Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

З лабораторної роботи №2 з дисципліни

«Основи програмування 2. Модульне програмування»  
  
«Файли даних. Бінарні файли»  
  
Варіант 26

Виконав студент ІП-14 Саіян Артур Михайлович

(шифр, прізвище, ім'я, по батькові)

Перевірив Вітковська Ірина Іванівна

(прізвище, ім'я, по батькові)

Київ 2022

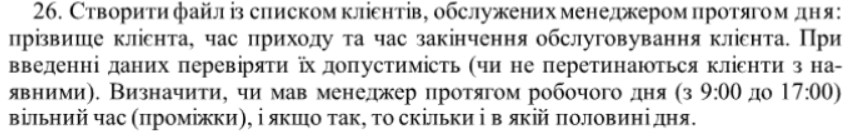
**Лабораторна робота №2**

**Тема роботи:** Бінарні файли

**Мета роботи:** Вивчити особливості створення й обробки бінарних файлів даних.

**Варіант:** 26

**Завдання:**

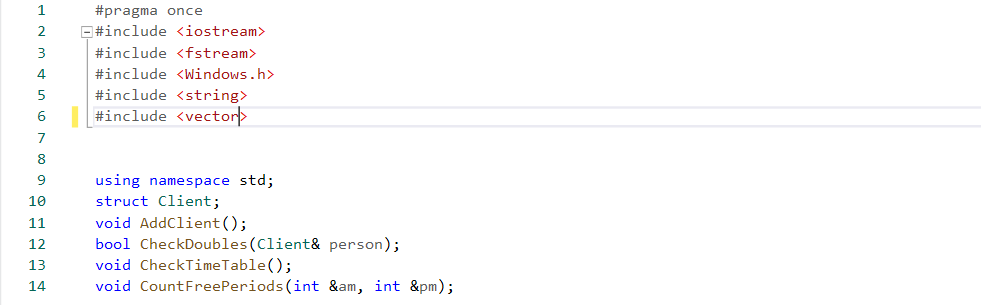


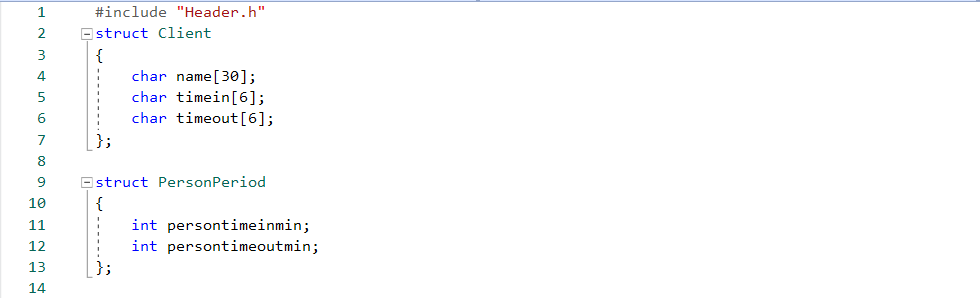
*Розв'язання*

**Постановка задачі:**

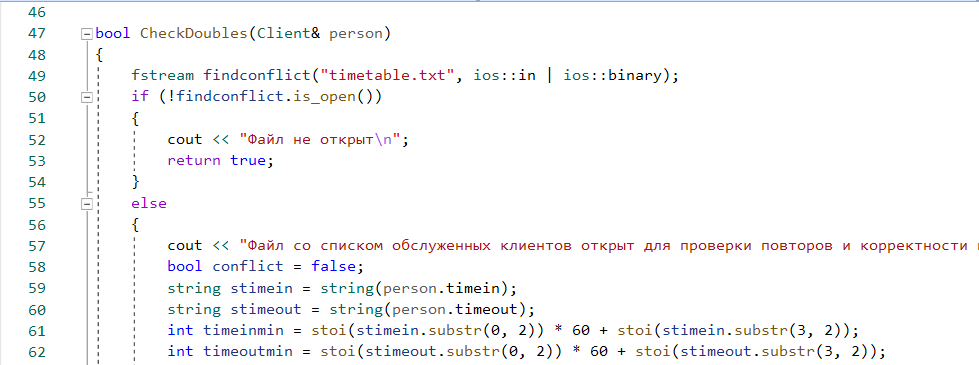
Опишемо структуру "Клієнт", яка буде містити такі поля: ім'я, тип рядок; час приходу, тип рядок; та час закінчення обслуговування клієнта, тип рядок. Опишемо функцію, яка буде створювати екземпляр структури та надавати можливість заповнювати його поля користувачем, а потім записувати у бінарний файл "*timetable.txt*" після перевірки на відсутність перетину часу з вже обслугованими клієнтами та коректність часу. Для перевірки опишемо функцію, яка буде послідовно зчитувати дані з бінарного файлу "*timetable.txt*" у інший екземпляр структури "Клієнт" для порівняння к наявним, а саме атрибутів, які відповідають за час приходу та закінчення обслуговування. У випадку коректності часу та відсутності конфліктів структура, яка була ініціалізована користувачем буде записана у файл "*timetable.txt*". Опишемо функцію для виводу списку клієнтів, записаних у бінарний файл "*timetable.txt*". Після закінчення вводу обслугованих клієнтів, порахуємо кількість вільних проміжків. Для цього попарно зчитаємо у структури "Клієнт" весь файл, та порівняємо час закінчення обслуговування першого клієнта з часом початку обслуговування другого. Якщо цей час не співпадає, наявний вільний проміжок. Якщо цей проміжок мав місце до 12:00 записуємо до кількості проміжків першої половини дня, якщо після 12:00 аналогічно до кількості проміжків другої половини дня.

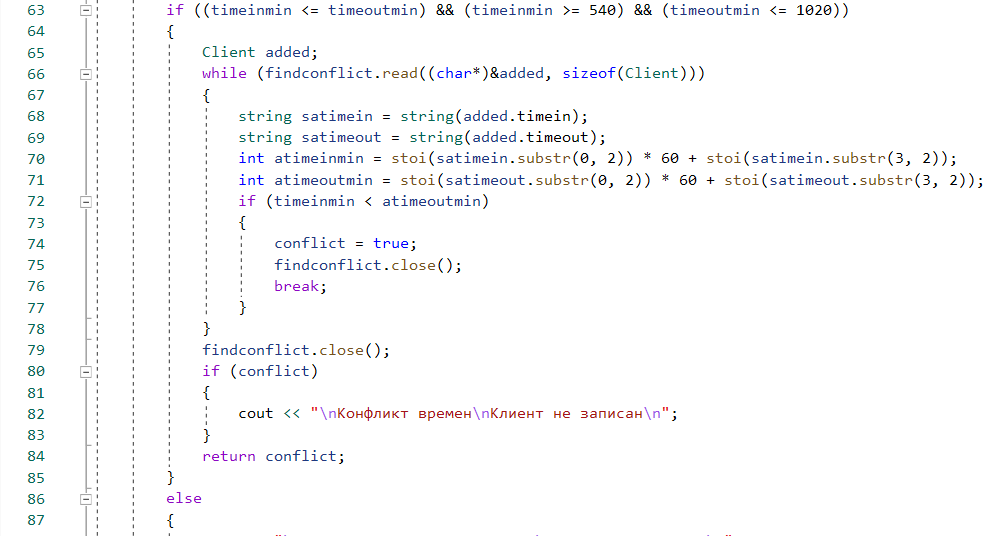
*Код на C++ та тестування:*

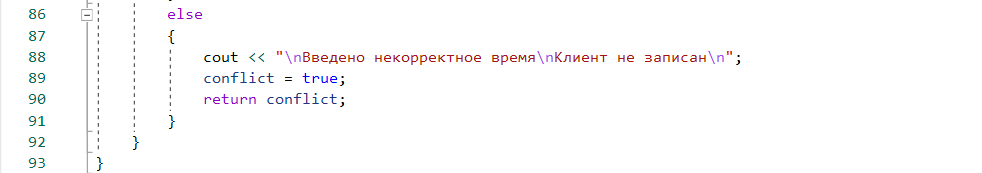


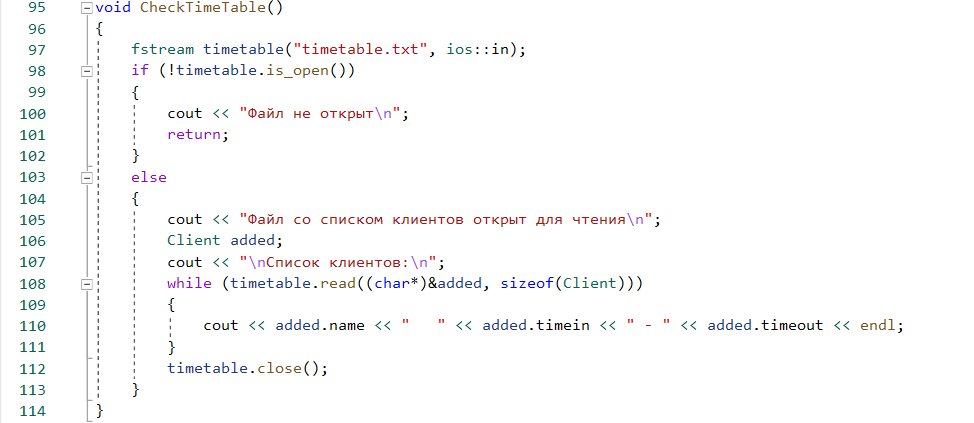


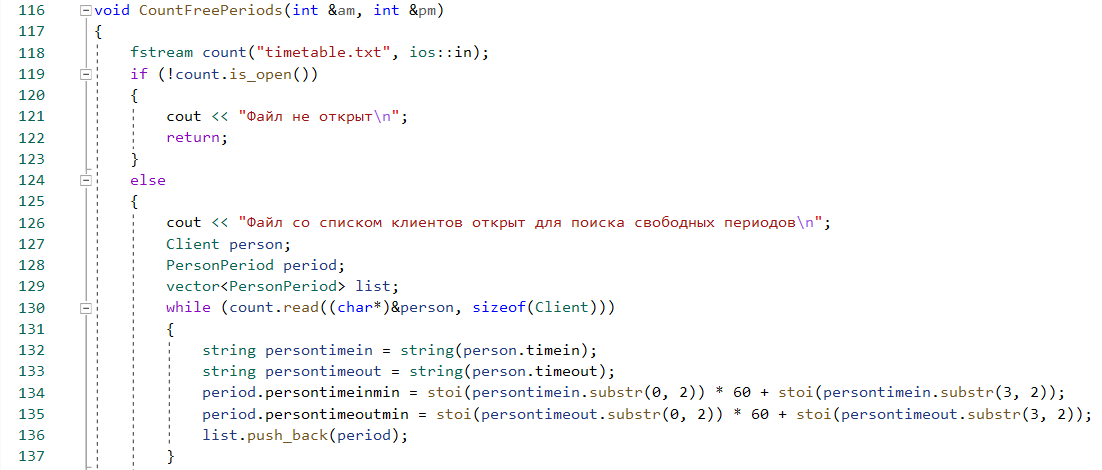


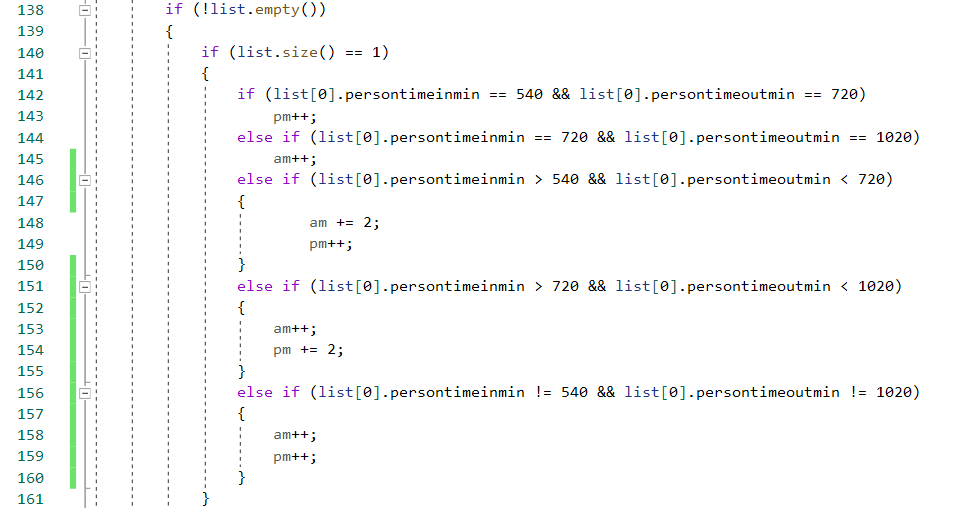


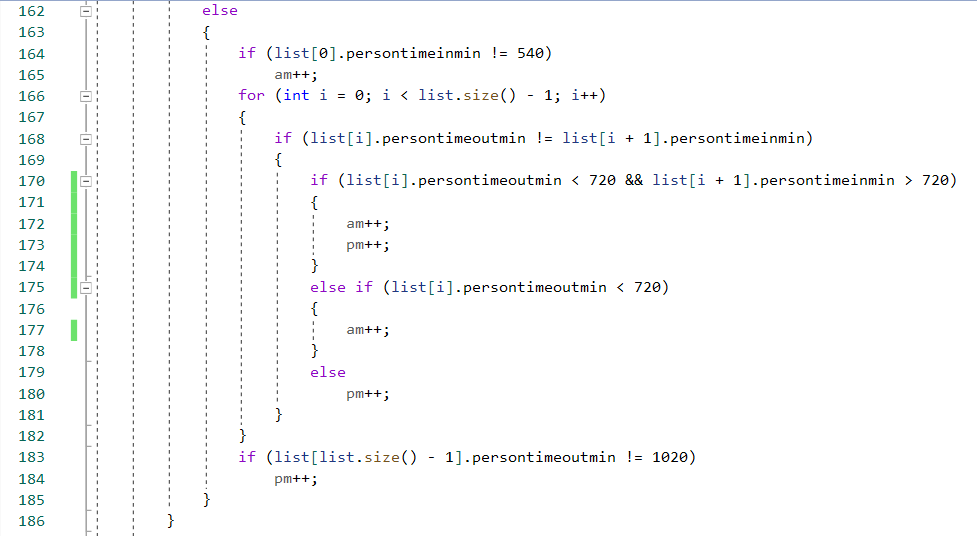


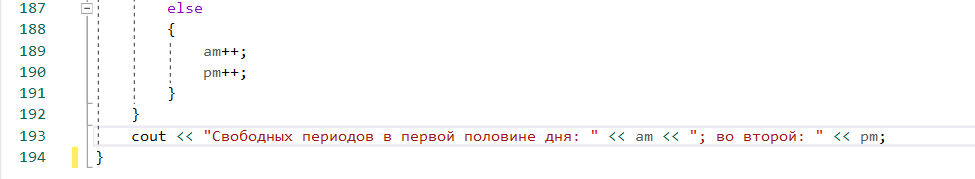




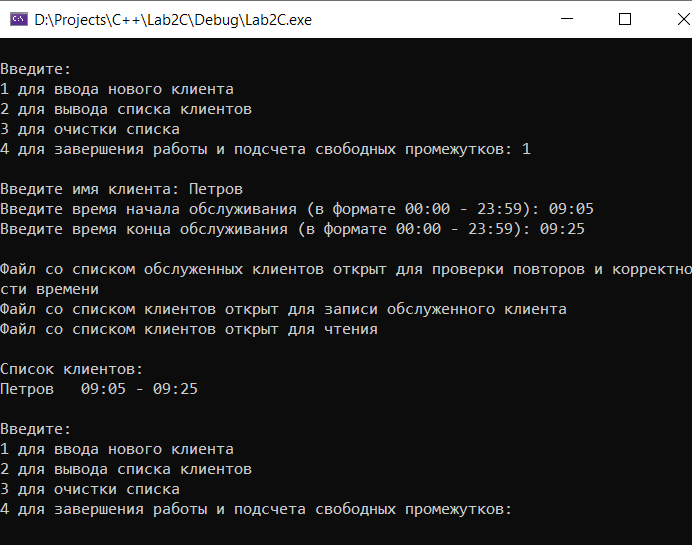


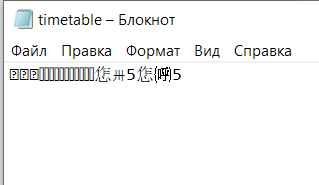
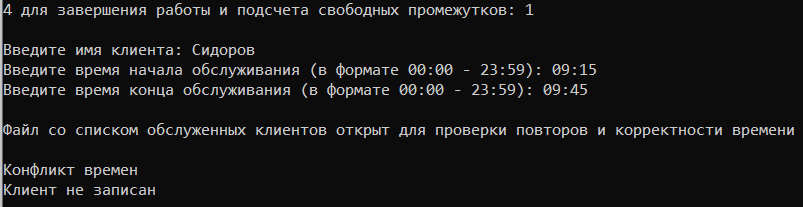


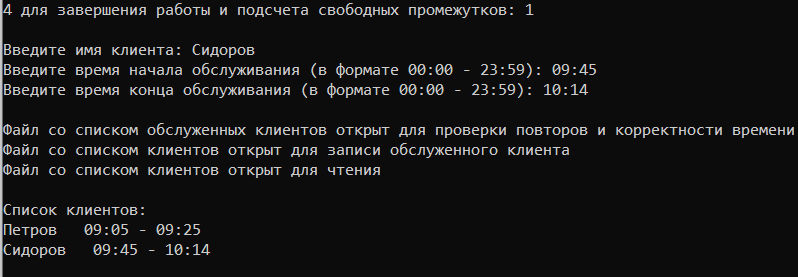


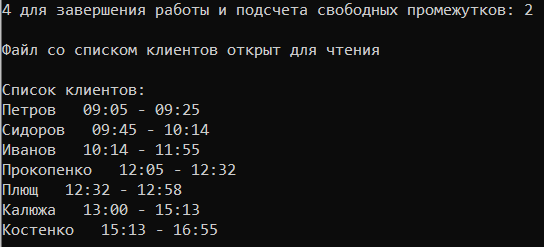
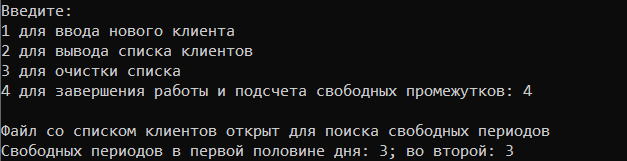




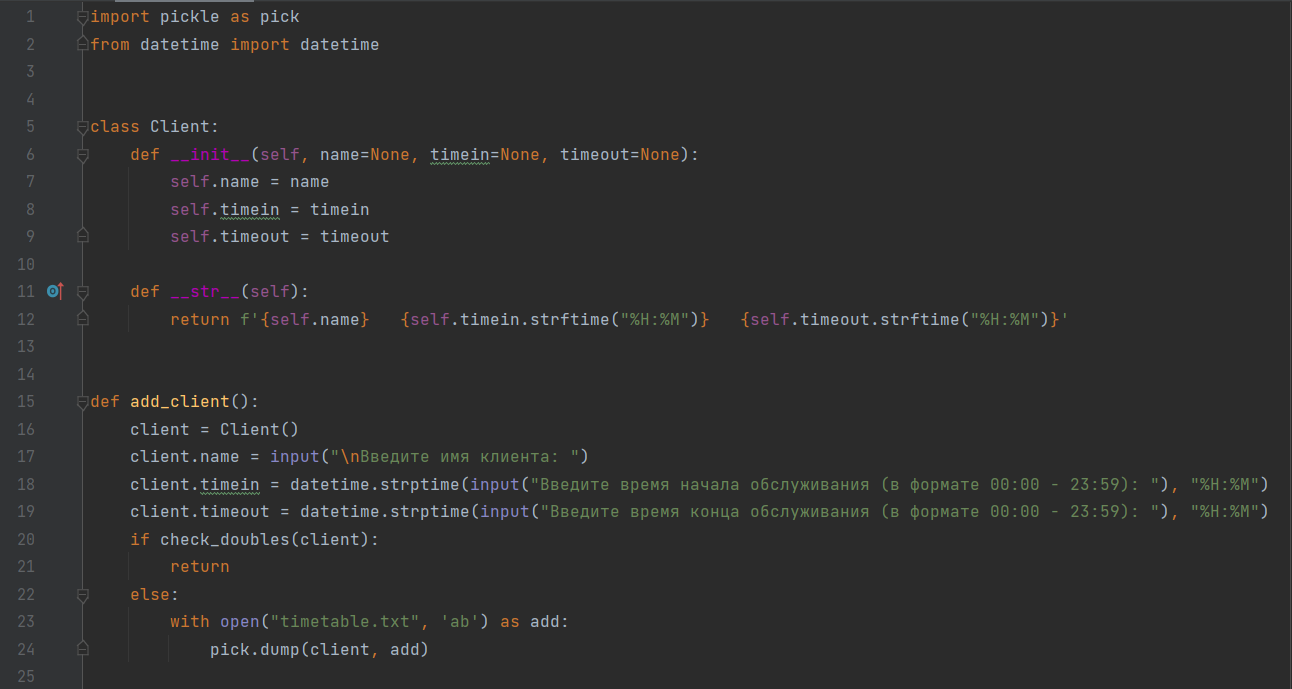


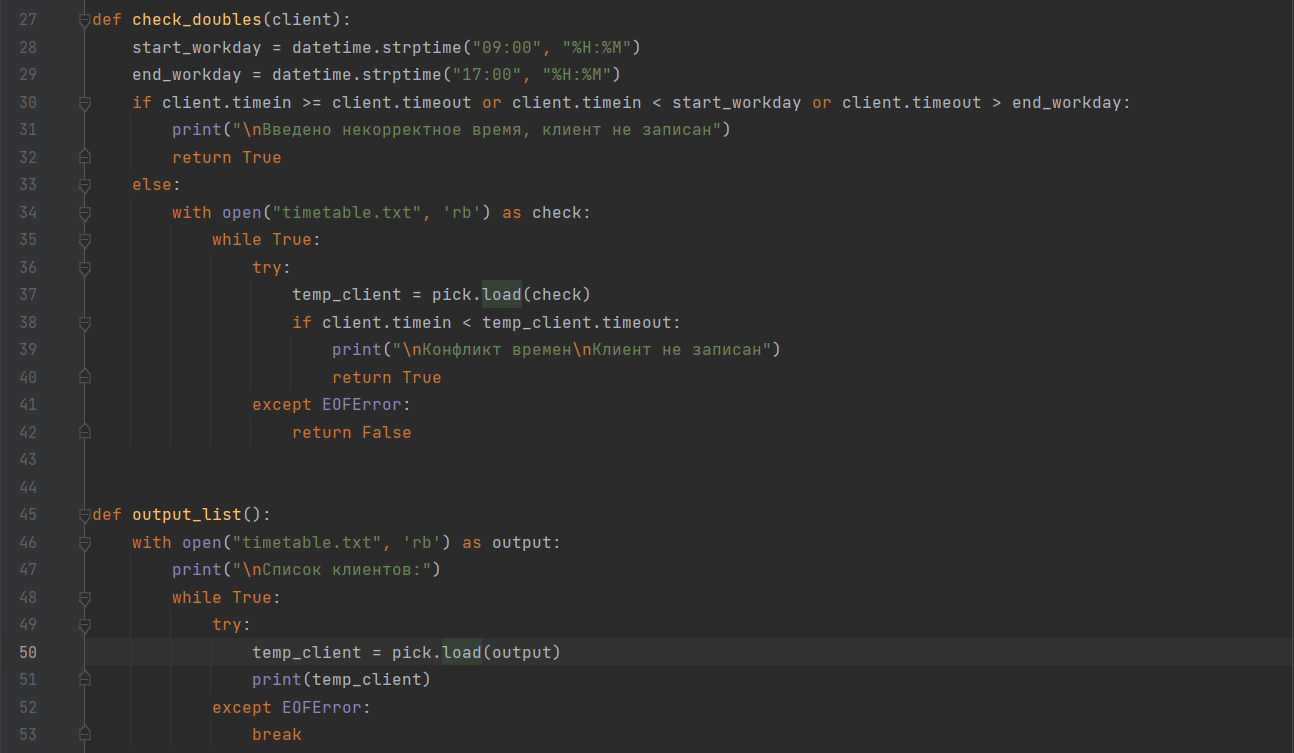
 

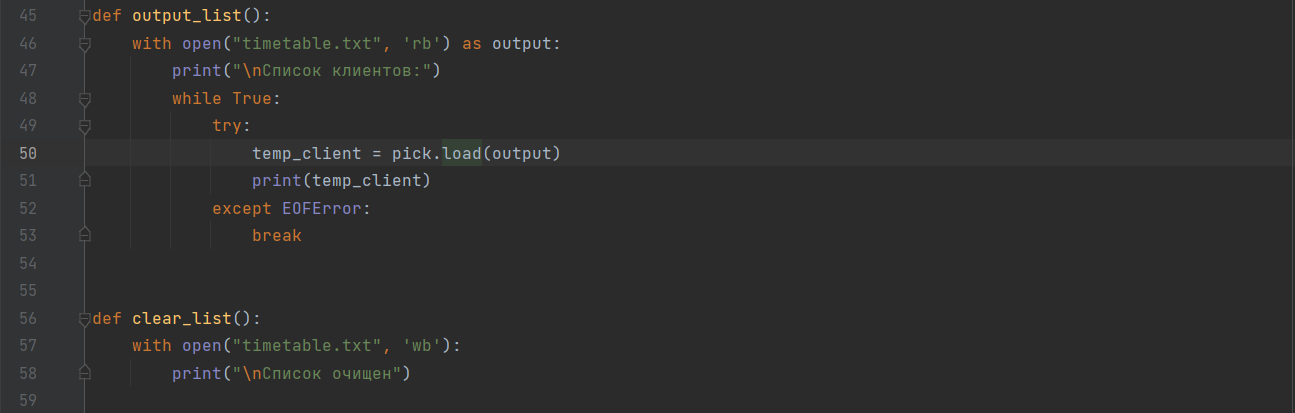


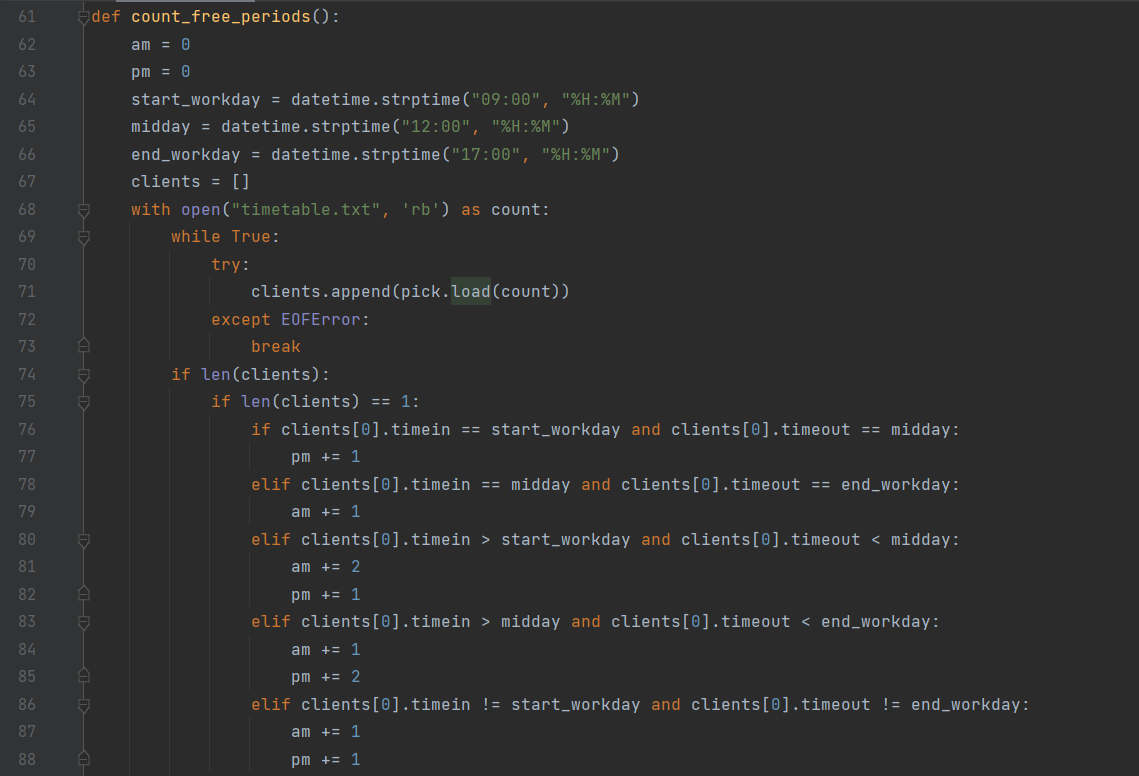
 

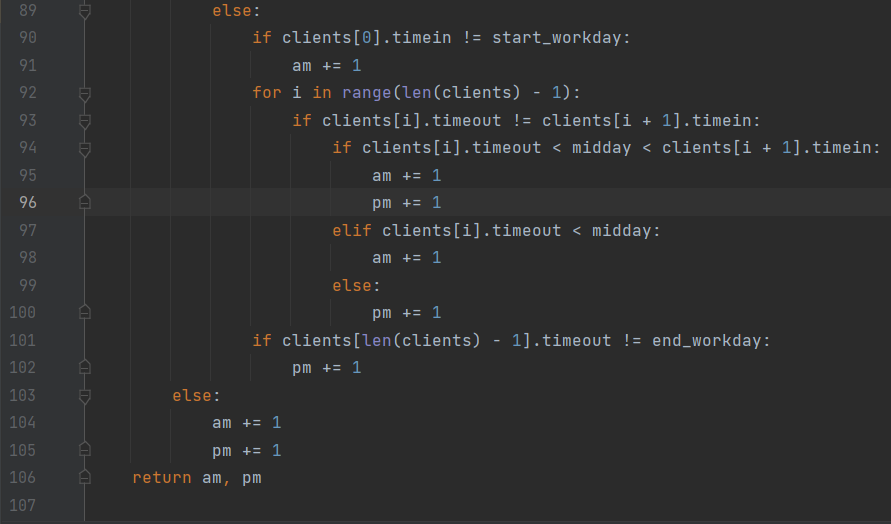
*Код на Python та тестування:*

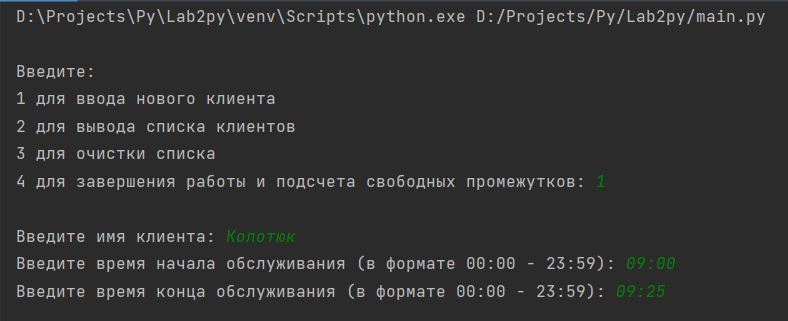


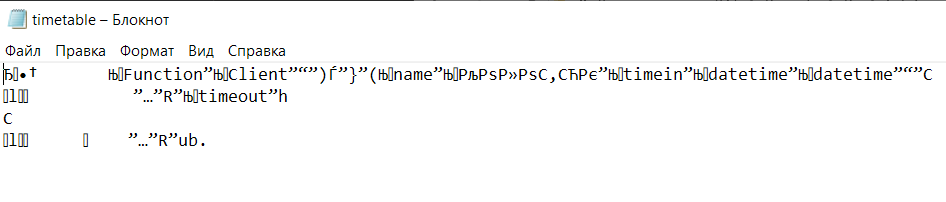


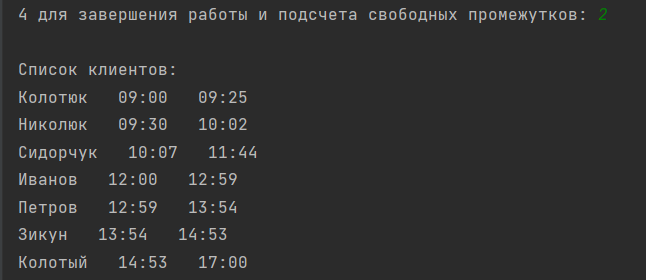




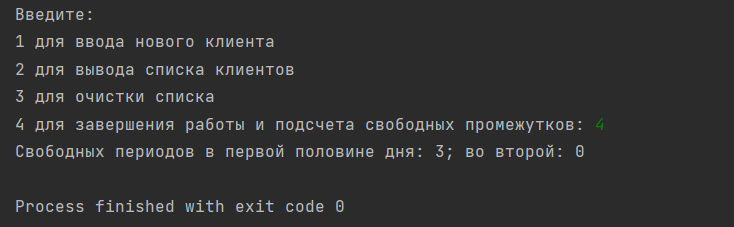












**Висновок:** Під час виконання лабораторної роботи я спроєктував програму, яка за допомогою потоків даних створює та записує дані у бінарний файл за допомогою парсінгу структур. На основі цього функционалу створений список вже обслугованих за день клієнтів. За допомогою описаних функцій була реалізована можливість перевірки конфликтів даних, рахування вільних проміжків, вивід клієнтів, записаних у список, а також очистка списку. На прикладі цього завдання вивчив особливості створювання, запису й обробки бінарних файлів; дослідив різницю між типами потоків та їх модифікаторами, основні з яких: append, trunc, in, out, binary.