

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

ФАКУЛЬТЕТ БЕЗОПАСНОСТИ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

УПРАВЛЕНИЕ МОБИЛЬНЫМИ УСТРОЙСТВАМИ

Лабораторная работа №1

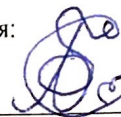
«Обработка и тарификация CDR (Call Detail Record)»

Работу выполнил

Студент группы № N3347

очного отделения:

Григорьев А.А.



Проверил

Дата защиты: 29.03.2020

Санкт-Петербург
2020

Цель:

Изучить процесс тарификации, реализовать простейшее правило тарификации для услуг типа «телефония»

«Обработка и тарификация CDR (Call Detail Record)»

При совершении звонка абонентом АТС формирует файлы с данными CDR (call detail records), которые загружаются в биллинг. Далее звонки тарифицируются. В процессе тарификации происходит начисление на операционные счета услуг, после чего выставляются платежные документы (биллинговые счета) и отправляются клиентам.

В данной работе необходимо реализовать простейшее правило тарификации для услуг типа “Телефония” по длительности разговора и “СМС” по общему количеству. Работа включает в себя 2 этапа:

1. Парсинг файла CDR и выборка нужных строк для обработки
2. Тарификация выбранных записей
- 3.

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 (у каждого варианта свой).

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 (у каждого варианта свой).

В качестве результата работы необходимо представить программный модуль для обработки CDR и тарификации абонента.

Средства реализации выбираются студентом самостоятельно. Рекомендуется делать модуль тарификации независимым от обработки, так как он пригодится в 2-й лабораторной работе (можно реализовать отдельным классом/функцией/процедурой).

Файл CDR:

<https://drive.google.com/file/d/1Sk5zwZwdUJCRdOgjOcBcLFBYyAUAYeIL/view?usp=sharing>

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

Вариант работы:

5. Протарифицировать абонента с номером 915642913 с коэффициентом k:
1руб/минута исходящие звонки,
1руб/минута входящие,
смс - первые 5шт бесплатно, далее 1руб/шт

Обоснование выбранного средства реализации:

В качестве средства реализации мной был выбран язык программирования Python. Передо мной стоял выбор между Python и С. Не смотря на надежность языка программирования си, он не обладает такой обширной библиотекой готовых инструментов, как Python. Требования к работе и ее объем позволили мне использовать Python.

Ход работы

Исходный код:

```
import csv

#filename = "D:\DEV\mobile\lab1\data.csv"
file = input("Enter path to CDR file: ")

sms_sum = 0
out_calls_sum = 0
inc_calls_sum = 0

#number = 915642913
number = input("Enter phone number: ")
sms_tariff = 1
#first 5 for free
out_calls_tariff = 1
inc_calls_tariff = 1

# initializing the titles and rows
fields = []
rows = []
# reading csv file
with open(file, 'r') as csvfile:
    # creating a csv reader object
    csvreader = csv.reader(csvfile)
    # extracting field names through first row
    fields = next(csvreader)
    # extracting each data row one by one
    for row in csvreader:
        rows.append(row)

# get total number of rows
print("Total no. of lines in a CDR file: %d"%(csvreader.line_num))

# calculating SMS tariffication
for row in rows[:csvreader.line_num]:
    if number in row[1]:
        sms_sum += float(row[4])
sms_price = sms_sum*sms_tariff-5
if sms_price < 0:
    sms_price = 0
print(sms_price, "rubles for SMS")

# calculating outgoing calls tariffication
for row in rows[:10]:
    if number in row[1]:
        out_calls_sum += float(row[3])
out_calls_price = out_calls_sum*out_calls_tariff
if out_calls_price < 0:
```

```

    out_calls_price = 0
print(out_calls_price, "rubles for outgoing calls")

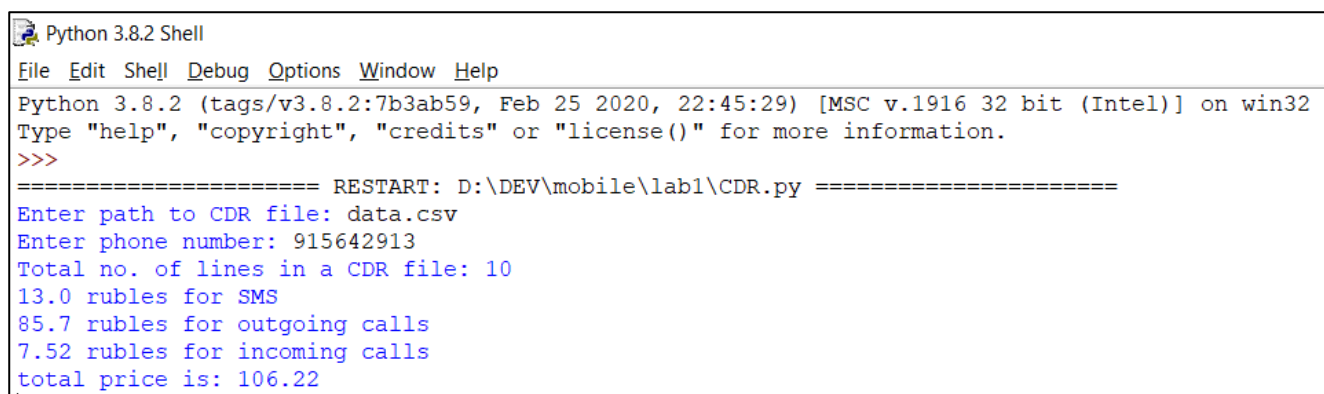
# calculating incoming calls tariffication
for row in rows[:10]:
    if number in row[2]:
        inc_calls_sum += float(row[3])
inc_calls_price = inc_calls_sum*inc_calls_tariff
if inc_calls_price < 0:
    inc_calls_price = 0
print(inc_calls_price, "rubles for incoming calls")

# total price
total_price = sms_price+out_calls_price+inc_calls_price
print("total price is:", total_price)

```

Результаты работы:

На рисунке 1 представлен результат выполнения программного модуля:



```

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\DEV\mobile\lab1\CDR.py =====
Enter path to CDR file: data.csv
Enter phone number: 915642913
Total no. of lines in a CDR file: 10
13.0 rubles for SMS
85.7 rubles for outgoing calls
7.52 rubles for incoming calls
total price is: 106.22

```

Рисунок 1 – результат выполнения программного модуля

При запуске программы необходимо ввести путь к файлу, содержащему записи о звонках и смс. Далее, необходимо ввести интересующий нас номер телефона. После выполнения этих действий будет произведена тарификация.

Выводы:

Цель работы достигнута, получены знания о тарификации услуг типа «Телефония». Создан программный модуль обработки CDR и тарификации абонента.