**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ,**

**СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА» (СПбГУТ)**

Факультет: Инфокоммуникационных сетей и систем Кафедра: Защищенных систем связи

Дисциплина: Защищенные операционные системы

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

Менеджер задач с приоритетами

*(тема отчета)*

Направление/специальность подготовки

10.03.01 Информационная безопасность

*(код и наименование направления/специальности)*

Студент:

Емельянов Н.Р. ИКБ-32

*(Ф.И.О., № группы) (подпись)*

Михайлов И.Д. ИКБ-32

*(Ф.И.О., № группы) (подпись)*

Преподаватель:

Асс. Смирнов Д.Н.

*(Должность, Ф.И.О. преподавателя) (подпись)*

**Цель**

Написать приложение-симулятор банка для управления финансами, запрашивающее у пользователя ввод и вывод средств с баланса

**Задача**

Выделим основные 5 задач:

1. Дать пользователю создать аккаунт
2. Дать пользователю возможность пополнить баланс
3. Дать пользователю возможность вывести средства в пределах баланса
4. Дать пользователю возможность проверки баланса
5. Дать пользователю возможность завершить работу

**Ход выполнения**

Создается класс Account для банковского аккаунта. Функции пополнения и снятия являются функциями класса. Используется метод input() для того, чтобы пользователь мог ввести данные. Считывается число, соответствующее целевому действию. Выполняется действие. Выводится итоговое значение при помощи print(). Повторяется до тех пор, пока пользователь не выйдет из программы.

**Листинг программы**

*import sys*

*class Account(object):*

*def \_\_init\_\_(self, balance):*

*self.balance = balance*

*def deposit(self, amount):*

*if amount > 0:*

*self.balance += amount*

*return "Баланс пополнен"*

*return "Пополнение не бывает отрицательным"*

*def withdraw(self, amount):*

*if self.balance >= amount:*

*self.balance -= amount*

*return "Успешно выведено"*

*return "Недостаточно средств на счете"*

*def check\_balance(self):*

*return self.balance*

*def create\_account():*

*bank\_acc = Account(0)*

*print(bank\_acc.balance)*

*bank\_acc = Account(0)*

*create\_account()*

*print('Вам создан банковский счет. Текущий баланс = 0')*

*while True:*

*print('Что будете делать?\n\n1 - пополнить баланс\n2 - вывести деньги со счета\n3 - проверить остаток средств\n0 - завершение работы')*

*selected = int(input())*

*if selected == 1:*

*print(bank\_acc.deposit(int(input('Введите сумму пополнения\n'))))*

*elif selected == 2:*

*print(bank\_acc.withdraw(int(input('Введите сумму вывода\n'))))*

*elif selected == 3:*

*print(bank\_acc.check\_balance())*

*elif selected == 0:*

*sys.exit()*

*else:*

*print('я тебя совсем не понял')*

**На рисунке 1 представлен вывод программы**

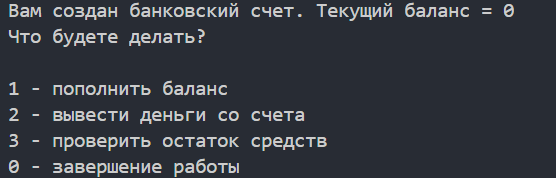
****

Рис. 1 – вывод программы

**Вывод**

В ходе выполнения лабораторной работы было написано приложение, которое запрашивает у пользователя ввод числа, обозначающее действие внутри приложения: пополнение баланса, вывод средств, вывод остатка средств на экран, выход из приложения. Приложение основано на концепции ООП.

В зависимости от выполнения условий вывели результат при помощи print()