

Московский государственный технический
университет им. Н.Э. Баумана

Факультет «Радиотехнический»
Кафедра ИУ5 «Информатика и вычислительная техника»

Курс «БКТИ»

**Отчет по лабораторной работе
«Написание Телеграмм Бота на языке Python»**

Выполнил:
студент группы РТ5-21

Васянин Н.Н.
Подпись и дата:

Проверил:
преподаватель каф.
ИУ5
Гапанюк Ю.Е.
Подпись и дата:

Москва, 2020 г.

1. Постановка задачи

Написать телеграмм бота, который представляет из себя викторину, на тему информатики. Общение с ботом, должно быть реализовано с помощью асинхронной библиотеки <aiogram>. Указание всех подзадач, после реализации, которых бот выполняет все команды(встроенные в викторину), отвечает на сообщения пользователя и имеет наиболее минимальное количество лазеек, с помощью которых можно было бы сломать алгоритмы бота:

- 1) создание базы данных с помощью списков, в которых содержались вопросы и ответы для викторины
- 2) создание счетчика для БД, чтобы вопросы и ответы отображались в верном порядке
- 3) использовать хендлеры для реагирования на те или иные действия пользователя
- 4) дополнить хендлеры асинхронными функциями, которые предоставят функционал бота пользователю
- 5) создать объекты клавиатуры, чтобы пользователь мог отвечать на вопросы из викторины
- 6) сделать корректное завершение и начало работы бота с помощью хендлеров

2. Текст программы

Файл <main.py>:

```
import logging
from aiogram.types import ReplyKeyboardMarkup, KeyboardButton
from aiogram import Bot, Dispatcher, executor, types
from config import BOT_TOKEN
from aiogram.types import Message, ReplyKeyboardRemove
from aiogram.dispatcher.filters import Command, Text
from bd_victor import qevictl
from value_bd import ValueBD

logging.basicConfig(level=logging.INFO)
bot = Bot(token=BOT_TOKEN)
dp = Dispatcher(bot)

async def ask(message: Message):
    menu = ReplyKeyboardMarkup(
        keyboard=[
            [
```

```

        KeyboardButton(text=qevict1[ValueBD.value1][1])
    ],
    [
        KeyboardButton(text=qevict1[ValueBD.value1][2]),
        KeyboardButton(text=qevict1[ValueBD.value1][3])
    ],
],
resize_keyboard=True
)
await message.answer(f"{qevict1[ValueBD.value1][0]}", reply_markup=menu)

async def askGame(message: Message):
    menu1 = ReplyKeyboardMarkup(
        keyboard=[
            [
                KeyboardButton(text="да"),
                KeyboardButton(text="нет")
            ],
        ],
        resize_keyboard=True
    )
    await message.answer("Хотите попробовать еще раз?", reply_markup=menu1)

@dp.message_handler(Command("start"))
async def start(message: Message):
    ValueBD.value1 = 0
    ValueBD.count = 0
    await ask(message)

@dp.message_handler(Text("4,4 Мегабайта"))
async def restart1(message: Message):
    await message.answer("Верный ответ")
    ValueBD.count = ValueBD.count + 1
    await message.answer(f"Количество правильных ответов: {ValueBD.count}")
    await message.answer("Игра завершена")
    ValueBD.value1 = 0
    ValueBD.count = 0
    await askGame(message)

@dp.message_handler(Text("4,4 Байта"))
async def restart2(message: Message):
    await message.answer(f"Количество правильных ответов: {ValueBD.count}")
    await message.answer("Игра завершена")
    ValueBD.value1 = 0
    ValueBD.count = 0
    await askGame(message)

@dp.message_handler(Text("4,4 Терабайта"))
async def restart3(message: Message):
    await message.answer(f"Количество правильных ответов: {ValueBD.count}")
    await message.answer("Игра завершена")
    ValueBD.value1 = 0
    ValueBD.count = 0
    await askGame(message)

@dp.message_handler(Text("да"))
async def restartAns1(message: Message):
    await message.answer("Нажмите на /start, чтобы начать игру заново",

```

```

reply_markup=ReplyKeyboardRemove())

@dp.message_handler(Text("нет"))
async def restartAns2(message: Message):
    await message.answer("Спасибо, что заглянули, но если все-таки появиться
желание сыграть, нажмите /start", reply_markup=ReplyKeyboardRemove())

@dp.message_handler()
async def verify(message: Message):
    if message.text == qevict1[ValueBD.value1][1] or message.text ==
qevict1[ValueBD.value1][2] or message.text == \
    qevict1[ValueBD.value1][3]:
        if ValueBD.value1 == 0 and message.text ==
qevict1[ValueBD.value2_1][ValueBD.value2_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 1 and message.text ==
qevict1[ValueBD.value3_1][ValueBD.value3_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 2 and message.text ==
qevict1[ValueBD.value4_1][ValueBD.value4_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 3 and message.text ==
qevict1[ValueBD.value5_1][ValueBD.value5_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 4 and message.text ==
qevict1[ValueBD.value6_1][ValueBD.value6_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 5 and message.text ==
qevict1[ValueBD.value7_1][ValueBD.value7_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 6 and message.text ==
qevict1[ValueBD.value8_1][ValueBD.value8_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 7 and message.text ==
qevict1[ValueBD.value9_1][ValueBD.value9_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 8 and message.text ==
qevict1[ValueBD.value10_1][ValueBD.value10_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 9 and message.text ==
qevict1[ValueBD.value11_1][ValueBD.value11_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 10 and message.text ==
qevict1[ValueBD.value12_1][ValueBD.value12_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        if ValueBD.value1 == 11 and message.text ==
qevict1[ValueBD.value13_1][ValueBD.value13_2]:
            ValueBD.count = ValueBD.count + 1
            await message.answer("Верный ответ")
        await message.answer(f"<----->",
reply_markup=ReplyKeyboardRemove())

```

```

        ValueBD.inc()
        await ask(message)

if __name__ == '__main__':
    executor.start_polling(dp, skip_updates=True)

```

Файл <bd_victor.py>:

```

qevict1 = [
    [
        "В результате компьютерного сбоя в первые часы 2000 года посетители
одной из Интернет-странички оказались в "
        "далеком будущем. Какой год они увидели на экране?",
        "19 100", "20 000", "20 100" # ответ 1
    ],
    [
        "В каком году выпущен первый персональный компьютер IBM PC?",
        "1991", "1981", "1971" # ответ 2
    ],
    [
        "Что означает слово "компьютер"?",
        "Мыслитель", "Вычислитель", "Умножитель" # ответ 2
    ],
    [
        "Какому устройству компьютера поставили памятник в Екатеринбурге на
набережной реки Исети?",
        "Системному блоку", "Монитору", "Клавиатуре" # ответ 3
    ],
    [
        "Недавно был представлен опытный образец нового офисного кресла,
оснащенного компьютерной системой, "
        "которое среди прочего может следить за осанкой сидящего. А что это
кресло может делать с вероятностью 96%??",
        "Определить, кто на нем сидит", "Определить вес сидящего", "Изменить
высоту кресла" # ответ 1
    ],
    [
        "В американском "Компьютерном словаре" написано: "Компьютер -
электронное устройство, которое не заменит "
        "человека до тех пор, пока не научится смеяться шуткам босса и..."
Продолжите фразу...",
        "Прятать свои ошибки", "Просить помощи у коллеги", "Сваливать вину на
другой компьютер" # ответ 3
    ],
    [
        "В музее компьютеров в США можно видеть блок памяти одного из первых
компьютеров, построенных в 50-х годах. "
        "Рядом с ним - современная микросхема с тем же объемом памяти.
Назовите третий предмет экспозиции.",
        "Микроскоп", "Телескоп", "Компакт-диск" # ответ 1
    ],
    [
        "Исследования, проведенные в лондонских офисах, показали, что в
среднем из находящегося внутри этих предметов "
        "мусора 15% приходится на чипсы, 15% - на сладости, 7% - на макароны;
по 1% - карандашная стружка, насекомые, "
        "ногти, волосы. Где же все это находится?",
        "В корзине для бумаг", "В клавиатуре", "В системном блоке" # ответ 2
    ],
    [
        "Устройство для компьютера, позволяющее ему связываться с другим
компьютером, через телефонную или кабельную "

```

```

        "сеть?",
        "Трекбол", "Телефон", "Модем" # ответ 3
    ],
    [
        "В фирме открылся музей истории компании "Microsoft". Среди прочих экспонатов на стене висит красный телефон. "
        "Сняв трубку, вы услышите голос самого важного человека в компании. Кто это?",
        "Билла Гейтса", "Директора компании", "Клиента" # ответ 3
    ],
    [
        "В начале XXI века эскимосы познакомились с изобретением, которое на их язык перевели как "ikiaqqivik" - "
        "путешествие сквозь слои". Что это за изобретение?",
        "Photo Shop", "Internet", "Compact Disk" # ответ 2
    ],
    [
        "Какое устройство памяти появилось раньше?",
        "Compact Disk", "USB Flash", "Floppy Disk" # ответ 3
    ],
    [
        "Какого объема был первый жесткий диск, который занимал место с холодильником?",
        "4,4 Мегабайта", "4,4 Байта", "4,4 Терабайта" # ответ 1
    ],
]

```

Файл <config.py>:

```

BOT_TOKEN = "1153959033:AAHUU0253Yiey1aU9vfXO2M7HPOx5k5wSBJk"

```

Файл <value_bd.py>:

```

class ValueBD:
    value1 = 0 # количество вопросов
    count = 0 # количество правильных ответов

    value2_1 = 0 # ответ 1
    value2_2 = 1

    value3_1 = 1 # ответ 2
    value3_2 = 2

    value4_1 = 2 # ответ 3
    value4_2 = 2

    value5_1 = 3 # ответ 4
    value5_2 = 3

    value6_1 = 4 # ответ 5
    value6_2 = 1

    value7_1 = 5 # ответ 6
    value7_2 = 3

    value8_1 = 6 # ответ 7
    value8_2 = 1

    value9_1 = 7 # ответ 8
    value9_2 = 2

```

```

value10_1 = 8 # ответ 9
value10_2 = 3

value11_1 = 9 # ответ 10
value11_2 = 3

value12_1 = 10 # ответ 11
value12_2 = 2

value13_1 = 11 # ответ 12
value13_2 = 3

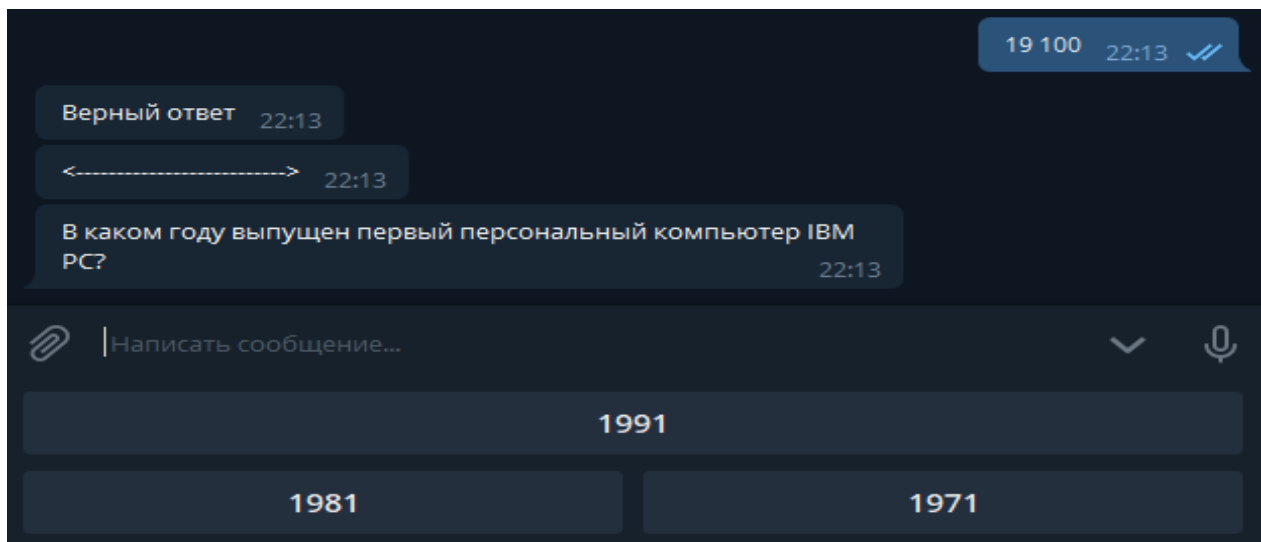
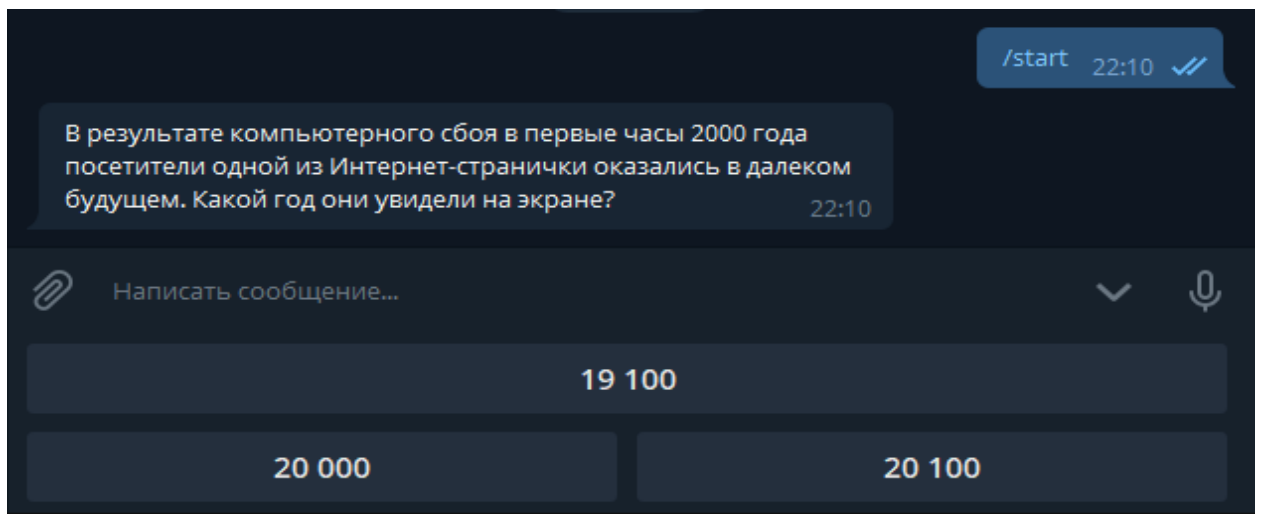
value14_1 = 12 # ответ 13
value14_2 = 1

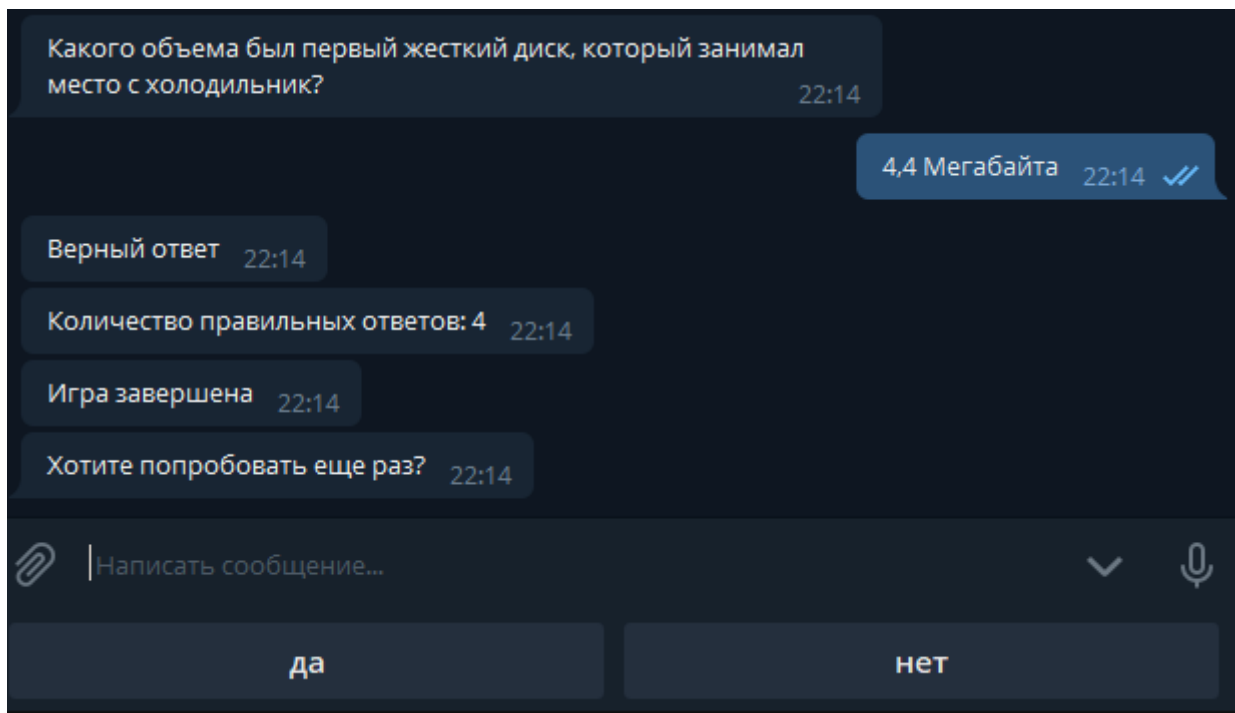
@staticmethod
def inc():
    ValueBD.value1 = ValueBD.value1 + 1
    if ValueBD.value1 < 13:
        return ValueBD.value1

```

3. Результаты работы телеграмм бота

На компьютере:





На телефоне:

