**Министерство образования и науки Российской Федерации**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Факультет Безопасности информационных технологий**

**Название предмета: Управление мобильными устройствами**

**ОТЧЕТ**

**по лабораторной работе**

**Название лабораторной работы: «Обработка и тарификация CDR (Call Detail Record)»**

**Обучающийся: Пищаева Анастасия Игоревна**

**Преподаватель: Федоров Иван Романович**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(подпись преподавателя)**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**(подпись)**

**Дата \_\_\_\_\_\_\_\_\_**

**Санкт-Петербург**

**2020**

Цель работы: реализовать простейшее правило тарификации для услуг типа “Телефония” по длительности разговора и “СМС” по общему количеству.

Правила тарификации услуг “Телефония”:

X = T \* k,

где X - итоговая стоимость всех звонков абонента, T - общая длительность звонков (сумма длительностей всех записей по абоненту в файле), k - множитель тарифного плана.

Правила тарификации услуг “СМС”:

Y = N \* k,

где Y - итоговая стоимость всех СМС абонента, N - общее количество СМС (сумма числа всех СМС в записях по абоненту в файле), k - множитель тарифного плана.  
  
Ход работы:

Вариант 12

1. Значение полей:

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

2.

import csv

path = 'data.csv'

number = '911926375'

k1 = 4

k2 = 1

y\_sms = 1

free\_sms = 5

data = []

sms = 0

outcome = 0.0

income = 0.0

total = 0.0

def read\_file (path):

with open(path) as csvfile:

reader = csv.DictReader(csvfile, delimiter=',', quotechar=',', quoting=csv.QUOTE\_MINIMAL)

for row in reader:

if row.get('msisdn\_origin') == number or row.get('msisdn\_dest') == number:

data.append(row)

return data

def count\_data (number):

global sms, income, outcome

for i in data:

if i.get('msisdn\_origin') == number:

sms += int(i.get('sms\_number'))

outcome += float(i.get('call\_duration'))

elif i.get('msisdn\_dest') == number:

income += float(i.get('call\_duration'))

data = read\_file(path)

count\_data (number)

if sms <= free\_sms: sms = 0

else: sms -= 5

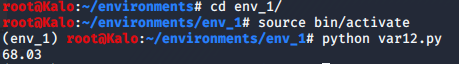
if income <= 5: income = 0

else: income -= 5

total += (outcome \* k1) + (income \* k2) + (sms \* y\_sms)

print (total)

3.



Вывод: с помощью парсера был протарифицирован номер, в соответсвии с вариантом, была найдена стоимость мобильны звонков и SMS-сообщений, которая равна 68.03 руб.