



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»

ИНСТИТУТ КИБЕРНЕТИКИ

КАФЕДРА ВЫСШЕЙ МАТЕМАТИКИ

Лабораторная работа 3

по курсу «Случайные процессы»

Тема: **Процесс роста и мутации популяции**

Выполнил:
Студент 4-го курса
Жолковский Д.А.

Группа: КМБО-01-16

МОСКВА 2019

Лабораторная работа по случайным процессам № 3

«Процесс роста и мутации популяции»

Задание

В популяции могут находиться объекты двух видов: N-объекты и M-объекты.

Дано:

- время жизни каждого N-объекта является случайной величиной, имеющей показательное распределение с параметром $\lambda(t_b) = \gamma_1 \cdot N(t_b - 0) + 2 \cdot \gamma_1 \cdot M(t_b - 0) + 0,1$ (где t_b – время рождения объекта, $N(t_b - 0)$ – число N-объектов до момента времени t_b , $M(t_b - 0)$ – число M-объектов до момента времени t_b , γ_1 – заданный коэффициент);

- время жизни каждого M-объекта является случайной величиной, имеющей показательное распределение с параметром $\mu(t_b) = 3 \cdot \gamma_2 \cdot N(t_b - 0) + \gamma_2 \cdot M(t_b - 0)$ (где t_b – время рождения объекта, $N(t_b - 0)$ – число N-объектов до момента времени t_b , $M(t_b - 0)$ – число M-объектов до момента времени t_b , γ_2 – заданный коэффициент);

- по окончании времени жизни каждый N-объект порождает с вероятностью pn_1 один N-объект (событие $S_N(1)$), с вероятностью pn_2 два N-объекта (событие $S_N(2)$), с вероятностью $pn_{11} = 1 - pn_1 - pn_2$ один N-объект и один M-объект (событие $S_N(3)$);

- по окончании времени жизни каждый M-объект порождает с вероятностью pm_1 один M-объект (событие $S_M(1)$), ничего не порождает с вероятностью $pm_0 = 1 - pm_1$ (событие $S_M(0)$);

- до начального момента $t=0$ не было объектов, в начальный момент происходит событие $S_N(1)$ и появляется первый объект: N-объект.

Состояние системы в момент времени t характеризуется параметрами $(N(t), M(t))$, где $N(t)$ – число N-объектов, $M(t)$ – число M-объектов. Событием в развитии системы называется момент окончания жизни (исчезновения) любого из объектов и (одновременно) появления новых объектов. События могут быть 5 типов: $S_N(1)$, $S_N(2)$, $S_N(3)$, $S_M(0)$, $S_M(1)$. При появлении каждого нового объекта случайным образом в соответствии с заданным законом распределения определяется время его жизни. Считать для первого события: момент наступления события $t_{cob}(1) = 0$; тип события $Type(1) = S_N(1)$.

Требуется:

1. Провести моделирование первых 100 событий в развитии системы.
2. Составить следующие таблицы в соответствии с **Указаниями**.

Таблица 1 (см. вид таблицы в **Указаниях**) с данными о событиях:

- номер события i ;
- момент наступления события $t_{sob}(i)$;
- тип события $Type(i)$;
- время жизни появившихся новых объектов (2 столбца) $t_{ж1}(i)$, $t_{ж2}(i)$;
- состояние системы после события $C(i)$;
- время ожидания до следующего события $t_{ож}(i)$;
- номер $J_{кж}(i)$ объекта, у которого раньше закончится жизнь;
- вид этого исчезающего объекта $Gen_{кж}(i)$.

Таблица 2 (см. вид таблицы в **Указаниях**) с данными об объектах:

- номер объекта j ;
- вид объекта $Gen(j)$ (N или M);
- момент появления (рождения) объекта $t_b(j)$;
- время жизни объекта $t_l(j)$;
- момент исчезновения объекта $t_d(j)$;
- номера объектов-потомков (2 столбца) $Des_1(j)$, $Des_2(j)$.

Таблица 3 с данными о типах событий (см. вид таблицы в **Указаниях**).

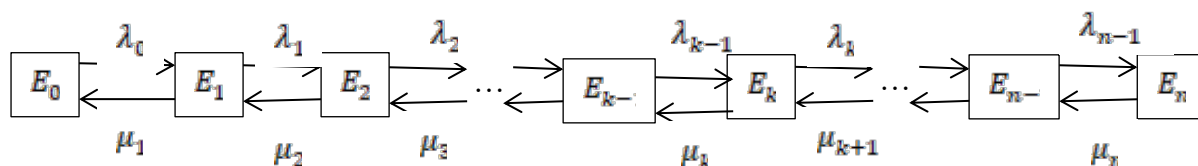
Таблица 4 с данными о видах объектов (см. вид таблицы в **Указаниях**).

Таблица 5 с данными о состояниях (см. вид таблицы в **Указаниях**).

Вычисления времени проводить с точностью до 0,00001 .

Краткие теоретические сведения

Процессом гибели и размножения называется марковский процесс с непрерывным временем, граф состояний которого изображён на рисунке:



λ_i – интенсивность размножения

μ_i – интенсивность гибели

Система уравнений Колмогорова для вероятностей состояний процесса гибели и размножения имеет вид:

$$\begin{cases} p'_0(t) = -\lambda_0 p_0(t) + \mu_1 p_1(t) \\ p'_k(t) = -(\lambda_k + \mu_k) p_k(t) + \lambda_{k-1} p_{k-1}(t) + \mu_{k+1} p_{k+1}(t), (k = 1, 2, \dots, n-1) \\ p'_n(t) = -\mu_n p_n(t) + \lambda_{n-1} p_{n-1}(t) \end{cases}$$

В том случае, когда у процесса гибели и рождения все интенсивности положительны $\lambda_i > 0, i = 0, 1, \dots, n-1, \mu_j > 0, j = 1, \dots, n$, а число состояний n конечно, процесс является эргодическим, и существуют предельные вероятности состояний $p_k(\infty) = p_k$:

$$p_0(\infty) = p_0, p_1(\infty) = \frac{\lambda_0}{\mu_1} p_0, \dots, p_k(\infty) = \frac{\lambda_0 \lambda_1 \dots \lambda_{k-1}}{\mu_1 \mu_2 \dots \mu_k} p_0, \dots,$$

$$p_n(\infty) = \frac{\lambda_0 \lambda_1 \dots \lambda_{n-1}}{\mu_1 \mu_2 \dots \mu_n} p_0, \quad p_0 = \left\{ \sum_{k=0}^n \frac{\lambda_0 \lambda_1 \dots \lambda_{k-1}}{\mu_1 \mu_2 \dots \mu_k} \right\}^{-1},$$

являющиеся также и стационарными вероятностями.

Средства высокоуровневого интерпретируемого языка программирования Python, которые использованы в программе расчета

`scipy.stats.expon` – класс для работы с экспоненциальным распределением

`max(x)` – выбор максимального значения в массиве `x`;

`min(x)` – выбор минимального значения в массиве `x`;

`np.zeros(m,n)` – создание нулевой матрицы размера `m` x `n`;

`abs(x)` – модуль числа `x`;

`np.rand` - генерация равномерного псевдослучайного числа в диапазоне 0..1

`np.random.exponential` – генерация `N` псевдослучайных чисел, распределенных по показательному закону с параметром $\lambda = \text{lambda}$;

`sorted(x)` – упорядочение по возрастанию массива `x`;

Результаты расчетов

Вариант №10

Исходные данные

$$pn_1 = 0,318$$

$$pn_2 = 0,318$$

$$pm_1 = 0,25$$

$$\gamma_1 = 0,35$$

$$\gamma_2 = 0,21$$

Таблица 1

| № события | $t_{\text{сб}}$ | Type | $t_{\text{ж1}}$ | $t_{\text{ж2}}$ | C | $t_{\text{ок}}$ | $I_{\text{ок}}$ | $Gen_{\text{ок}}$ |
|-----------|-----------------|----------|-----------------|-----------------|--------|-----------------|-----------------|-------------------|
| 1 | 0 | $S_N(1)$ | 6.64915 | -1 | (1,0) | 6.64915 | 1 | N |
| 2 | 6.64915 | $S_N(2)$ | 0.09554 | 2.83492 | (2,0) | 0.09554 | 2 | N |
| 3 | 6.74469 | $S_N(3)$ | 0.40989 | 0.8441 | (2,1) | 0.40989 | 4 | N |
| 4 | 7.15458 | $S_M(0)$ | 0.00775 | 0.03357 | (3,1) | 0.00775 | 6 | N |
| 5 | 7.16233 | $S_N(3)$ | 0.1076 | 0.4915 | (4,1) | 0.02582 | 7 | N |
| 6 | 7.18815 | $S_N(2)$ | 0.04375 | 0.58323 | (4,2) | 0.04375 | 10 | M |
| 7 | 7.2319 | $S_N(2)$ | -1 | -1 | (4,1) | 0.03803 | 8 | N |
| 8 | 7.26993 | $S_M(1)$ | 0.20504 | 0.51793 | (5,1) | 0.20504 | 12 | N |
| 9 | 7.47497 | $S_N(1)$ | 0.06481 | 1.15915 | (5,2) | 0.06481 | 14 | M |
| 10 | 7.53978 | $S_N(2)$ | -1 | -1 | (5,1) | 0.04901 | 5 | M |
| 11 | 7.58879 | $S_N(3)$ | 0.38379 | -1 | (5,1) | 0.06504 | 9 | N |
| 12 | 7.65383 | $S_M(0)$ | 0.39782 | 0.51644 | (6,1) | 0.11755 | 11 | N |
| 13 | 7.77138 | $S_M(1)$ | 0.19322 | -1 | (6,1) | 0.01648 | 13 | N |
| 14 | 7.78786 | $S_N(3)$ | 0.02582 | 0.06851 | (7,1) | 0.02582 | 20 | N |
| 15 | 7.81368 | $S_N(1)$ | 0.25677 | 0.43915 | (7,2) | 0.04269 | 21 | N |
| 16 | 7.85637 | $S_M(1)$ | 0.13855 | -1 | (7,2) | 0.10823 | 19 | N |
| 17 | 7.9646 | $S_M(0)$ | 0.12213 | 0.56656 | (8,2) | 0.00798 | 16 | M |
| 18 | 7.97258 | $S_M(1)$ | -1 | -1 | (8,1) | 0.02234 | 24 | N |
| 19 | 7.99492 | $S_N(2)$ | 0.01396 | 0.172 | (9,1) | 0.01396 | 27 | N |
| 20 | 8.00888 | $S_M(0)$ | 0.00363 | 0.00679 | (10,1) | 0.00363 | 29 | N |
| 21 | 8.01251 | $S_N(1)$ | 0.1874 | 0.26046 | (11,1) | 0.00316 | 30 | N |
| 22 | 8.01567 | $S_N(3)$ | 0.47083 | 0.50479 | (12,1) | 0.03598 | 17 | N |
| 23 | 8.05165 | $S_M(1)$ | 0.47137 | -1 | (12,1) | 0.0188 | 22 | M |
| 24 | 8.07045 | $S_N(3)$ | 0.03263 | -1 | (12,1) | 0.01628 | 25 | N |
| 25 | 8.08673 | $S_M(1)$ | 0.06125 | 0.12362 | (12,2) | 0.01635 | 36 | M |
| 26 | 8.10308 | $S_N(2)$ | -1 | -1 | (12,1) | 0.0449 | 37 | N |
| 27 | 8.14798 | $S_M(0)$ | 0.27266 | -1 | (12,1) | 0.01894 | 28 | N |
| 28 | 8.16692 | $S_M(1)$ | 0.10938 | 0.50261 | (12,2) | 0.00335 | 18 | N |
| 29 | 8.17027 | $S_N(2)$ | 0.16493 | -1 | (12,2) | 0.02964 | 31 | N |
| 30 | 8.19991 | $S_M(0)$ | 0.10686 | 0.53546 | (13,2) | 0.01044 | 38 | M |
| 31 | 8.21035 | $S_N(3)$ | -1 | -1 | (13,1) | 0.04248 | 23 | N |
| 32 | 8.25283 | $S_M(1)$ | 0.01913 | -1 | (13,1) | 0.01913 | 45 | N |
| 33 | 8.27196 | $S_N(1)$ | 0.00317 | 0.30067 | (14,1) | 0.00101 | 32 | N |
| 34 | 8.27297 | $S_M(0)$ | 0.00722 | 0.15491 | (14,2) | 0.00216 | 46 | N |

| | | | | | | | | |
|----|---------|----------|---------|---------|--------|---------|-----|---|
| 35 | 8.27513 | $S_N(1)$ | 0.04413 | -1 | (14,2) | 0.00117 | 40 | M |
| 36 | 8.2763 | $S_N(1)$ | -1 | -1 | (14,1) | 0.00389 | 48 | M |
| 37 | 8.28019 | $S_N(1)$ | -1 | -1 | (14,0) | 0.02658 | 43 | N |
| 38 | 8.30677 | $S_N(1)$ | 0.02846 | 0.3388 | (14,1) | 0.01249 | 50 | N |
| 39 | 8.31926 | $S_N(3)$ | 0.00247 | 0.21346 | (14,2) | 0.00247 | 53 | M |
| 40 | 8.32173 | $S_N(3)$ | -1 | -1 | (14,1) | 0.01347 | 42 | N |
| 41 | 8.3352 | $S_M(1)$ | 0.0008 | 0.19234 | (15,1) | 3e-05 | 51 | N |
| 42 | 8.33523 | $S_M(0)$ | 0.01268 | 0.08333 | (16,1) | 0.00077 | 55 | N |
| 43 | 8.336 | $S_N(2)$ | 0.05211 | -1 | (16,1) | 0.01191 | 57 | N |
| 44 | 8.34791 | $S_N(2)$ | 0.03693 | 0.19265 | (16,2) | 0.03693 | 60 | M |
| 45 | 8.38484 | $S_N(1)$ | -1 | -1 | (16,1) | 0.00327 | 59 | N |
| 46 | 8.38811 | $S_M(0)$ | 0.30824 | -1 | (16,1) | 0.03045 | 58 | N |
| 47 | 8.41856 | $S_N(1)$ | 0.08658 | 0.10593 | (16,2) | 0.00208 | 39 | N |
| 48 | 8.42064 | $S_N(1)$ | 0.17674 | 0.48851 | (17,2) | 0.00724 | 49 | N |
| 49 | 8.42788 | $S_N(3)$ | 0.07656 | 0.22754 | (18,2) | 0.05862 | 33 | N |
| 50 | 8.4865 | $S_N(3)$ | 0.0464 | 0.17112 | (19,2) | 0.01794 | 67 | N |
| 51 | 8.50444 | $S_N(3)$ | 0.05958 | 0.33443 | (20,2) | 0.0007 | 63 | N |
| 52 | 8.50514 | $S_M(1)$ | 0.01549 | 0.032 | (20,3) | 0.01532 | 34 | N |
| 53 | 8.52046 | $S_M(1)$ | 0.04698 | 0.06799 | (20,4) | 0.00017 | 73 | N |
| 54 | 8.52063 | $S_N(2)$ | 0.02806 | -1 | (20,4) | 0.00239 | 35 | N |
| 55 | 8.52302 | $S_N(1)$ | 0.11685 | -1 | (20,4) | 0.00147 | 64 | M |
| 56 | 8.52449 | $S_M(0)$ | -1 | -1 | (20,3) | 0.00305 | 56 | N |
| 57 | 8.52754 | $S_N(2)$ | 0.02731 | 0.05932 | (20,4) | 0.00362 | 26 | N |
| 58 | 8.53116 | $S_M(1)$ | 0.1798 | -1 | (20,4) | 0.00156 | 54 | N |
| 59 | 8.53272 | $S_M(1)$ | 0.01178 | 0.08637 | (20,5) | 0.00018 | 69 | N |
| 60 | 8.5329 | $S_N(1)$ | 0.14056 | -1 | (20,5) | 0.00424 | 74 | M |
| 61 | 8.53714 | $S_N(3)$ | 0.14127 | -1 | (20,5) | 0.00342 | 61 | N |
| 62 | 8.54056 | $S_M(1)$ | 0.01914 | 0.08257 | (21,5) | 0.00394 | 82 | N |
| 63 | 8.5445 | $S_N(1)$ | 0.23829 | -1 | (21,5) | 0.00419 | 77 | N |
| 64 | 8.54869 | $S_N(2)$ | 0.05653 | 0.08447 | (22,5) | 0.00616 | 79 | M |
| 65 | 8.55485 | $S_M(0)$ | -1 | -1 | (22,4) | 0.00485 | 86 | N |
| 66 | 8.5597 | $S_N(1)$ | 0.52953 | -1 | (22,4) | 0.00432 | 71 | N |
| 67 | 8.56402 | $S_N(3)$ | 0.03716 | -1 | (22,4) | 0.00342 | 75 | N |
| 68 | 8.56744 | $S_N(3)$ | 0.04961 | 0.15757 | (22,5) | 0.00519 | 47 | N |
| 69 | 8.57263 | $S_M(1)$ | 0.05779 | 0.11153 | (22,6) | 0.01423 | 80 | N |
| 70 | 8.58686 | $S_N(1)$ | 0.04805 | -1 | (22,6) | 0.00159 | 76 | M |
| 71 | 8.58845 | $S_N(3)$ | 0.06961 | -1 | (22,6) | 0.00893 | 65 | N |
| 72 | 8.59738 | $S_M(1)$ | 0.10442 | 0.1619 | (23,6) | 0.0038 | 92 | N |
| 73 | 8.60118 | $S_M(1)$ | 0.01916 | 0.0586 | (23,7) | 0.00404 | 89 | N |
| 74 | 8.60522 | $S_M(1)$ | 0.09245 | 0.23269 | (24,7) | 0.01183 | 93 | M |
| 75 | 8.61705 | $S_N(3)$ | -1 | -1 | (24,6) | 0.00204 | 83 | M |
| 76 | 8.61909 | $S_M(0)$ | -1 | -1 | (24,5) | 0.00125 | 101 | N |
| 77 | 8.62034 | $S_M(0)$ | 0.03758 | 0.09307 | (25,5) | 0.00279 | 87 | N |
| 78 | 8.62313 | $S_N(3)$ | 0.01363 | -1 | (25,5) | 0.00729 | 95 | M |
| 79 | 8.63042 | $S_N(1)$ | -1 | -1 | (25,4) | 0.00274 | 90 | N |
| 80 | 8.63316 | $S_M(0)$ | 0.03207 | 0.03241 | (25,5) | 0.00096 | 15 | N |
| 81 | 8.63412 | $S_N(3)$ | 0.01434 | 0.19302 | (26,5) | 0.00079 | 97 | N |
| 82 | 8.63491 | $S_N(1)$ | 0.00218 | 0.03477 | (26,6) | 0.00185 | 107 | N |
| 83 | 8.63676 | $S_M(1)$ | 0.00804 | -1 | (26,6) | 0.00033 | 112 | N |
| 84 | 8.63709 | $S_N(3)$ | 0.00337 | 0.05425 | (27,6) | 0.00278 | 78 | N |
| 85 | 8.63987 | $S_M(1)$ | 0.08546 | 0.08666 | (27,7) | 0.00059 | 115 | N |
| 86 | 8.64046 | $S_N(2)$ | 0.11119 | 0.12149 | (27,8) | 0.00434 | 114 | N |

| | | | | | | | | |
|-----|---------|----------|---------|---------|--------|---------|-----|---|
| 87 | 8.6448 | $S_M(1)$ | 0.08946 | -1 | (27,8) | 0.00077 | 52 | M |
| 88 | 8.64557 | $S_M(0)$ | -1 | -1 | (27,7) | 0.00289 | 110 | N |
| 89 | 8.64846 | $S_M(0)$ | 0.05146 | 0.08298 | (27,8) | 0.00696 | 68 | N |
| 90 | 8.65542 | $S_M(0)$ | 0.00104 | -1 | (27,8) | 0.00104 | 124 | N |
| 91 | 8.65646 | $S_N(1)$ | 0.05484 | 0.27359 | (27,9) | 0.00116 | 70 | N |
| 92 | 8.65762 | $S_N(3)$ | 0.03498 | -1 | (27,9) | 0.0003 | 105 | N |
| 93 | 8.65792 | $S_N(1)$ | 0.13386 | -1 | (27,9) | 0.00014 | 98 | M |
| 94 | 8.65806 | $S_N(3)$ | 0.0256 | -1 | (27,9) | 0.00172 | 102 | M |
| 95 | 8.65978 | $S_N(1)$ | -1 | -1 | (27,8) | 0.00545 | 108 | M |
| 96 | 8.66523 | $S_M(1)$ | -1 | -1 | (27,7) | 0.00034 | 109 | N |
| 97 | 8.66557 | $S_N(1)$ | 0.02451 | 0.04799 | (28,7) | 0.00396 | 41 | N |
| 98 | 8.66953 | $S_M(0)$ | 0.02545 | -1 | (28,7) | 0.00015 | 113 | M |
| 99 | 8.66968 | $S_N(2)$ | -1 | -1 | (28,6) | 0.00378 | 84 | N |
| 100 | 8.67346 | $S_M(1)$ | 0.0422 | 0.17393 | (28,7) | 0.00495 | 85 | M |

Таблица 2

| № объекта | Gen | t_p | t_l | t_d | Des_1 | Des_2 |
|-----------|-------|---------|---------|---------|---------|---------|
| 1 | N | 0 | 6.64915 | 6.64915 | 2 | 3 |
| 2 | N | 6.64915 | 0.09554 | 6.74469 | 4 | 5 |
| 3 | N | 6.64915 | 2.83492 | 9.48407 | -1 | -1 |
| 4 | N | 6.74469 | 0.40989 | 7.15458 | 6 | 7 |
| 5 | M | 6.74469 | 0.8441 | 7.58879 | 16 | -1 |
| 6 | N | 7.15458 | 0.00775 | 7.16233 | 8 | 9 |
| 7 | N | 7.15458 | 0.03357 | 7.18815 | 10 | 11 |
| 8 | N | 7.16233 | 0.1076 | 7.26993 | 12 | 13 |
| 9 | N | 7.16233 | 0.4915 | 7.65383 | 17 | 18 |
| 10 | M | 7.18815 | 0.04375 | 7.2319 | -1 | -1 |
| 11 | N | 7.18815 | 0.58323 | 7.77138 | 19 | -1 |
| 12 | N | 7.26993 | 0.20504 | 7.47497 | 14 | 15 |
| 13 | N | 7.26993 | 0.51793 | 7.78786 | 20 | 21 |
| 14 | M | 7.47497 | 0.06481 | 7.53978 | -1 | -1 |
| 15 | N | 7.47497 | 1.15915 | 8.63412 | 110 | 111 |
| 16 | M | 7.58879 | 0.38379 | 7.97258 | -1 | -1 |
| 17 | N | 7.65383 | 0.39782 | 8.05165 | 35 | -1 |
| 18 | N | 7.65383 | 0.51644 | 8.17027 | 42 | -1 |
| 19 | N | 7.77138 | 0.19322 | 7.9646 | 25 | 26 |
| 20 | N | 7.78786 | 0.02582 | 7.81368 | 22 | 23 |
| 21 | N | 7.78786 | 0.06851 | 7.85637 | 24 | -1 |
| 22 | M | 7.81368 | 0.25677 | 8.07045 | 36 | -1 |
| 23 | N | 7.81368 | 0.43915 | 8.25283 | 45 | -1 |
| 24 | N | 7.85637 | 0.13855 | 7.99492 | 27 | 28 |
| 25 | N | 7.9646 | 0.12213 | 8.08673 | 37 | 38 |
| 26 | N | 7.9646 | 0.56656 | 8.53116 | 81 | -1 |
| 27 | N | 7.99492 | 0.01396 | 8.00888 | 29 | 30 |
| 28 | N | 7.99492 | 0.172 | 8.16692 | 40 | 41 |
| 29 | N | 8.00888 | 0.00363 | 8.01251 | 31 | 32 |
| 30 | N | 8.00888 | 0.00679 | 8.01567 | 33 | 34 |
| 31 | N | 8.01251 | 0.1874 | 8.19991 | 43 | 44 |
| 32 | N | 8.01251 | 0.26046 | 8.27297 | 48 | 49 |
| 33 | N | 8.01567 | 0.47083 | 8.4865 | 69 | 70 |
| 34 | N | 8.01567 | 0.50479 | 8.52046 | 75 | 76 |
| 35 | N | 8.05165 | 0.47137 | 8.52302 | 78 | -1 |

| | | | | | | |
|----|---|---------|---------|---------|-----|-----|
| 36 | M | 8.07045 | 0.03263 | 8.10308 | -1 | -1 |
| 37 | N | 8.08673 | 0.06125 | 8.14798 | 39 | -1 |
| 38 | M | 8.08673 | 0.12362 | 8.21035 | -1 | -1 |
| 39 | N | 8.14798 | 0.27266 | 8.42064 | 65 | 66 |
| 40 | M | 8.16692 | 0.10938 | 8.2763 | -1 | -1 |
| 41 | N | 8.16692 | 0.50261 | 8.66953 | 132 | -1 |
| 42 | N | 8.17027 | 0.16493 | 8.3352 | 55 | 56 |
| 43 | N | 8.19991 | 0.10686 | 8.30677 | 51 | 52 |
| 44 | N | 8.19991 | 0.53546 | 8.73537 | -1 | -1 |
| 45 | N | 8.25283 | 0.01913 | 8.27196 | 46 | 47 |
| 46 | N | 8.27196 | 0.00317 | 8.27513 | 50 | -1 |
| 47 | N | 8.27196 | 0.30067 | 8.57263 | 95 | 96 |
| 48 | M | 8.27297 | 0.00722 | 8.28019 | -1 | -1 |
| 49 | N | 8.27297 | 0.15491 | 8.42788 | 67 | 68 |
| 50 | N | 8.27513 | 0.04413 | 8.31926 | 53 | 54 |
| 51 | N | 8.30677 | 0.02846 | 8.33523 | 57 | 58 |
| 52 | M | 8.30677 | 0.3388 | 8.64557 | -1 | -1 |
| 53 | M | 8.31926 | 0.00247 | 8.32173 | -1 | -1 |
| 54 | N | 8.31926 | 0.21346 | 8.53272 | 82 | 83 |
| 55 | N | 8.3352 | 0.0008 | 8.336 | 59 | -1 |
| 56 | N | 8.3352 | 0.19234 | 8.52754 | 79 | 80 |
| 57 | N | 8.33523 | 0.01268 | 8.34791 | 60 | 61 |
| 58 | N | 8.33523 | 0.08333 | 8.41856 | 63 | 64 |
| 59 | N | 8.336 | 0.05211 | 8.38811 | 62 | -1 |
| 60 | M | 8.34791 | 0.03693 | 8.38484 | -1 | -1 |
| 61 | N | 8.34791 | 0.19265 | 8.54056 | 86 | 87 |
| 62 | N | 8.38811 | 0.30824 | 8.69635 | -1 | -1 |
| 63 | N | 8.41856 | 0.08658 | 8.50514 | 73 | 74 |
| 64 | M | 8.41856 | 0.10593 | 8.52449 | -1 | -1 |
| 65 | N | 8.42064 | 0.17674 | 8.59738 | 99 | 100 |
| 66 | N | 8.42064 | 0.48851 | 8.90915 | -1 | -1 |
| 67 | N | 8.42788 | 0.07656 | 8.50444 | 71 | 72 |
| 68 | N | 8.42788 | 0.22754 | 8.65542 | 124 | -1 |
| 69 | N | 8.4865 | 0.0464 | 8.5329 | 84 | -1 |
| 70 | N | 8.4865 | 0.17112 | 8.65762 | 127 | -1 |
| 71 | N | 8.50444 | 0.05958 | 8.56402 | 92 | -1 |
| 72 | N | 8.50444 | 0.33443 | 8.83887 | -1 | -1 |
| 73 | N | 8.50514 | 0.01549 | 8.52063 | 77 | -1 |
| 74 | M | 8.50514 | 0.032 | 8.53714 | 85 | -1 |
| 75 | N | 8.52046 | 0.04698 | 8.56744 | 93 | 94 |
| 76 | M | 8.52046 | 0.06799 | 8.58845 | 98 | -1 |
| 77 | N | 8.52063 | 0.02806 | 8.54869 | 89 | 90 |
| 78 | N | 8.52302 | 0.11685 | 8.63987 | 117 | 118 |
| 79 | M | 8.52754 | 0.02731 | 8.55485 | -1 | -1 |
| 80 | N | 8.52754 | 0.05932 | 8.58686 | 97 | -1 |
| 81 | N | 8.53116 | 0.1798 | 8.71096 | -1 | -1 |
| 82 | N | 8.53272 | 0.01178 | 8.5445 | 88 | -1 |
| 83 | M | 8.53272 | 0.08637 | 8.61909 | -1 | -1 |
| 84 | N | 8.5329 | 0.14056 | 8.67346 | 133 | 134 |
| 85 | M | 8.53714 | 0.14127 | 8.67841 | -1 | -1 |
| 86 | N | 8.54056 | 0.01914 | 8.5597 | 91 | -1 |
| 87 | N | 8.54056 | 0.08257 | 8.62313 | 107 | -1 |

| | | | | | | |
|-----|---|---------|---------|---------|-----|-----|
| 88 | N | 8.5445 | 0.23829 | 8.78279 | -1 | -1 |
| 89 | N | 8.54869 | 0.05653 | 8.60522 | 103 | 104 |
| 90 | N | 8.54869 | 0.08447 | 8.63316 | 108 | 109 |
| 91 | N | 8.5597 | 0.52953 | 9.08923 | -1 | -1 |
| 92 | N | 8.56402 | 0.03716 | 8.60118 | 101 | 102 |
| 93 | M | 8.56744 | 0.04961 | 8.61705 | -1 | -1 |
| 94 | N | 8.56744 | 0.15757 | 8.72501 | -1 | -1 |
| 95 | M | 8.57263 | 0.05779 | 8.63042 | -1 | -1 |
| 96 | N | 8.57263 | 0.11153 | 8.68416 | -1 | -1 |
| 97 | N | 8.58686 | 0.04805 | 8.63491 | 112 | 113 |
| 98 | M | 8.58845 | 0.06961 | 8.65806 | 129 | -1 |
| 99 | N | 8.59738 | 0.10442 | 8.7018 | -1 | -1 |
| 100 | N | 8.59738 | 0.1619 | 8.75928 | -1 | -1 |
| 101 | N | 8.60118 | 0.01916 | 8.62034 | 105 | 106 |
| 102 | M | 8.60118 | 0.0586 | 8.65978 | -1 | -1 |
| 103 | N | 8.60522 | 0.09245 | 8.69767 | -1 | -1 |
| 104 | N | 8.60522 | 0.23269 | 8.83791 | -1 | -1 |
| 105 | N | 8.62034 | 0.03758 | 8.65792 | 128 | -1 |
| 106 | N | 8.62034 | 0.09307 | 8.71341 | -1 | -1 |
| 107 | N | 8.62313 | 0.01363 | 8.63676 | 114 | -1 |
| 108 | M | 8.63316 | 0.03207 | 8.66523 | -1 | -1 |
| 109 | N | 8.63316 | 0.03241 | 8.66557 | 130 | 131 |
| 110 | N | 8.63412 | 0.01434 | 8.64846 | 122 | 123 |
| 111 | N | 8.63412 | 0.19302 | 8.82714 | -1 | -1 |
| 112 | N | 8.63491 | 0.00218 | 8.63709 | 115 | 116 |
| 113 | M | 8.63491 | 0.03477 | 8.66968 | -1 | -1 |
| 114 | N | 8.63676 | 0.00804 | 8.6448 | 121 | -1 |
| 115 | N | 8.63709 | 0.00337 | 8.64046 | 119 | 120 |
| 116 | N | 8.63709 | 0.05425 | 8.69134 | -1 | -1 |
| 117 | N | 8.63987 | 0.08546 | 8.72533 | -1 | -1 |
| 118 | M | 8.63987 | 0.08666 | 8.72653 | -1 | -1 |
| 119 | N | 8.64046 | 0.11119 | 8.75165 | -1 | -1 |
| 120 | M | 8.64046 | 0.12149 | 8.76195 | -1 | -1 |
| 121 | N | 8.6448 | 0.08946 | 8.73426 | -1 | -1 |
| 122 | N | 8.64846 | 0.05146 | 8.69992 | -1 | -1 |
| 123 | M | 8.64846 | 0.08298 | 8.73144 | -1 | -1 |
| 124 | N | 8.65542 | 0.00104 | 8.65646 | 125 | 126 |
| 125 | M | 8.65646 | 0.05484 | 8.7113 | -1 | -1 |
| 126 | N | 8.65646 | 0.27359 | 8.93005 | -1 | -1 |
| 127 | N | 8.65762 | 0.03498 | 8.6926 | -1 | -1 |
| 128 | N | 8.65792 | 0.13386 | 8.79178 | -1 | -1 |
| 129 | M | 8.65806 | 0.0256 | 8.68366 | -1 | -1 |
| 130 | N | 8.66557 | 0.02451 | 8.69008 | -1 | -1 |
| 131 | N | 8.66557 | 0.04799 | 8.71356 | -1 | -1 |
| 132 | N | 8.66953 | 0.02545 | 8.69498 | -1 | -1 |
| 133 | M | 8.67346 | 0.0422 | 8.71566 | -1 | -1 |
| 134 | N | 8.67346 | 0.17393 | 8.84739 | -1 | -1 |

Анализ результатов и выводы

Таблица 3

| Тип события | $S_N(1)$ | $S_N(2)$ | $S_N(3)$ | $S_M(1)$ | $S_M(0)$ | |
|-----------------------|----------|----------|----------|----------|----------|-----|
| Число событий | 25 | 27 | 25 | 18 | 5 | 100 |
| Относительная частота | 0.25000 | 0.27000 | 0.25000 | 0.18000 | 0.05000 | 1 |

Таблица 4

| Вид объекта | Число появившихся объектов за время $[0, t_{\text{собр}}(100)]$ | Число объектов в момент $t_{\text{собр}}(100)$ |
|-------------|---|--|
| N | 104 | 28 |
| M | 30 | 7 |

Таблица 5

| Состояние | $n_{\text{сост}}$ | $v_{\text{сост}}$ | $T_{\text{сост}}$ | $\Delta_{\text{сост}}$ |
|-----------|-------------------|-------------------|-------------------|------------------------|
| (1,0) | 1 | 0.01 | 6.64915 | 0.76661 |
| (2,0) | 1 | 0.01 | 0.09554 | 0.01102 |
| (2,1) | 1 | 0.01 | 0.40989 | 0.04726 |
| (3,1) | 1 | 0.01 | 0.00775 | 0.00089 |
| (4,1) | 2 | 0.02 | 0.06385 | 0.00736 |
| (4,2) | 1 | 0.01 | 0.04375 | 0.00504 |
| (5,1) | 3 | 0.03 | 0.31909 | 0.03679 |
| (5,2) | 1 | 0.01 | 0.06481 | 0.00747 |
| (6,1) | 2 | 0.02 | 0.13403 | 0.01545 |
| (7,1) | 1 | 0.01 | 0.02582 | 0.00298 |
| (7,2) | 2 | 0.02 | 0.15092 | 0.0174 |
| (8,2) | 1 | 0.01 | 0.00798 | 0.00092 |
| (8,1) | 1 | 0.01 | 0.02234 | 0.00258 |
| (9,1) | 1 | 0.01 | 0.01396 | 0.00161 |
| (10,1) | 1 | 0.01 | 0.00363 | 0.00042 |
| (11,1) | 1 | 0.01 | 0.00316 | 0.00036 |
| (12,1) | 5 | 0.05 | 0.1349 | 0.01555 |
| (12,2) | 3 | 0.03 | 0.04934 | 0.00569 |
| (13,2) | 1 | 0.01 | 0.01044 | 0.0012 |
| (13,1) | 2 | 0.02 | 0.06161 | 0.0071 |
| (14,1) | 4 | 0.04 | 0.03086 | 0.00356 |
| (14,2) | 3 | 0.03 | 0.0058 | 0.00067 |
| (14,0) | 1 | 0.01 | 0.02658 | 0.00306 |
| (15,1) | 1 | 0.01 | 3e-05 | 0.0 |
| (16,1) | 4 | 0.04 | 0.0464 | 0.00535 |
| (16,2) | 2 | 0.02 | 0.03901 | 0.0045 |
| (17,2) | 1 | 0.01 | 0.00724 | 0.00083 |
| (18,2) | 1 | 0.01 | 0.05862 | 0.00676 |
| (19,2) | 1 | 0.01 | 0.01794 | 0.00207 |

| | | | | |
|--------|-----|---------|---------|---------|
| (20,2) | 1 | 0.01 | 0.0007 | 8e-05 |
| (20,3) | 2 | 0.02 | 0.01837 | 0.00212 |
| (20,4) | 5 | 0.05 | 0.00921 | 0.00106 |
| (20,5) | 3 | 0.03 | 0.00784 | 0.0009 |
| (21,5) | 2 | 0.02 | 0.00813 | 0.00094 |
| (22,5) | 2 | 0.02 | 0.01135 | 0.00131 |
| (22,4) | 3 | 0.03 | 0.01259 | 0.00145 |
| (22,6) | 3 | 0.03 | 0.02475 | 0.00285 |
| (23,6) | 1 | 0.01 | 0.0038 | 0.00044 |
| (23,7) | 1 | 0.01 | 0.00404 | 0.00047 |
| (24,7) | 1 | 0.01 | 0.01183 | 0.00136 |
| (24,6) | 1 | 0.01 | 0.00204 | 0.00024 |
| (24,5) | 1 | 0.01 | 0.00125 | 0.00014 |
| (25,5) | 3 | 0.03 | 0.01104 | 0.00127 |
| (25,4) | 1 | 0.01 | 0.00274 | 0.00032 |
| (26,5) | 1 | 0.01 | 0.00079 | 9e-05 |
| (26,6) | 2 | 0.02 | 0.00218 | 0.00025 |
| (27,6) | 1 | 0.01 | 0.00278 | 0.00032 |
| (27,7) | 3 | 0.03 | 0.00382 | 0.00044 |
| (27,8) | 5 | 0.05 | 0.01856 | 0.00214 |
| (27,9) | 4 | 0.04 | 0.00332 | 0.00038 |
| (28,7) | 3 | 0.03 | 0.00906 | 0.00104 |
| (28,6) | 1 | 0.01 | 0.00378 | 0.00044 |
| | 100 | 1.00000 | 8.67841 | 1.00000 |

Список литературы

1. Лобузов А.А., Гумляева С.Д., Норин Н.В. Задачи по теории случайных процессов. — М.: МИРЭА, 1993.
2. Булинский А. В., А. Н. Ширяев А. Н. Теория случайных процессов: Учебник для вузов. — М.: ФИЗМАТЛИТ, 2005
3. Вентцель Е. С., Овчаров Л. А. Теория случайных процессов и ее инженерные приложения: Учеб. пособие для вузов. — М.: Высшая школа, 2007.

Приложение (Листинг программы)

```
# coding: utf-8

import numpy as np
from scipy.stats import expon
import random

#####

pn1 = 0.318
pn2 = 0.318
pm1 = 0.250
gamma1 = 0.35
gamma2 = 0.21

#####

Objects_arr = []
Num_of_N_objects = 0
Num_of_M_objects = 0

Type = 'S_n(1)'
t_life = round(np.random.exponential(1/0.1), 5)

Objects_arr.append([1, 0, t_life, t_life, 'N'])
Num_of_N_objects += 1

t_wait = t_life
total_num = 1
Arr_of_types = [1,0,0,0,0]
Total_N_objects = 1
Total_M_objects = 0

#####

def add_to_arr_objects(obj):
    global Num_of_N_objects
    global Num_of_M_objects
    if(obj[-1] == 'N'):
        Num_of_N_objects += 1
    else:
        Num_of_M_objects += 1
    Objects_arr.append(obj)
```

```

j = len(Objects_arr) - 1
while(j > 0 and Objects_arr[j - 1][-2] > Objects_arr[j][-2]):
    temp = Objects_arr[j-1]
    Objects_arr[j - 1] = Objects_arr[j]
    Objects_arr[j] = temp
    j -= 1

def delete_object():
    global Num_of_N_objects
    global Num_of_M_objects
    if(Objects_arr[0][-1] == 'N'):
        Num_of_N_objects -= 1
    else:
        Num_of_M_objects -= 1
    Objects_arr.pop(0)

#####

table1 = open('Data/table1.txt', 'w')
table5 = dict()
table5['(1,0)'] = [1,t_life]

table2 = []
table2.append([1,'N',0,Objects_arr[0][2],Objects_arr[0][2], -1, -1])
table1.write('1 ' + str(Objects_arr[0][1]) + ' ' + Type + ' ' +
str(Objects_arr[0][2]) + ' - (1,0) ' + str(Objects_arr[0][3]) + ' ' + '1 ' + 'N
')
Total_events = 1
Total_time = t_life
while(Total_events < 100):
    Time_spent = 0 #сколько времени проведено в данном состоянии
    print(Objects_arr)
    Total_events += 1
    p = np.random.rand()
    if(Objects_arr[0][-1] == 'N'):
        if(0 <= p <= pn1):
            Type = 'S_n(1)'
            Arr_of_types[0] += 1
            total_num += 1
            #создание нового объекта
            t_life = round(np.random.exponential(1/(gamma1*Num_of_N_objects +
2*gamma1*Num_of_M_objects + 0.1)), 5)
            new_obj = [total_num, Objects_arr[0][-2], t_life,
round(Objects_arr[0][-2]+t_life,5), 'N']
            Total_N_objects += 1
            #добавление предков во 2 таблицу

            table2[Objects_arr[0][0] - 1][-2] = (new_obj[0])
            #table2[Objects_arr[0][0] - 1].append(-1)

```



```

delete_object()
add_to_arr_objects(new_obj)

table1.write(str(Total_events)+' '+str(new_obj[1])+' '+Type+'
'+str(new_obj[2])+' - ('+ str(Num_of_N_objects)+' '+str(Num_of_M_objects)+'
')

table1.write(str(round(Objects_arr[0][-2] - new_obj[1],5)) +'
'+str(Objects_arr[0][0])+' '+str(Objects_arr[0][-1])+' ')
Total_spent = round(Objects_arr[0][-2] - new_obj[1],5)
Total_time = new_obj[1]

table2.append([total_num, new_obj[-1], new_obj[1], new_obj[2],
new_obj[3], -1, -1])

elif(pn1 < p <= pn1+pn2):
    Type = 'S_n(2)'
    Arr_of_types[1] += 1
    first_t_life =
round(np.random.exponential(1/(gamma1*Num_of_N_objects +
2*gamma1*Num_of_M_objects + 0.1)), 5)
    second_t_life =
round(np.random.exponential(1/(gamma1*Num_of_N_objects +
2*gamma1*Num_of_M_objects + 0.1)), 5)
    Total_N_objects += 2
    if(second_t_life < first_t_life):
        first_t_life, second_t_life = second_t_life, first_t_life

    total_num += 1
    first_obj = [total_num, Objects_arr[0][-2], first_t_life,
round(Objects_arr[0][-2]+first_t_life,5), 'N']
    total_num += 1
    second_obj = [total_num, Objects_arr[0][-2], second_t_life,
round(Objects_arr[0][-2] + second_t_life,5), 'N']

    table2[Objects_arr[0][0] - 1][-2] = (first_obj[0])
    table2[Objects_arr[0][0] - 1][-1] = (second_obj[0])

delete_object()
add_to_arr_objects(first_obj)
add_to_arr_objects(second_obj)

table1.write(str(Total_events)+' '+str(first_obj[1])+' '+Type+'
'+str(first_obj[2])+' '+str(second_obj[2])+'
('+str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ')
table1.write(str(round(Objects_arr[0][-2] - first_obj[1],5))+'
'+str(Objects_arr[0][0])+' '+str(Objects_arr[0][-1])+' ')
Total_spent = round(Objects_arr[0][-2] - first_obj[1],5)
Total_time = first_obj[1]

```

```

        table2.append([total_num - 1, first_obj[-1], first_obj[1],
first_obj[2], first_obj[3], -1, -1])
        table2.append([total_num, second_obj[-1],
second_obj[1],second_obj[2],second_obj[3],-1,-1])
    else:
        Type = 'S_n(3)'
        Arr_of_types[2] += 1
        first_t_life =
[round(np.random.exponential(1/(gamma1*Num_of_N_objects +
2*gamma1*Num_of_M_objects + 0.1)), 5), 'N']
        second_t_life =
[round(np.random.exponential(1/(3*gamma2*Num_of_N_objects+gamma2*Num_of_M_obj
ects)),5), 'M']
        if(second_t_life[0] < first_t_life[0]):
            first_t_life, second_t_life = second_t_life, first_t_life

        Total_N_objects += 1
        Total_M_objects += 1

        total_num += 1
        first_obj = [total_num, Objects_arr[0][-2], first_t_life[0],
round(Objects_arr[0][-2]+first_t_life[0],5), first_t_life[1]]
        total_num += 1
        second_obj = [total_num, Objects_arr[0][-2], second_t_life[0],
round(Objects_arr[0][-2] + second_t_life[0],5), second_t_life[1]]

        table2[Objects_arr[0][0] - 1][-2] = (first_obj[0])
        table2[Objects_arr[0][0] - 1][-1] = (second_obj[0])

        delete_object()
        add_to_arr_objects(first_obj)
        add_to_arr_objects(second_obj)

        table1.write(str(Total_events)+' '+str(first_obj[1])+' '+Type+'
'+str(first_obj[2])+' '+str(second_obj[2])+'
'+str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ')
        table1.write(str(round(Objects_arr[0][-2] - first_obj[1],5))+'
'+str(Objects_arr[0][0])+' '+str(Objects_arr[0][-1])+' ')
        Total_spent = round(Objects_arr[0][-2] - first_obj[1],5)
        Total_time = first_obj[1]

        table2.append([total_num - 1, first_obj[-1], first_obj[1],
first_obj[2], first_obj[3], -1, -1])
        table2.append([total_num, second_obj[-1],
second_obj[1],second_obj[2],second_obj[3],-1,-1])
    else:
        if(0 <= p <= pm1):
            Type = 'S_m(1)'
            Arr_of_types[4] += 1
            total_num+=1

```

```

        t_life =
round(np.random.exponential(1/(3*gamma2*Num_of_N_objects+gamma2*Num_of_M_obje
cts)), 5)

        Total_M_objects += 1
        new_obj = [total_num, Objects_arr[0][-2], t_life,
round(Objects_arr[0][-2]+t_life,5), 'M']

        table2[Objects_arr[0][0] - 1][-2] = (new_obj[0])

        delete_object()
        add_to_arr_objects(new_obj)

        table1.write(str(Total_events)+' '+str(new_obj[1])+' '+ Type +'
'+str(new_obj[2])+' - ('+ str(Num_of_N_objects)+' '+str(Num_of_M_objects)+'
')
        table1.write(str(round(Objects_arr[0][-2] - new_obj[1],5)) +'
'+str(Objects_arr[0][0])+' '+str(Objects_arr[0][-1])+' ')
        Total_spent = round(Objects_arr[0][-2] - new_obj[1],5)
        Total_time = new_obj[1]

        table2.append([total_num, new_obj[-1], new_obj[1], new_obj[2],
new_obj[3], -1, -1])
    else:
        Type = 'S_m(0)'
        Arr_of_types[3] += 1

        table1.write(str(Total_events)+' '+str(Objects_arr[0][3])+'
'+Type+' '+str(-1)+' '+str(-1)+'
'+str(Num_of_N_objects)+' '+str(Num_of_M_objects - 1)+' ')
        table1.write(str(round(Objects_arr[1][-2] -
Objects_arr[0][3],5))+' '+str(Objects_arr[1][0])+' '+str(Objects_arr[1][-
1])+' ')
        Total_spent = round(Objects_arr[1][-2] - Objects_arr[0][3],5)
        Total_time = Objects_arr[0][3]

        delete_object()

    if(table5.get('(' +str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ') !=
None):
        table5['(' +str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ')[0] +=
1
        table5['(' +str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ')[1] =
round(table5['(' +str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ')[1]+Tota
l_spent,5)
    else:
        table5['(' +str(Num_of_N_objects)+' '+str(Num_of_M_objects)+' ') =
[1,Total_spent]

table1.close()

```

```

file2 = open('Data/table2.txt','w')
for item in table2:
    for i in item:
        file2.write(str(i)+' ')
file2.close()

#####

print(Arr_of_types)
table3 = open('Data/table3.txt','w')
for item in Arr_of_types:
    table3.write(str(item)+ ' ')
table3.close()

table4 = open('Data/table4.txt','w')
table4.write(str(Total_N_objects)+' '+str(Num_of_N_objects)+' '+str(Total_M_objects)+' '+str(Num_of_M_objects))
table4.close()

for i in table5:
    table5[i].append(round(table5[i][-1] / Total_time,5))
    table5[i].append(table5[i][0]/100)

file5 = open('Data/table5.txt','w')
for key, value in table5.items():
    file5.write(str(key)+' '+str(value[0])+' '+str(value[1])+' '+str(value[2])+' '+str(value[3])+' ')
file5.close()
print('Total_time: ', Total_time)

```