0.8 | 0.5 | 300 | 15 | 0 |
| 0.8 | 0.5 | 400 | 15 | 0 |
| 0.8 | 0.5 | 500 | 15 | 0 |
| 0.8 | 0.6 | 10 | 15 | 1 |
| 0.8 | 0.6 | 25 | 15 | 0 |
| 0.8 | 0.6 | 50 | 15 | 0 |
| 0.8 | 0.6 | 100 | 15 | 0 |
| 0.8 | 0.6 | 200 | 15 | 0 |
| 0.8 | 0.6 | 300 | 15 | 0 |
| 0.8 | 0.6 | 400 | 15 | 0 |
| 0.8 | 0.6 | 500 | 15 | 0 |
| 0.8 | 0.7 | 10 | 15 | 0 |
| 0.8 | 0.7 | 25 | 15 | 1 |
| 0.8 | 0.7 | 50 | 15 | 0 |
| 0.8 | 0.7 | 100 | 15 | 0 |
| 0.8 | 0.7 | 200 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.8 | 0.7 | 300 | 15 | 0 |
| 0.8 | 0.7 | 400 | 15 | 0 |
| 0.8 | 0.7 | 500 | 15 | 0 |
| 0.8 | 0.8 | 10 | 15 | 0 |
| 0.8 | 0.8 | 25 | 15 | 1 |
| 0.8 | 0.8 | 50 | 15 | 0 |
| 0.8 | 0.8 | 100 | 15 | 0 |
| 0.8 | 0.8 | 200 | 15 | 0 |
| 0.8 | 0.8 | 300 | 15 | 0 |
| 0.8 | 0.8 | 400 | 15 | 0 |
| 0.8 | 0.8 | 500 | 15 | 0 |
| 0.8 | 0.9 | 10 | 15 | 0 |
| 0.8 | 0.9 | 25 | 15 | 0 |
| 0.8 | 0.9 | 50 | 15 | 1 |
| 0.8 | 0.9 | 100 | 15 | 0 |
| 0.8 | 0.9 | 200 | 15 | 0 |
| 0.8 | 0.9 | 300 | 15 | 0 |
| 0.8 | 0.9 | 400 | 15 | 0 |
| 0.8 | 0.9 | 500 | 15 | 0 |
| 0.9 | 0.1 | 10 | 15 | 5 |
| 0.9 | 0.1 | 25 | 15 | 2 |
| 0.9 | 0.1 | 50 | 15 | 0 |
| 0.9 | 0.1 | 100 | 15 | 0 |
| 0.9 | 0.1 | 200 | 15 | 0 |
| 0.9 | 0.1 | 300 | 15 | 0 |
| 0.9 | 0.1 | 400 | 15 | 0 |
| 0.9 | 0.1 | 500 | 15 | 0 |
| 0.9 | 0.2 | 10 | 15 | 1 |
| 0.9 | 0.2 | 25 | 15 | 1 |
| 0.9 | 0.2 | 50 | 15 | 1 |
| 0.9 | 0.2 | 100 | 15 | 0 |
| 0.9 | 0.2 | 200 | 15 | 0 |
| 0.9 | 0.2 | 300 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.9 | 0.2 | 400 | 15 | 0 |
| 0.9 | 0.2 | 500 | 15 | 0 |
| 0.9 | 0.3 | 10 | 15 | 2 |
| 0.9 | 0.3 | 25 | 15 | 0 |
| 0.9 | 0.3 | 50 | 15 | 0 |
| 0.9 | 0.3 | 100 | 15 | 0 |
| 0.9 | 0.3 | 200 | 15 | 0 |
| 0.9 | 0.3 | 300 | 15 | 0 |
| 0.9 | 0.3 | 400 | 15 | 0 |
| 0.9 | 0.3 | 500 | 15 | 0 |
| 0.9 | 0.4 | 10 | 15 | 2 |
| 0.9 | 0.4 | 25 | 15 | 2 |
| 0.9 | 0.4 | 50 | 15 | 0 |
| 0.9 | 0.4 | 100 | 15 | 0 |
| 0.9 | 0.4 | 200 | 15 | 0 |
| 0.9 | 0.4 | 300 | 15 | 0 |
| 0.9 | 0.4 | 400 | 15 | 0 |
| 0.9 | 0.4 | 500 | 15 | 0 |
| 0.9 | 0.5 | 10 | 15 | 3 |
| 0.9 | 0.5 | 25 | 15 | 0 |
| 0.9 | 0.5 | 50 | 15 | 0 |
| 0.9 | 0.5 | 100 | 15 | 0 |
| 0.9 | 0.5 | 200 | 15 | 0 |
| 0.9 | 0.5 | 300 | 15 | 0 |
| 0.9 | 0.5 | 400 | 15 | 0 |
| 0.9 | 0.5 | 500 | 15 | 0 |
| 0.9 | 0.6 | 10 | 15 | 2 |
| 0.9 | 0.6 | 25 | 15 | 0 |
| 0.9 | 0.6 | 50 | 15 | 0 |
| 0.9 | 0.6 | 100 | 15 | 0 |
| 0.9 | 0.6 | 200 | 15 | 0 |
| 0.9 | 0.6 | 300 | 15 | 0 |
| 0.9 | 0.6 | 400 | 15 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.9 | 0.6 | 500 | 15 | 0 |
| 0.9 | 0.7 | 10 | 15 | 2 |
| 0.9 | 0.7 | 25 | 15 | 2 |
| 0.9 | 0.7 | 50 | 15 | 1 |
| 0.9 | 0.7 | 100 | 15 | 0 |
| 0.9 | 0.7 | 200 | 15 | 0 |
| 0.9 | 0.7 | 300 | 15 | 0 |
| 0.9 | 0.7 | 400 | 15 | 0 |
| 0.9 | 0.7 | 500 | 15 | 0 |
| 0.9 | 0.8 | 10 | 15 | 0 |
| 0.9 | 0.8 | 25 | 15 | 0 |
| 0.9 | 0.8 | 50 | 15 | 1 |
| 0.9 | 0.8 | 100 | 15 | 0 |
| 0.9 | 0.8 | 200 | 15 | 0 |
| 0.9 | 0.8 | 300 | 15 | 0 |
| 0.9 | 0.8 | 400 | 15 | 0 |
| 0.9 | 0.8 | 500 | 15 | 0 |
| 0.9 | 0.9 | 10 | 15 | 2 |
| 0.9 | 0.9 | 25 | 15 | 2 |
| 0.9 | 0.9 | 50 | 15 | 0 |
| 0.9 | 0.9 | 100 | 15 | 0 |
| 0.9 | 0.9 | 200 | 15 | 0 |
| 0.9 | 0.9 | 300 | 15 | 0 |
| 0.9 | 0.9 | 400 | 15 | 0 |
| 0.9 | 0.9 | 500 | 15 | 0 |

# Приложение 2

Таблица 4.5 – Параметризация для класса данных 2

| $\alpha$ | $\rho$ | days | ideal | mistake |
|---|---|---|---|---|
| 0.1 | 0.1 | 25 | 47326 | 1141 |
| 0.1 | 0.1 | 50 | 47326 | 0 |
| 0.1 | 0.1 | 100 | 47326 | 0 |
| 0.1 | 0.1 | 200 | 47326 | 0 |
| 0.1 | 0.1 | 300 | 47326 | 0 |
| 0.1 | 0.1 | 400 | 47326 | 0 |
| 0.1 | 0.1 | 500 | 47326 | 0 |
| 0.1 | 0.2 | 10 | 47326 | 1573 |
| 0.1 | 0.2 | 25 | 47326 | 712 |
| 0.1 | 0.2 | 50 | 47326 | 294 |
| 0.1 | 0.2 | 100 | 47326 | 0 |
| 0.1 | 0.2 | 200 | 47326 | 712 |
| 0.1 | 0.2 | 300 | 47326 | 0 |
| 0.1 | 0.2 | 400 | 47326 | 0 |
| 0.1 | 0.2 | 500 | 47326 | 0 |
| 0.1 | 0.3 | 10 | 47326 | 2957 |
| 0.1 | 0.3 | 25 | 47326 | 294 |
| 0.1 | 0.3 | 50 | 47326 | 294 |
| 0.1 | 0.3 | 100 | 47326 | 294 |
| 0.1 | 0.3 | 200 | 47326 | 0 |
| 0.1 | 0.3 | 300 | 47326 | 0 |
| 0.1 | 0.3 | 400 | 47326 | 0 |
| 0.1 | 0.3 | 500 | 47326 | 0 |
| 0.1 | 0.4 | 10 | 47326 | 721 |
| 0.1 | 0.4 | 25 | 47326 | 721 |
| 0.1 | 0.4 | 50 | 47326 | 0 |
| 0.1 | 0.4 | 100 | 47326 | 0 |
| 0.1 | 0.4 | 200 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.1 | 0.4 | 300 | 47326 | 0 |
| 0.1 | 0.4 | 400 | 47326 | 0 |
| 0.1 | 0.4 | 500 | 47326 | 0 |
| 0.1 | 0.5 | 10 | 47326 | 2418 |
| 0.1 | 0.5 | 25 | 47326 | 1360 |
| 0.1 | 0.5 | 50 | 47326 | 0 |
| 0.1 | 0.5 | 100 | 47326 | 294 |
| 0.1 | 0.5 | 200 | 47326 | 0 |
| 0.1 | 0.5 | 300 | 47326 | 0 |
| 0.1 | 0.5 | 400 | 47326 | 0 |
| 0.1 | 0.5 | 500 | 47326 | 0 |
| 0.1 | 0.6 | 10 | 47326 | 1975 |
| 0.1 | 0.6 | 25 | 47326 | 883 |
| 0.1 | 0.6 | 50 | 47326 | 0 |
| 0.1 | 0.6 | 100 | 47326 | 0 |
| 0.1 | 0.6 | 200 | 47326 | 712 |
| 0.1 | 0.6 | 300 | 47326 | 0 |
| 0.1 | 0.6 | 400 | 47326 | 0 |
| 0.1 | 0.6 | 500 | 47326 | 0 |
| 0.1 | 0.7 | 10 | 47326 | 1330 |
| 0.1 | 0.7 | 25 | 47326 | 294 |
| 0.1 | 0.7 | 50 | 47326 | 294 |
| 0.1 | 0.7 | 100 | 47326 | 0 |
| 0.1 | 0.7 | 200 | 47326 | 0 |
| 0.1 | 0.7 | 300 | 47326 | 0 |
| 0.1 | 0.7 | 400 | 47326 | 0 |
| 0.1 | 0.7 | 500 | 47326 | 0 |
| 0.1 | 0.8 | 10 | 47326 | 721 |
| 0.1 | 0.8 | 25 | 47326 | 294 |
| 0.1 | 0.8 | 50 | 47326 | 294 |
| 0.1 | 0.8 | 100 | 47326 | 294 |
| 0.1 | 0.8 | 200 | 47326 | 0 |
| 0.1 | 0.8 | 300 | 47326 | 294 |

| | | | | |
|---|---|---|---|---|
| 0.1 | 0.8 | 400 | 47326 | 0 |
| 0.1 | 0.8 | 500 | 47326 | 294 |
| 0.1 | 0.9 | 10 | 47326 | 1360 |
| 0.1 | 0.9 | 25 | 47326 | 712 |
| 0.1 | 0.9 | 50 | 47326 | 0 |
| 0.1 | 0.9 | 100 | 47326 | 827 |
| 0.1 | 0.9 | 200 | 47326 | 0 |
| 0.1 | 0.9 | 300 | 47326 | 0 |
| 0.1 | 0.9 | 400 | 47326 | 0 |
| 0.1 | 0.9 | 500 | 47326 | 0 |
| 0.2 | 0.1 | 10 | 47326 | 1145 |
| 0.2 | 0.1 | 25 | 47326 | 1145 |
| 0.2 | 0.1 | 50 | 47326 | 0 |
| 0.2 | 0.1 | 100 | 47326 | 712 |
| 0.2 | 0.1 | 200 | 47326 | 0 |
| 0.2 | 0.1 | 300 | 47326 | 0 |
| 0.2 | 0.1 | 400 | 47326 | 0 |
| 0.2 | 0.1 | 500 | 47326 | 0 |
| 0.2 | 0.2 | 10 | 47326 | 1105 |
| 0.2 | 0.2 | 25 | 47326 | 1088 |
| 0.2 | 0.2 | 50 | 47326 | 1088 |
| 0.2 | 0.2 | 100 | 47326 | 294 |
| 0.2 | 0.2 | 200 | 47326 | 0 |
| 0.2 | 0.2 | 300 | 47326 | 0 |
| 0.2 | 0.2 | 400 | 47326 | 0 |
| 0.2 | 0.2 | 500 | 47326 | 0 |
| 0.2 | 0.3 | 10 | 47326 | 1360 |
| 0.2 | 0.3 | 25 | 47326 | 1088 |
| 0.2 | 0.3 | 50 | 47326 | 0 |
| 0.2 | 0.3 | 100 | 47326 | 0 |
| 0.2 | 0.3 | 200 | 47326 | 0 |
| 0.2 | 0.3 | 300 | 47326 | 0 |
| 0.2 | 0.3 | 400 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.2 | 0.3 | 500 | 47326 | 0 |
| 0.2 | 0.4 | 10 | 47326 | 1591 |
| 0.2 | 0.4 | 25 | 47326 | 712 |
| 0.2 | 0.4 | 50 | 47326 | 827 |
| 0.2 | 0.4 | 100 | 47326 | 712 |
| 0.2 | 0.4 | 200 | 47326 | 0 |
| 0.2 | 0.4 | 300 | 47326 | 0 |
| 0.2 | 0.4 | 400 | 47326 | 294 |
| 0.2 | 0.4 | 500 | 47326 | 0 |
| 0.2 | 0.5 | 10 | 47326 | 2269 |
| 0.2 | 0.5 | 25 | 47326 | 0 |
| 0.2 | 0.5 | 50 | 47326 | 0 |
| 0.2 | 0.5 | 100 | 47326 | 721 |
| 0.2 | 0.5 | 200 | 47326 | 0 |
| 0.2 | 0.5 | 300 | 47326 | 0 |
| 0.2 | 0.5 | 400 | 47326 | 0 |
| 0.2 | 0.5 | 500 | 47326 | 0 |
| 0.2 | 0.6 | 10 | 47326 | 1145 |
| 0.2 | 0.6 | 25 | 47326 | 1359 |
| 0.2 | 0.6 | 50 | 47326 | 294 |
| 0.2 | 0.6 | 100 | 47326 | 294 |
| 0.2 | 0.6 | 200 | 47326 | 0 |
| 0.2 | 0.6 | 300 | 47326 | 294 |
| 0.2 | 0.6 | 400 | 47326 | 0 |
| 0.2 | 0.6 | 500 | 47326 | 0 |
| 0.2 | 0.7 | 10 | 47326 | 1591 |
| 0.2 | 0.7 | 25 | 47326 | 1088 |
| 0.2 | 0.7 | 50 | 47326 | 712 |
| 0.2 | 0.7 | 100 | 47326 | 0 |
| 0.2 | 0.7 | 200 | 47326 | 0 |
| 0.2 | 0.7 | 300 | 47326 | 0 |
| 0.2 | 0.7 | 400 | 47326 | 0 |
| 0.2 | 0.7 | 500 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.2 | 0.8 | 10 | 47326 | 712 |
| 0.2 | 0.8 | 25 | 47326 | 1088 |
| 0.2 | 0.8 | 50 | 47326 | 721 |
| 0.2 | 0.8 | 100 | 47326 | 294 |
| 0.2 | 0.8 | 200 | 47326 | 712 |
| 0.2 | 0.8 | 300 | 47326 | 0 |
| 0.2 | 0.8 | 400 | 47326 | 0 |
| 0.2 | 0.8 | 500 | 47326 | 0 |
| 0.2 | 0.9 | 10 | 47326 | 1591 |
| 0.2 | 0.9 | 25 | 47326 | 2229 |
| 0.2 | 0.9 | 50 | 47326 | 294 |
| 0.2 | 0.9 | 100 | 47326 | 0 |
| 0.2 | 0.9 | 200 | 47326 | 0 |
| 0.2 | 0.9 | 300 | 47326 | 0 |
| 0.2 | 0.9 | 400 | 47326 | 0 |
| 0.2 | 0.9 | 500 | 47326 | 0 |
| 0.3 | 0.1 | 10 | 47326 | 883 |
| 0.3 | 0.1 | 25 | 47326 | 1088 |
| 0.3 | 0.1 | 50 | 47326 | 294 |
| 0.3 | 0.1 | 100 | 47326 | 294 |
| 0.3 | 0.1 | 200 | 47326 | 0 |
| 0.3 | 0.1 | 300 | 47326 | 0 |
| 0.3 | 0.1 | 400 | 47326 | 0 |
| 0.3 | 0.1 | 500 | 47326 | 0 |
| 0.3 | 0.2 | 10 | 47326 | 1105 |
| 0.3 | 0.2 | 25 | 47326 | 0 |
| 0.3 | 0.2 | 50 | 47326 | 827 |
| 0.3 | 0.2 | 100 | 47326 | 0 |
| 0.3 | 0.2 | 200 | 47326 | 0 |
| 0.3 | 0.2 | 300 | 47326 | 0 |
| 0.3 | 0.2 | 400 | 47326 | 0 |
| 0.3 | 0.2 | 500 | 47326 | 0 |
| 0.3 | 0.3 | 10 | 47326 | 712 |

| | | | | |
|---|---|---|---|---|
| 0.3 | 0.3 | 25 | 47326 | 1634 |
| 0.3 | 0.3 | 50 | 47326 | 721 |
| 0.3 | 0.3 | 100 | 47326 | 294 |
| 0.3 | 0.3 | 200 | 47326 | 0 |
| 0.3 | 0.3 | 300 | 47326 | 0 |
| 0.3 | 0.3 | 400 | 47326 | 0 |
| 0.3 | 0.3 | 500 | 47326 | 0 |
| 0.3 | 0.4 | 10 | 47326 | 1840 |
| 0.3 | 0.4 | 25 | 47326 | 1105 |
| 0.3 | 0.4 | 50 | 47326 | 0 |
| 0.3 | 0.4 | 100 | 47326 | 712 |
| 0.3 | 0.4 | 200 | 47326 | 0 |
| 0.3 | 0.4 | 300 | 47326 | 0 |
| 0.3 | 0.4 | 400 | 47326 | 0 |
| 0.3 | 0.4 | 500 | 47326 | 0 |
| 0.3 | 0.5 | 10 | 47326 | 1439 |
| 0.3 | 0.5 | 25 | 47326 | 883 |
| 0.3 | 0.5 | 50 | 47326 | 0 |
| 0.3 | 0.5 | 100 | 47326 | 712 |
| 0.3 | 0.5 | 200 | 47326 | 0 |
| 0.3 | 0.5 | 300 | 47326 | 0 |
| 0.3 | 0.5 | 400 | 47326 | 0 |
| 0.3 | 0.5 | 500 | 47326 | 0 |
| 0.3 | 0.6 | 10 | 47326 | 1006 |
| 0.3 | 0.6 | 25 | 47326 | 1359 |
| 0.3 | 0.6 | 50 | 47326 | 721 |
| 0.3 | 0.6 | 100 | 47326 | 0 |
| 0.3 | 0.6 | 200 | 47326 | 0 |
| 0.3 | 0.6 | 300 | 47326 | 0 |
| 0.3 | 0.6 | 400 | 47326 | 0 |
| 0.3 | 0.6 | 500 | 47326 | 0 |
| 0.3 | 0.7 | 10 | 47326 | 2098 |
| 0.3 | 0.7 | 25 | 47326 | 1141 |

| | | | | |
|---|---|---|---|---|
| 0.3 | 0.7 | 50 | 47326 | 827 |
| 0.3 | 0.7 | 100 | 47326 | 0 |
| 0.3 | 0.7 | 200 | 47326 | 294 |
| 0.3 | 0.7 | 300 | 47326 | 0 |
| 0.3 | 0.7 | 400 | 47326 | 0 |
| 0.3 | 0.7 | 500 | 47326 | 0 |
| 0.3 | 0.8 | 10 | 47326 | 1360 |
| 0.3 | 0.8 | 25 | 47326 | 1359 |
| 0.3 | 0.8 | 50 | 47326 | 1141 |
| 0.3 | 0.8 | 100 | 47326 | 294 |
| 0.3 | 0.8 | 200 | 47326 | 0 |
| 0.3 | 0.8 | 300 | 47326 | 0 |
| 0.3 | 0.8 | 400 | 47326 | 0 |
| 0.3 | 0.8 | 500 | 47326 | 0 |
| 0.3 | 0.9 | 10 | 47326 | 2402 |
| 0.3 | 0.9 | 25 | 47326 | 294 |
| 0.3 | 0.9 | 50 | 47326 | 712 |
| 0.3 | 0.9 | 100 | 47326 | 0 |
| 0.3 | 0.9 | 200 | 47326 | 0 |
| 0.3 | 0.9 | 300 | 47326 | 0 |
| 0.3 | 0.9 | 400 | 47326 | 0 |
| 0.3 | 0.9 | 500 | 47326 | 0 |
| 0.4 | 0.1 | 10 | 47326 | 2260 |
| 0.4 | 0.1 | 25 | 47326 | 1088 |
| 0.4 | 0.1 | 50 | 47326 | 827 |
| 0.4 | 0.1 | 100 | 47326 | 0 |
| 0.4 | 0.1 | 200 | 47326 | 0 |
| 0.4 | 0.1 | 300 | 47326 | 0 |
| 0.4 | 0.1 | 400 | 47326 | 0 |
| 0.4 | 0.1 | 500 | 47326 | 0 |
| 0.4 | 0.2 | 10 | 47326 | 2229 |
| 0.4 | 0.2 | 25 | 47326 | 294 |
| 0.4 | 0.2 | 50 | 47326 | 827 |

| | | | | |
|---|---|---|---|---|
| 0.4 | 0.2 | 100 | 47326 | 294 |
| 0.4 | 0.2 | 200 | 47326 | 0 |
| 0.4 | 0.2 | 300 | 47326 | 294 |
| 0.4 | 0.2 | 400 | 47326 | 0 |
| 0.4 | 0.2 | 500 | 47326 | 0 |
| 0.4 | 0.3 | 10 | 47326 | 294 |
| 0.4 | 0.3 | 25 | 47326 | 1439 |
| 0.4 | 0.3 | 50 | 47326 | 0 |
| 0.4 | 0.3 | 100 | 47326 | 0 |
| 0.4 | 0.3 | 200 | 47326 | 0 |
| 0.4 | 0.3 | 300 | 47326 | 294 |
| 0.4 | 0.3 | 400 | 47326 | 0 |
| 0.4 | 0.3 | 500 | 47326 | 0 |
| 0.4 | 0.4 | 10 | 47326 | 3728 |
| 0.4 | 0.4 | 25 | 47326 | 0 |
| 0.4 | 0.4 | 50 | 47326 | 1105 |
| 0.4 | 0.4 | 100 | 47326 | 294 |
| 0.4 | 0.4 | 200 | 47326 | 0 |
| 0.4 | 0.4 | 300 | 47326 | 0 |
| 0.4 | 0.4 | 400 | 47326 | 0 |
| 0.4 | 0.4 | 500 | 47326 | 0 |
| 0.4 | 0.5 | 10 | 47326 | 2229 |
| 0.4 | 0.5 | 25 | 47326 | 827 |
| 0.4 | 0.5 | 50 | 47326 | 712 |
| 0.4 | 0.5 | 100 | 47326 | 0 |
| 0.4 | 0.5 | 200 | 47326 | 712 |
| 0.4 | 0.5 | 300 | 47326 | 0 |
| 0.4 | 0.5 | 400 | 47326 | 0 |
| 0.4 | 0.5 | 500 | 47326 | 0 |
| 0.4 | 0.6 | 10 | 47326 | 1145 |
| 0.4 | 0.6 | 25 | 47326 | 0 |
| 0.4 | 0.6 | 50 | 47326 | 712 |
| 0.4 | 0.6 | 100 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.4 | 0.6 | 200 | 47326 | 0 |
| 0.4 | 0.6 | 300 | 47326 | 0 |
| 0.4 | 0.6 | 400 | 47326 | 0 |
| 0.4 | 0.6 | 500 | 47326 | 0 |
| 0.4 | 0.7 | 10 | 47326 | 1840 |
| 0.4 | 0.7 | 25 | 47326 | 294 |
| 0.4 | 0.7 | 50 | 47326 | 0 |
| 0.4 | 0.7 | 100 | 47326 | 0 |
| 0.4 | 0.7 | 200 | 47326 | 0 |
| 0.4 | 0.7 | 300 | 47326 | 0 |
| 0.4 | 0.7 | 400 | 47326 | 0 |
| 0.4 | 0.7 | 500 | 47326 | 0 |
| 0.4 | 0.8 | 10 | 47326 | 1634 |
| 0.4 | 0.8 | 25 | 47326 | 1088 |
| 0.4 | 0.8 | 50 | 47326 | 0 |
| 0.4 | 0.8 | 100 | 47326 | 0 |
| 0.4 | 0.8 | 200 | 47326 | 0 |
| 0.4 | 0.8 | 300 | 47326 | 0 |
| 0.4 | 0.8 | 400 | 47326 | 0 |
| 0.4 | 0.8 | 500 | 47326 | 0 |
| 0.4 | 0.9 | 10 | 47326 | 827 |
| 0.4 | 0.9 | 25 | 47326 | 827 |
| 0.4 | 0.9 | 50 | 47326 | 721 |
| 0.4 | 0.9 | 100 | 47326 | 0 |
| 0.4 | 0.9 | 200 | 47326 | 0 |
| 0.4 | 0.9 | 300 | 47326 | 0 |
| 0.4 | 0.9 | 400 | 47326 | 0 |
| 0.4 | 0.9 | 500 | 47326 | 0 |
| 0.5 | 0.1 | 10 | 47326 | 294 |
| 0.5 | 0.1 | 25 | 47326 | 0 |
| 0.5 | 0.1 | 50 | 47326 | 712 |
| 0.5 | 0.1 | 100 | 47326 | 294 |
| 0.5 | 0.1 | 200 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.1 | 300 | 47326 | 0 |
| 0.5 | 0.1 | 400 | 47326 | 0 |
| 0.5 | 0.1 | 500 | 47326 | 0 |
| 0.5 | 0.2 | 10 | 47326 | 2556 |
| 0.5 | 0.2 | 25 | 47326 | 0 |
| 0.5 | 0.2 | 50 | 47326 | 0 |
| 0.5 | 0.2 | 100 | 47326 | 0 |
| 0.5 | 0.2 | 200 | 47326 | 0 |
| 0.5 | 0.2 | 300 | 47326 | 0 |
| 0.5 | 0.2 | 400 | 47326 | 0 |
| 0.5 | 0.2 | 500 | 47326 | 0 |
| 0.5 | 0.3 | 10 | 47326 | 294 |
| 0.5 | 0.3 | 25 | 47326 | 1761 |
| 0.5 | 0.3 | 50 | 47326 | 294 |
| 0.5 | 0.3 | 100 | 47326 | 294 |
| 0.5 | 0.3 | 200 | 47326 | 0 |
| 0.5 | 0.3 | 300 | 47326 | 0 |
| 0.5 | 0.3 | 400 | 47326 | 0 |
| 0.5 | 0.3 | 500 | 47326 | 0 |
| 0.5 | 0.4 | 10 | 47326 | 721 |
| 0.5 | 0.4 | 25 | 47326 | 0 |
| 0.5 | 0.4 | 50 | 47326 | 294 |
| 0.5 | 0.4 | 100 | 47326 | 0 |
| 0.5 | 0.4 | 200 | 47326 | 294 |
| 0.5 | 0.4 | 300 | 47326 | 0 |
| 0.5 | 0.4 | 400 | 47326 | 0 |
| 0.5 | 0.4 | 500 | 47326 | 0 |
| 0.5 | 0.5 | 10 | 47326 | 294 |
| 0.5 | 0.5 | 25 | 47326 | 1105 |
| 0.5 | 0.5 | 50 | 47326 | 721 |
| 0.5 | 0.5 | 100 | 47326 | 294 |
| 0.5 | 0.5 | 200 | 47326 | 294 |
| 0.5 | 0.5 | 300 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.5 | 400 | 47326 | 0 |
| 0.5 | 0.5 | 500 | 47326 | 0 |
| 0.5 | 0.6 | 10 | 47326 | 883 |
| 0.5 | 0.6 | 25 | 47326 | 1088 |
| 0.5 | 0.6 | 50 | 47326 | 712 |
| 0.5 | 0.6 | 100 | 47326 | 0 |
| 0.5 | 0.6 | 200 | 47326 | 0 |
| 0.5 | 0.6 | 300 | 47326 | 0 |
| 0.5 | 0.6 | 400 | 47326 | 0 |
| 0.5 | 0.6 | 500 | 47326 | 0 |
| 0.5 | 0.7 | 10 | 47326 | 2402 |
| 0.5 | 0.7 | 25 | 47326 | 294 |
| 0.5 | 0.7 | 50 | 47326 | 883 |
| 0.5 | 0.7 | 100 | 47326 | 0 |
| 0.5 | 0.7 | 200 | 47326 | 0 |
| 0.5 | 0.7 | 300 | 47326 | 0 |
| 0.5 | 0.7 | 400 | 47326 | 0 |
| 0.5 | 0.7 | 500 | 47326 | 0 |
| 0.5 | 0.8 | 10 | 47326 | 1402 |
| 0.5 | 0.8 | 25 | 47326 | 0 |
| 0.5 | 0.8 | 50 | 47326 | 712 |
| 0.5 | 0.8 | 100 | 47326 | 0 |
| 0.5 | 0.8 | 200 | 47326 | 0 |
| 0.5 | 0.8 | 300 | 47326 | 0 |
| 0.5 | 0.8 | 400 | 47326 | 0 |
| 0.5 | 0.8 | 500 | 47326 | 0 |
| 0.5 | 0.9 | 10 | 47326 | 1088 |
| 0.5 | 0.9 | 25 | 47326 | 712 |
| 0.5 | 0.9 | 50 | 47326 | 827 |
| 0.5 | 0.9 | 100 | 47326 | 0 |
| 0.5 | 0.9 | 200 | 47326 | 0 |
| 0.5 | 0.9 | 300 | 47326 | 0 |
| 0.5 | 0.9 | 400 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.5 | 0.9 | 500 | 47326 | 0 |
| 0.6 | 0.1 | 10 | 47326 | 2418 |
| 0.6 | 0.1 | 25 | 47326 | 1402 |
| 0.6 | 0.1 | 50 | 47326 | 294 |
| 0.6 | 0.1 | 100 | 47326 | 0 |
| 0.6 | 0.1 | 200 | 47326 | 0 |
| 0.6 | 0.1 | 300 | 47326 | 0 |
| 0.6 | 0.1 | 400 | 47326 | 0 |
| 0.6 | 0.1 | 500 | 47326 | 0 |
| 0.6 | 0.2 | 10 | 47326 | 2238 |
| 0.6 | 0.2 | 25 | 47326 | 827 |
| 0.6 | 0.2 | 50 | 47326 | 1330 |
| 0.6 | 0.2 | 100 | 47326 | 712 |
| 0.6 | 0.2 | 200 | 47326 | 0 |
| 0.6 | 0.2 | 300 | 47326 | 294 |
| 0.6 | 0.2 | 400 | 47326 | 0 |
| 0.6 | 0.2 | 500 | 47326 | 0 |
| 0.6 | 0.3 | 10 | 47326 | 1105 |
| 0.6 | 0.3 | 25 | 47326 | 827 |
| 0.6 | 0.3 | 50 | 47326 | 883 |
| 0.6 | 0.3 | 100 | 47326 | 0 |
| 0.6 | 0.3 | 200 | 47326 | 0 |
| 0.6 | 0.3 | 300 | 47326 | 0 |
| 0.6 | 0.3 | 400 | 47326 | 0 |
| 0.6 | 0.3 | 500 | 47326 | 0 |
| 0.6 | 0.4 | 10 | 47326 | 1591 |
| 0.6 | 0.4 | 25 | 47326 | 1006 |
| 0.6 | 0.4 | 50 | 47326 | 0 |
| 0.6 | 0.4 | 100 | 47326 | 0 |
| 0.6 | 0.4 | 200 | 47326 | 0 |
| 0.6 | 0.4 | 300 | 47326 | 294 |
| 0.6 | 0.4 | 400 | 47326 | 0 |
| 0.6 | 0.4 | 500 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.6 | 0.5 | 10 | 47326 | 294 |
| 0.6 | 0.5 | 25 | 47326 | 1006 |
| 0.6 | 0.5 | 50 | 47326 | 0 |
| 0.6 | 0.5 | 100 | 47326 | 712 |
| 0.6 | 0.5 | 200 | 47326 | 294 |
| 0.6 | 0.5 | 300 | 47326 | 0 |
| 0.6 | 0.5 | 400 | 47326 | 0 |
| 0.6 | 0.5 | 500 | 47326 | 0 |
| 0.6 | 0.6 | 10 | 47326 | 2063 |
| 0.6 | 0.6 | 25 | 47326 | 1006 |
| 0.6 | 0.6 | 50 | 47326 | 294 |
| 0.6 | 0.6 | 100 | 47326 | 0 |
| 0.6 | 0.6 | 200 | 47326 | 721 |
| 0.6 | 0.6 | 300 | 47326 | 0 |
| 0.6 | 0.6 | 400 | 47326 | 0 |
| 0.6 | 0.6 | 500 | 47326 | 0 |
| 0.6 | 0.7 | 10 | 47326 | 712 |
| 0.6 | 0.7 | 25 | 47326 | 1105 |
| 0.6 | 0.7 | 50 | 47326 | 712 |
| 0.6 | 0.7 | 100 | 47326 | 0 |
| 0.6 | 0.7 | 200 | 47326 | 0 |
| 0.6 | 0.7 | 300 | 47326 | 0 |
| 0.6 | 0.7 | 400 | 47326 | 0 |
| 0.6 | 0.7 | 500 | 47326 | 0 |
| 0.6 | 0.8 | 10 | 47326 | 721 |
| 0.6 | 0.8 | 25 | 47326 | 1145 |
| 0.6 | 0.8 | 50 | 47326 | 721 |
| 0.6 | 0.8 | 100 | 47326 | 0 |
| 0.6 | 0.8 | 200 | 47326 | 0 |
| 0.6 | 0.8 | 300 | 47326 | 0 |
| 0.6 | 0.8 | 400 | 47326 | 0 |
| 0.6 | 0.8 | 500 | 47326 | 0 |
| 0.6 | 0.9 | 10 | 47326 | 2726 |

| | | | | |
|---|---|---|---|---|
| 0.6 | 0.9 | 25 | 47326 | 1840 |
| 0.6 | 0.9 | 50 | 47326 | 0 |
| 0.6 | 0.9 | 100 | 47326 | 0 |
| 0.6 | 0.9 | 200 | 47326 | 0 |
| 0.6 | 0.9 | 300 | 47326 | 0 |
| 0.6 | 0.9 | 400 | 47326 | 0 |
| 0.6 | 0.9 | 500 | 47326 | 0 |
| 0.7 | 0.1 | 10 | 47326 | 4282 |
| 0.7 | 0.1 | 25 | 47326 | 1088 |
| 0.7 | 0.1 | 50 | 47326 | 1141 |
| 0.7 | 0.1 | 100 | 47326 | 712 |
| 0.7 | 0.1 | 200 | 47326 | 0 |
| 0.7 | 0.1 | 300 | 47326 | 0 |
| 0.7 | 0.1 | 400 | 47326 | 0 |
| 0.7 | 0.1 | 500 | 47326 | 0 |
| 0.7 | 0.2 | 10 | 47326 | 2482 |
| 0.7 | 0.2 | 25 | 47326 | 0 |
| 0.7 | 0.2 | 50 | 47326 | 1145 |
| 0.7 | 0.2 | 100 | 47326 | 712 |
| 0.7 | 0.2 | 200 | 47326 | 721 |
| 0.7 | 0.2 | 300 | 47326 | 0 |
| 0.7 | 0.2 | 400 | 47326 | 0 |
| 0.7 | 0.2 | 500 | 47326 | 0 |
| 0.7 | 0.3 | 10 | 47326 | 1359 |
| 0.7 | 0.3 | 25 | 47326 | 712 |
| 0.7 | 0.3 | 50 | 47326 | 294 |
| 0.7 | 0.3 | 100 | 47326 | 712 |
| 0.7 | 0.3 | 200 | 47326 | 294 |
| 0.7 | 0.3 | 300 | 47326 | 712 |
| 0.7 | 0.3 | 400 | 47326 | 0 |
| 0.7 | 0.3 | 500 | 47326 | 0 |
| 0.7 | 0.4 | 10 | 47326 | 1359 |
| 0.7 | 0.4 | 25 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.7 | 0.4 | 50 | 47326 | 294 |
| 0.7 | 0.4 | 100 | 47326 | 294 |
| 0.7 | 0.4 | 200 | 47326 | 0 |
| 0.7 | 0.4 | 300 | 47326 | 827 |
| 0.7 | 0.4 | 400 | 47326 | 0 |
| 0.7 | 0.4 | 500 | 47326 | 0 |
| 0.7 | 0.5 | 10 | 47326 | 2309 |
| 0.7 | 0.5 | 25 | 47326 | 1105 |
| 0.7 | 0.5 | 50 | 47326 | 883 |
| 0.7 | 0.5 | 100 | 47326 | 294 |
| 0.7 | 0.5 | 200 | 47326 | 0 |
| 0.7 | 0.5 | 300 | 47326 | 0 |
| 0.7 | 0.5 | 400 | 47326 | 0 |
| 0.7 | 0.5 | 500 | 47326 | 0 |
| 0.7 | 0.6 | 10 | 47326 | 294 |
| 0.7 | 0.6 | 25 | 47326 | 1359 |
| 0.7 | 0.6 | 50 | 47326 | 294 |
| 0.7 | 0.6 | 100 | 47326 | 294 |
| 0.7 | 0.6 | 200 | 47326 | 0 |
| 0.7 | 0.6 | 300 | 47326 | 0 |
| 0.7 | 0.6 | 400 | 47326 | 0 |
| 0.7 | 0.6 | 500 | 47326 | 0 |
| 0.7 | 0.7 | 10 | 47326 | 2014 |
| 0.7 | 0.7 | 25 | 47326 | 1006 |
| 0.7 | 0.7 | 50 | 47326 | 712 |
| 0.7 | 0.7 | 100 | 47326 | 0 |
| 0.7 | 0.7 | 200 | 47326 | 712 |
| 0.7 | 0.7 | 300 | 47326 | 0 |
| 0.7 | 0.7 | 400 | 47326 | 0 |
| 0.7 | 0.7 | 500 | 47326 | 0 |
| 0.7 | 0.8 | 10 | 47326 | 1840 |
| 0.7 | 0.8 | 25 | 47326 | 1359 |
| 0.7 | 0.8 | 50 | 47326 | 883 |

| | | | | |
|---|---|---|---|---|
| 0.7 | 0.8 | 100 | 47326 | 0 |
| 0.7 | 0.8 | 200 | 47326 | 721 |
| 0.7 | 0.8 | 300 | 47326 | 0 |
| 0.7 | 0.8 | 400 | 47326 | 294 |
| 0.7 | 0.8 | 500 | 47326 | 0 |
| 0.7 | 0.9 | 10 | 47326 | 1006 |
| 0.7 | 0.9 | 25 | 47326 | 1591 |
| 0.7 | 0.9 | 50 | 47326 | 0 |
| 0.7 | 0.9 | 100 | 47326 | 0 |
| 0.7 | 0.9 | 200 | 47326 | 0 |
| 0.7 | 0.9 | 300 | 47326 | 294 |
| 0.7 | 0.9 | 400 | 47326 | 0 |
| 0.7 | 0.9 | 500 | 47326 | 0 |
| 0.8 | 0.1 | 10 | 47326 | 0 |
| 0.8 | 0.1 | 25 | 47326 | 1330 |
| 0.8 | 0.1 | 50 | 47326 | 1105 |
| 0.8 | 0.1 | 100 | 47326 | 0 |
| 0.8 | 0.1 | 200 | 47326 | 0 |
| 0.8 | 0.1 | 300 | 47326 | 0 |
| 0.8 | 0.1 | 400 | 47326 | 0 |
| 0.8 | 0.1 | 500 | 47326 | 0 |
| 0.8 | 0.2 | 10 | 47326 | 2269 |
| 0.8 | 0.2 | 25 | 47326 | 0 |
| 0.8 | 0.2 | 50 | 47326 | 294 |
| 0.8 | 0.2 | 100 | 47326 | 712 |
| 0.8 | 0.2 | 200 | 47326 | 0 |
| 0.8 | 0.2 | 300 | 47326 | 0 |
| 0.8 | 0.2 | 400 | 47326 | 0 |
| 0.8 | 0.2 | 500 | 47326 | 294 |
| 0.8 | 0.3 | 10 | 47326 | 1006 |
| 0.8 | 0.3 | 25 | 47326 | 2418 |
| 0.8 | 0.3 | 50 | 47326 | 1088 |
| 0.8 | 0.3 | 100 | 47326 | 294 |

| | | | | |
|---|---|---|---|---|
| 0.8 | 0.3 | 200 | 47326 | 0 |
| 0.8 | 0.3 | 300 | 47326 | 0 |
| 0.8 | 0.3 | 400 | 47326 | 0 |
| 0.8 | 0.3 | 500 | 47326 | 0 |
| 0.8 | 0.4 | 10 | 47326 | 294 |
| 0.8 | 0.4 | 25 | 47326 | 1591 |
| 0.8 | 0.4 | 50 | 47326 | 2163 |
| 0.8 | 0.4 | 100 | 47326 | 294 |
| 0.8 | 0.4 | 200 | 47326 | 0 |
| 0.8 | 0.4 | 300 | 47326 | 294 |
| 0.8 | 0.4 | 400 | 47326 | 0 |
| 0.8 | 0.4 | 500 | 47326 | 0 |
| 0.8 | 0.5 | 10 | 47326 | 1330 |
| 0.8 | 0.5 | 25 | 47326 | 2229 |
| 0.8 | 0.5 | 50 | 47326 | 827 |
| 0.8 | 0.5 | 100 | 47326 | 712 |
| 0.8 | 0.5 | 200 | 47326 | 0 |
| 0.8 | 0.5 | 300 | 47326 | 0 |
| 0.8 | 0.5 | 400 | 47326 | 0 |
| 0.8 | 0.5 | 500 | 47326 | 0 |
| 0.8 | 0.6 | 10 | 47326 | 1867 |
| 0.8 | 0.6 | 25 | 47326 | 294 |
| 0.8 | 0.6 | 50 | 47326 | 721 |
| 0.8 | 0.6 | 100 | 47326 | 0 |
| 0.8 | 0.6 | 200 | 47326 | 0 |
| 0.8 | 0.6 | 300 | 47326 | 0 |
| 0.8 | 0.6 | 400 | 47326 | 0 |
| 0.8 | 0.6 | 500 | 47326 | 0 |
| 0.8 | 0.7 | 10 | 47326 | 1359 |
| 0.8 | 0.7 | 25 | 47326 | 1105 |
| 0.8 | 0.7 | 50 | 47326 | 1006 |
| 0.8 | 0.7 | 100 | 47326 | 294 |
| 0.8 | 0.7 | 200 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.8 | 0.7 | 300 | 47326 | 0 |
| 0.8 | 0.7 | 400 | 47326 | 0 |
| 0.8 | 0.7 | 500 | 47326 | 0 |
| 0.8 | 0.8 | 10 | 47326 | 712 |
| 0.8 | 0.8 | 25 | 47326 | 827 |
| 0.8 | 0.8 | 50 | 47326 | 1141 |
| 0.8 | 0.8 | 100 | 47326 | 294 |
| 0.8 | 0.8 | 200 | 47326 | 0 |
| 0.8 | 0.8 | 300 | 47326 | 0 |
| 0.8 | 0.8 | 400 | 47326 | 712 |
| 0.8 | 0.8 | 500 | 47326 | 294 |
| 0.8 | 0.9 | 10 | 47326 | 1867 |
| 0.8 | 0.9 | 25 | 47326 | 1324 |
| 0.8 | 0.9 | 50 | 47326 | 721 |
| 0.8 | 0.9 | 100 | 47326 | 827 |
| 0.8 | 0.9 | 200 | 47326 | 294 |
| 0.8 | 0.9 | 300 | 47326 | 0 |
| 0.8 | 0.9 | 400 | 47326 | 0 |
| 0.8 | 0.9 | 500 | 47326 | 0 |
| 0.9 | 0.1 | 10 | 47326 | 1761 |
| 0.9 | 0.1 | 25 | 47326 | 712 |
| 0.9 | 0.1 | 50 | 47326 | 721 |
| 0.9 | 0.1 | 100 | 47326 | 0 |
| 0.9 | 0.1 | 200 | 47326 | 0 |
| 0.9 | 0.1 | 300 | 47326 | 0 |
| 0.9 | 0.1 | 400 | 47326 | 0 |
| 0.9 | 0.1 | 500 | 47326 | 0 |
| 0.9 | 0.2 | 10 | 47326 | 2547 |
| 0.9 | 0.2 | 25 | 47326 | 1360 |
| 0.9 | 0.2 | 50 | 47326 | 1105 |
| 0.9 | 0.2 | 100 | 47326 | 294 |
| 0.9 | 0.2 | 200 | 47326 | 294 |
| 0.9 | 0.2 | 300 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.9 | 0.2 | 400 | 47326 | 294 |
| 0.9 | 0.2 | 500 | 47326 | 0 |
| 0.9 | 0.3 | 10 | 47326 | 2735 |
| 0.9 | 0.3 | 25 | 47326 | 712 |
| 0.9 | 0.3 | 50 | 47326 | 1141 |
| 0.9 | 0.3 | 100 | 47326 | 294 |
| 0.9 | 0.3 | 200 | 47326 | 294 |
| 0.9 | 0.3 | 300 | 47326 | 0 |
| 0.9 | 0.3 | 400 | 47326 | 0 |
| 0.9 | 0.3 | 500 | 47326 | 0 |
| 0.9 | 0.4 | 10 | 47326 | 2664 |
| 0.9 | 0.4 | 25 | 47326 | 2707 |
| 0.9 | 0.4 | 50 | 47326 | 294 |
| 0.9 | 0.4 | 100 | 47326 | 0 |
| 0.9 | 0.4 | 200 | 47326 | 712 |
| 0.9 | 0.4 | 300 | 47326 | 0 |
| 0.9 | 0.4 | 400 | 47326 | 0 |
| 0.9 | 0.4 | 500 | 47326 | 0 |
| 0.9 | 0.5 | 10 | 47326 | 721 |
| 0.9 | 0.5 | 25 | 47326 | 0 |
| 0.9 | 0.5 | 50 | 47326 | 1360 |
| 0.9 | 0.5 | 100 | 47326 | 0 |
| 0.9 | 0.5 | 200 | 47326 | 0 |
| 0.9 | 0.5 | 300 | 47326 | 0 |
| 0.9 | 0.5 | 400 | 47326 | 0 |
| 0.9 | 0.5 | 500 | 47326 | 294 |
| 0.9 | 0.6 | 10 | 47326 | 294 |
| 0.9 | 0.6 | 25 | 47326 | 0 |
| 0.9 | 0.6 | 50 | 47326 | 0 |
| 0.9 | 0.6 | 100 | 47326 | 712 |
| 0.9 | 0.6 | 200 | 47326 | 0 |
| 0.9 | 0.6 | 300 | 47326 | 0 |
| 0.9 | 0.6 | 400 | 47326 | 0 |

| | | | | |
|---|---|---|---|---|
| 0.9 | 0.6 | 500 | 47326 | 0 |
| 0.9 | 0.7 | 10 | 47326 | 2238 |
| 0.9 | 0.7 | 25 | 47326 | 0 |
| 0.9 | 0.7 | 50 | 47326 | 712 |
| 0.9 | 0.7 | 100 | 47326 | 721 |
| 0.9 | 0.7 | 200 | 47326 | 0 |
| 0.9 | 0.7 | 300 | 47326 | 0 |
| 0.9 | 0.7 | 400 | 47326 | 0 |
| 0.9 | 0.7 | 500 | 47326 | 0 |
| 0.9 | 0.8 | 10 | 47326 | 2309 |
| 0.9 | 0.8 | 25 | 47326 | 2957 |
| 0.9 | 0.8 | 50 | 47326 | 0 |
| 0.9 | 0.8 | 100 | 47326 | 294 |
| 0.9 | 0.8 | 200 | 47326 | 0 |
| 0.9 | 0.8 | 300 | 47326 | 0 |
| 0.9 | 0.8 | 400 | 47326 | 0 |
| 0.9 | 0.8 | 500 | 47326 | 0 |
| 0.9 | 0.9 | 10 | 47326 | 1105 |
| 0.9 | 0.9 | 25 | 47326 | 3134 |
| 0.9 | 0.9 | 50 | 47326 | 0 |
| 0.9 | 0.9 | 100 | 47326 | 0 |
| 0.9 | 0.9 | 200 | 47326 | 0 |
| 0.9 | 0.9 | 300 | 47326 | 0 |
| 0.9 | 0.9 | 400 | 47326 | 0 |
| 0.9 | 0.9 | 500 | 47326 | 0 |