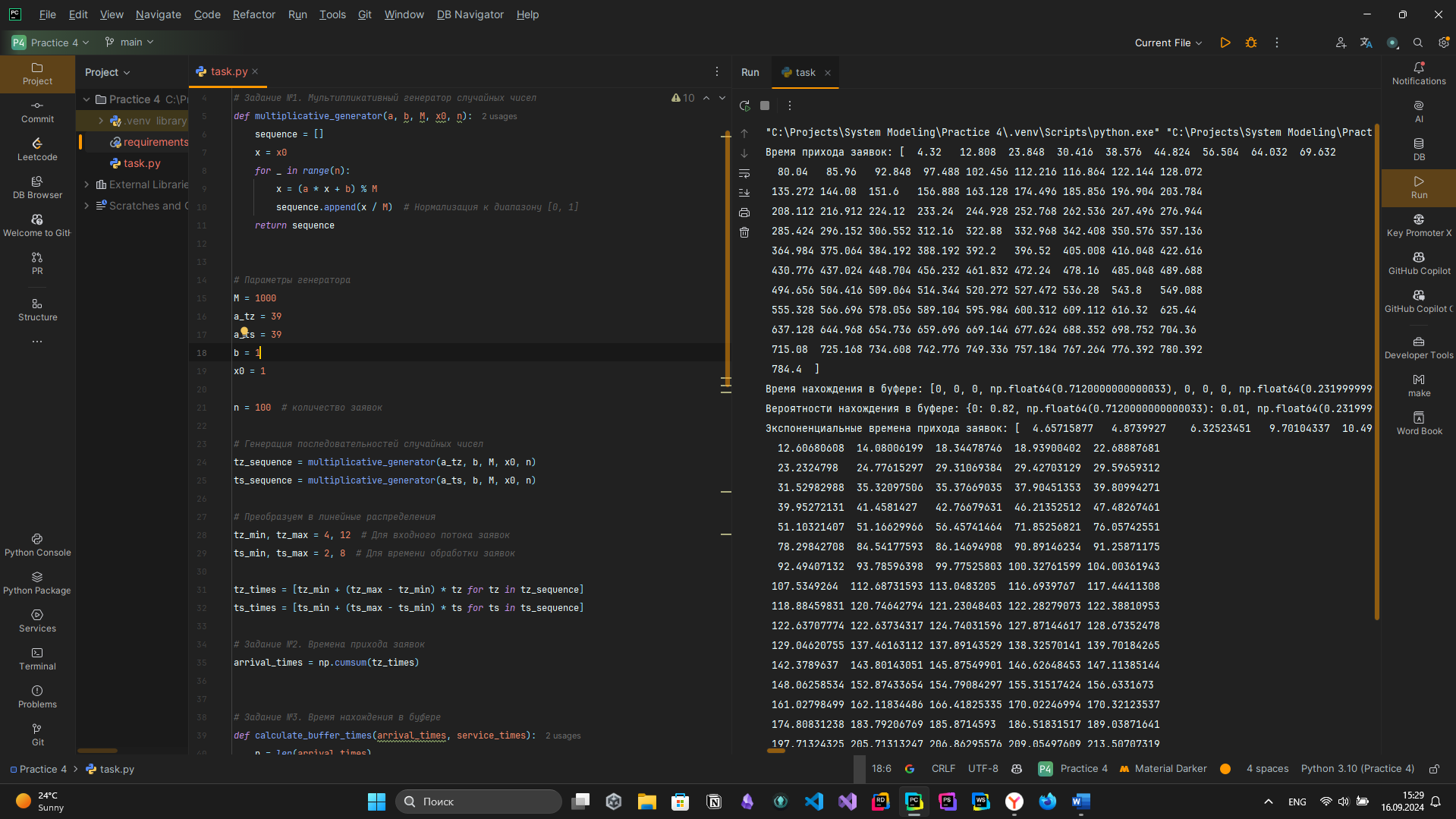
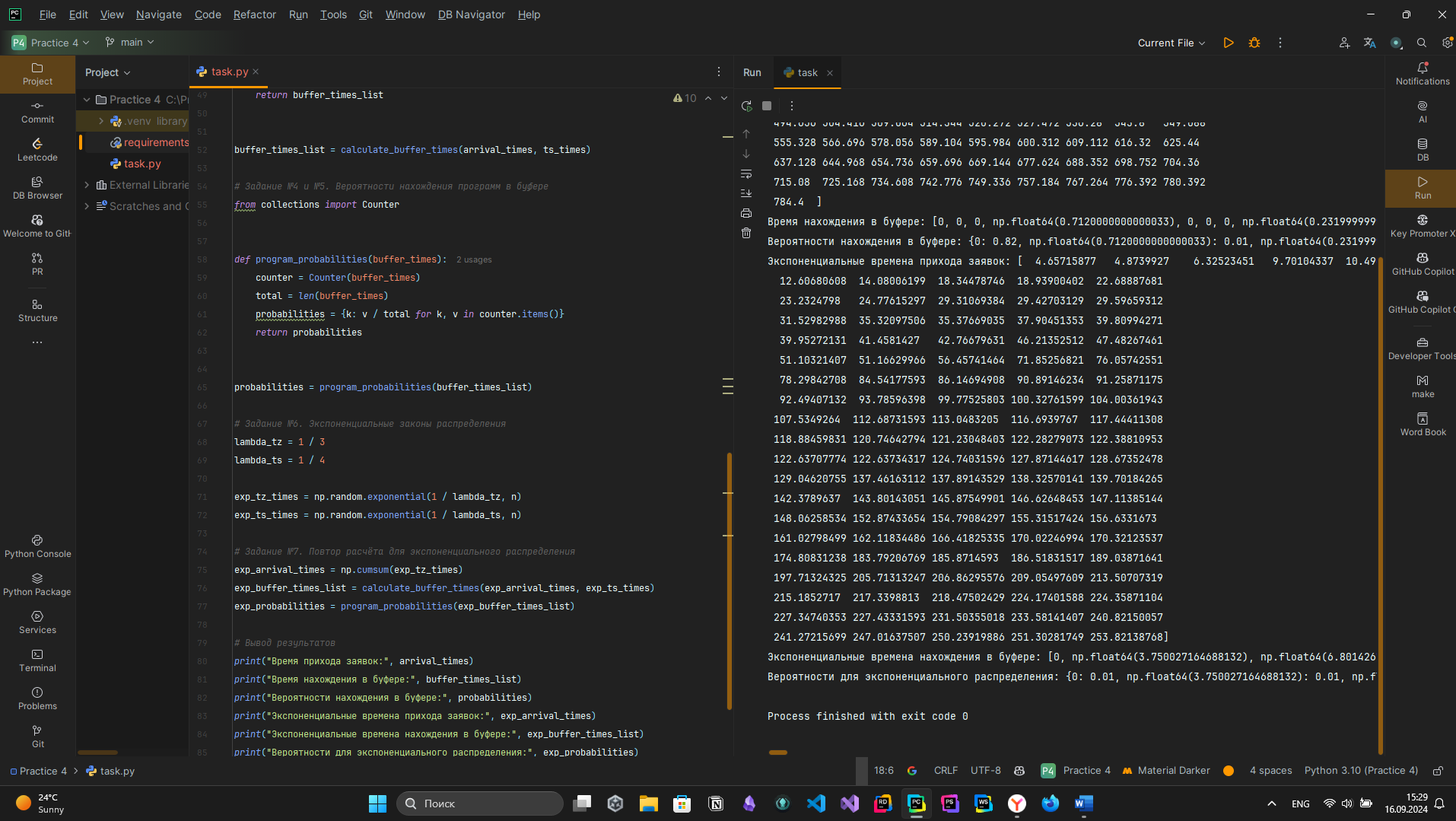
Практика 4

БСБО-09-23

Шутов Кирилл Сергеевич

Пример работы





Листинг кода

|  |
| --- |
| **import** numpy **as** np   *# Задание №1. Мультипликативный генератор случайных чисел* **def** multiplicative\_generator(a, b, M, x0, n):  sequence = []  x = x0  **for** \_ **in** range(n):  x = (a \* x + b) % M  sequence.append(x / M) *# Нормализация к диапазону [0, 1]* **return** sequence   *# Параметры генератора* M = 1000 a\_tz = 39 a\_ts = 39 b = 1 x0 = 1  n = 100 *# количество заявок  # Генерация последовательностей случайных чисел* tz\_sequence = multiplicative\_generator(a\_tz, b, M, x0, n) ts\_sequence = multiplicative\_generator(a\_ts, b, M, x0, n)  *# Преобразуем в линейные распределения* tz\_min, tz\_max = 4, 12 *# Для входного потока заявок* ts\_min, ts\_max = 2, 8 *# Для времени обработки заявок* tz\_times = [tz\_min + (tz\_max - tz\_min) \* tz **for** tz **in** tz\_sequence] ts\_times = [ts\_min + (ts\_max - ts\_min) \* ts **for** ts **in** ts\_sequence]  *# Задание №2. Времена прихода заявок* arrival\_times = np.cumsum(tz\_times)   *# Задание №3. Время нахождения в буфере* **def** calculate\_buffer\_times(arrival\_times, service\_times):  n = len(arrival\_times)  buffer\_times\_list = [0] \* n  finish\_time = 0   **for** i **in** range(n):  **if** arrival\_times[i] < finish\_time:  buffer\_times\_list[i] = finish\_time - arrival\_times[i]  finish\_time = max(arrival\_times[i], finish\_time) + service\_times[i]   **return** buffer\_times\_list   buffer\_times\_list = calculate\_buffer\_times(arrival\_times, ts\_times)  *# Задание №4 и №5. Вероятности нахождения программ в буфере* **from** collections **import** Counter   **def** program\_probabilities(buffer\_times):  counter = Counter(buffer\_times)  total = len(buffer\_times)  probabilities = {k: v / total **for** k, v **in** counter.items()}  **return** probabilities   probabilities = program\_probabilities(buffer\_times\_list)  *# Задание №6. Экспоненциальные законы распределения* lambda\_tz = 1 / 3 lambda\_ts = 1 / 4  exp\_tz\_times = np.random.exponential(1 / lambda\_tz, n) exp\_ts\_times = np.random.exponential(1 / lambda\_ts, n)  *# Задание №7. Повтор расчёта для экспоненциального распределения* exp\_arrival\_times = np.cumsum(exp\_tz\_times) exp\_buffer\_times\_list = calculate\_buffer\_times(exp\_arrival\_times, exp\_ts\_times) exp\_probabilities = program\_probabilities(exp\_buffer\_times\_list)  *# Вывод результатов* print(**"Время прихода заявок:"**, arrival\_times) print(**"Время нахождения в буфере:"**, buffer\_times\_list) print(**"Вероятности нахождения в буфере:"**, probabilities) print(**"Экспоненциальные времена прихода заявок:"**, exp\_arrival\_times) print(**"Экспоненциальные времена нахождения в буфере:"**, exp\_buffer\_times\_list) print(**"Вероятности для экспоненциального распределения:"**, exp\_probabilities) |