

**Министерство образования и науки Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**  
**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**ФАКУЛЬТЕТ БЕЗОПАСНЫХ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**Лабораторная работа №1**  
**«Обработка и тарификация CDR (Call Detail Record)»**  
**Вариант-9**

Работу выполнил:  
студент группы N3347  
очного отделения  
Звягинцев Н.Ю.  
Дата защиты: 04.04.2020



Проверил:  
Таранов С.В.

Санкт-Петербург  
2020

Цель: обработать файл CDR и на его основании реализовать правило тарификации для услуг типа «Телефония» по длительности разговора и «СМС» по общему количеству.

Задание: Протарифицировать абонента с номером 933156729 с коэффициентом к:  
2руб/минута исходящие звонки, но 20 минут бесплатно, 0руб/минута входящие, смс -  
2руб/шт.

## Ход работы

Для реализации лабораторной работы мною был выбран язык программирования Python 3. Выбрал его по той причине, что изначально в Python есть удобный модуль, который позволяет быстро и удобно распарсить файл CDR формата .csv. Сам модуль тарификации я реализовал отдельным классом с параметрами, для того, чтобы в дальнейшем можно было пользоваться функцией тарификации при других условиях тарифа.

Листинг кода:

```
...
    Протарифицировать абонента с номером 933156729 с коэффициентом к: 2руб/минута
    исходящие звонки, но 20 минут бесплатно,
    0руб/минута входящие,
    смс - 2руб/шт
    timestamp - время звонка
    msisdn_origin - кто совершил звонок
    msisdn_dest - кому звонили
    call_duration - длительность звонка в минутах
    sms_number - количество отправленных смс для абонента msisdn_origin
...

import csv

filepath = r"D:\Downloads\data (2).csv"
timestamp = []
msisdn_origin = []
msisdn_dest = []
call_duration = []
sms_number = []

with open(filepath, "r", newline="") as file:
    reader = csv.reader(file) #чтение файла csv

    for row in reader:
        list0 = row[0].split(',')
        list1 = row[1].split(',')
        list2 = row[2].split(',')
        list3 = row[3].split(',')
```

```

list4 = row[4].split(',')

timestamp.extend(list0)
msisdn_origin.extend(list1)
msisdn_dest.extend(list2)
call_duration.extend(list3)
sms_number.extend(list4)

bill_for_income_calls = 0
bill_for_out_calls = 0
bill_for_sms = 0
total = 0

class Tarification:
    price_income_calls=2
    minutes_for_free=20
    price_out_calls =0
    price_sms=2

    def tarification(self):
        sum_income_call = 0
        sum_out_call = 0
        sms = 0

        for i in range(len(msisdn_origin)):
            if msisdn_origin[i] == '933156729':
                sum_income_call = sum_income_call + float(call_duration[i])
                sms = float(sms_number[i])

        for j in range(len(msisdn_dest)):
            if msisdn_dest[j] == '933156729':
                sum_out_call = sum_out_call + float(call_duration[j])

        if sum_income_call > 20:
            bill_for_income_calls = (sum_income_call - self.minutes_for_free) *
self.price_income_calls
        else:
            bill_for_income_calls = 0
        print("Bill for incoming calls: ", bill_for_income_calls)
        bill_for_out_calls = sum_out_call * self.price_out_calls
        print("Bill for outcoming calls: ", bill_for_out_calls)

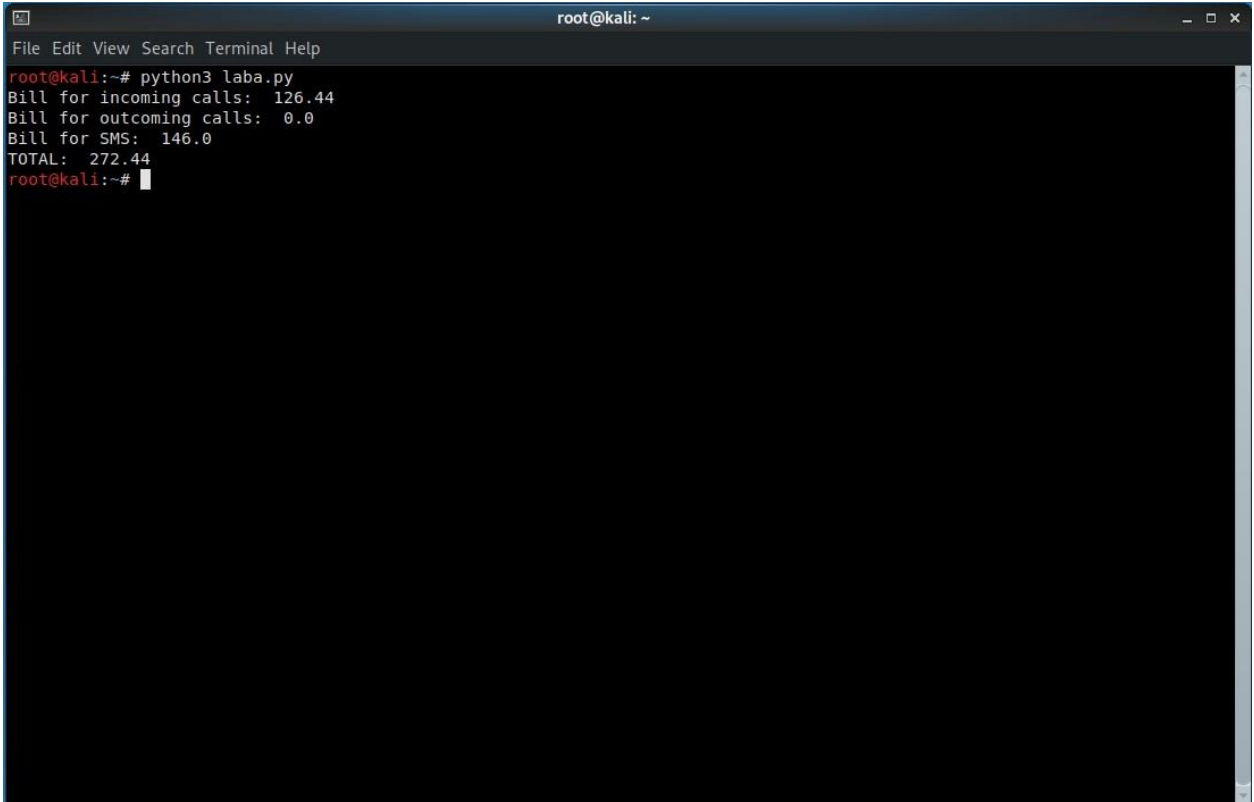
        bill_for_sms = sms * self.price_sms
        print("Bill for SMS: ", bill_for_sms)

        total = bill_for_income_calls + bill_for_out_calls + bill_for_sms
        ('TOTAL: ', total)

tarif=Tarification()
tarif.tarification()

```

Результат выполнения программы:

A screenshot of a terminal window titled 'root@kali: ~'. The terminal shows the execution of a Python script named 'laba.py'. The output of the script is displayed in red text: 'Bill for incoming calls: 126.44', 'Bill for outgoing calls: 0.0', 'Bill for SMS: 146.0', and 'TOTAL: 272.44'. The prompt 'root@kali:~#' is visible at the end of the output.

```
root@kali:~# python3 laba.py
Bill for incoming calls: 126.44
Bill for outgoing calls: 0.0
Bill for SMS: 146.0
TOTAL: 272.44
root@kali:~#
```

На скриншоте видно, что программа выводит в консоль счета за входящие, исходящие звонки, смс и отдельно суммарный счет.

Вывод: в ходе лабораторной работы я узнал основные принципы работы биллинговых систем и правила тарификации абонентов, так же я научился обработать файлы CDR формата .csv и на его основании реализовал правило тарификации для услуг типа «Телефония» по длительности разговора и “СМС” по общему количеству.