

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Радиотехнический»  
Кафедра «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования »

Домашнее задание

Выполнил:  
студент группы  
РТ5-31Б  
Данилова А.С.

Москва, 2024 г

## Постановка задачи:

Разработать телеграм-бот, в котором можно будет пройти тесты по математике, русскому языку и английскому языку. В конце бот будет выводить результат тестирования.

## Текст программы:

```
import telebot
from telebot import types

API_TOKEN = '7820479723:AAEuI6wgtNRx72h2-kfeIlnXJ21sRYYc5I4'
bot = telebot.TeleBot(API_TOKEN)

# Состояния
START, SUBJECT, MATH, RUSSIAN, ENGLISH, ENGLISH_TOPIC, QUIZ, RESULT = range(8)

# Вопросы
math_questions = [
    ("2 + 2 =", "4"),
    ("3 + 5 =", "8"),
    ("10 + 15 =", "25"),
    ("7 + 3 =", "10"),
    ("5 + 9 =", "14"),
    ("2 * 2 =", "4"),
    ("3 * 5 =", "15"),
    ("10 * 15 =", "150"),
    ("7 * 3 =", "21"),
    ("5 * 9 =", "45"),
]

russian_questions = [
    ("Укажите предложения, в которых нужно поставить ОДНУ запятую. Запишите номера этих предложений.\n1) Плоды этого растения полезные и вкусные и обладают прекрасным ароматом.\n2) Стало нестерпимо душно и пришлось открыть все окна.\n3) Из окна были видны стволы вишен да кусочек аллеи.\n4) Изучение роста необычных кристаллов имеет и теоретическое и практическое и общенаучное значение.\n5) Древние испанские мастера при строительстве замков применяли либо каменную либо кирпичную кладку.", "25"),
    ("Укажите предложения, в которых нужно поставить ОДНУ запятую. Запишите номера этих предложений.\n1) Для развития личности полезны и искусство и наука и жизненный опыт.\n2) Грубое или даже нелюбезное поведение людей может испортить настроение окружающим.\n3) Память накапливает добрый опыт и традиции и постоянно противостоит уничтожающей силе времени.\n4) Хорошие манеры и правильно выработанное поведение принесут человеку как хорошее настроение так и уважение окружающих.\n5) Белый теплоход уверенно рассекал невысокие волны и только по лёгкому дрожанию корпуса пассажиры догадывались о начале морской качки.", "45"),
    ("Укажите предложения, в которых нужно поставить ОДНУ запятую. Запишите номера этих предложений.\n1) Прогулка или разговор с другом были одинаково приятны для меня.\n2) Сумрак скрыл очертания лица и фигуры Ольги и набросил на неё как будто покрывало.\n3) На земле жилось нелегко и поэтому я очень полюбил бездонное небо.\n4) Ученье да труд всё перетрут.\n5) В колледже он с увлечением занимался как гуманитарными так и естественно-математическими дисциплинами.", "35"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-ые).\nИздали (1) он увидел дом (2) непохожий на другие (3) построенный (4) каким-то итальянским архитектором.", "23"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
```

```

ые).\nНад ещё не улётшимся (1) после недавней бури (2) бескрайним морем (3)
возвышалось небо (4) унизанное (5) ярко мерцавшими звёздами.", "4"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
ые).\nГород (1) вдали сверкающий на солнце (2) синие леса (3) окаймляющие
берега залива (4) казались мне особенно торжественными.", "1234"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
ые).\nБольшой пруд (1) густо заросший кувшинками (2) располагался (3) в
удалённой от дома (4) части старого парка.", "12"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
ые).\nВладимир (1) махавший косой не переставая (2) резал траву (3) не
выказывая (4) ни малейшего усилия.", "123"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
ые).\nТуча (1) нависшая (2) над высокими вершинами тополей (3) уже сыпала (4)
морозящим дождиком.", "13"),
    ("Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-
ые).\nПробравшись (1) через мокрый папоротник и какую-то (2) стелющуюся
растительность (3) выбираемся на едва приметную тропинку.", "3"),
]

```

```

english_questions = {
    "fruits": [
        ("Яблоко", ["Apple", "Banana", "Cherry", "Grape"]),
        ("Банан", ["Banana", "Orange", "Peach", "Mango"]),
        ("Виноград", ["Grape", "Apple", "Lemon", "Kiwi"]),
        ("Апельсин", ["Orange", "Peach", "Plum", "Berry"]),
        ("Персик", ["Peach", "Apple", "Banana", "Grape"]),
    ],
    "clothes": [
        ("Рубашка", ["Shirt", "Pants", "Hat", "Shoes"]),
        ("Штаны", ["Pants", "Shirt", "Belt", "Socks"]),
        ("Шляпа", ["Hat", "Scarf", "Gloves", "Coat"]),
        ("Обувь", ["Shoes", "Boots", "Sandals", "Slippers"]),
        ("Платье", ["Dress", "Skirt", "Blouse", "Jacket"]),
    ],
    "animals": [
        ("Собака", ["Dog", "Cat", "Mouse", "Bird"]),
        ("Кошка", ["Cat", "Dog", "Fish", "Rabbit"]),
        ("Птица", ["Bird", "Insect", "Mammal", "Reptile"]),
        ("Лошадь", ["Horse", "Donkey", "Mule", "Zebra"]),
        ("Корова", ["Cow", "Goat", "Sheep", "Pig"]),
    ],
}

```

```

user_data = {}

```

```

@bot.message_handler(commands=['start'])

```

```

def start_quiz(message):
    user_data[message.chat.id] = {'state': START, 'score': 0,
'question_index': 0}
    bot.send_message(message.chat.id, "Добро пожаловать! Выберите предмет:
/math, /russian, /english")

```

```

@bot.message_handler(commands=['math'])

```

```

def math_quiz(message):
    user_data[message.chat.id]['state'] = MATH
    user_data[message.chat.id]['score'] = 0
    user_data[message.chat.id]['question_index'] = 0
    ask_math_question(message)

```

```

@bot.message_handler(commands=['russian'])

```

```

def russian_quiz(message):
    user_data[message.chat.id]['state'] = RUSSIAN
    user_data[message.chat.id]['score'] = 0
    user_data[message.chat.id]['question_index'] = 0

```

```

ask_russian_question(message)

@bot.message_handler(commands=['english'])
def english_quiz(message):
    user_data[message.chat.id]['state'] = ENGLISH
    bot.send_message(message.chat.id, "Выберите тему: /fruits, /clothes, /animals")

@bot.message_handler(commands=['fruits', 'clothes', 'animals'])
def english_topic(message):
    topic = message.text[1:] # Убираем '/'
    user_data[message.chat.id]['state'] = ENGLISH_TOPIC
    user_data[message.chat.id]['score'] = 0
    user_data[message.chat.id]['question_index'] = 0
    user_data[message.chat.id]['topic'] = topic
    ask_english_question(message)

def ask_math_question(message):
    user_id = message.chat.id
    index = user_data[user_id]['question_index']
    if index < len(math_questions):
        question, correct_answer = math_questions[index]
        bot.send_message(user_id, question)
        user_data[user_id]['correct_answer'] = correct_answer
    else:
        show_result(message)

def ask_russian_question(message):
    user_id = message.chat.id
    index = user_data[user_id]['question_index']
    if index < len(russian_questions):
        question, correct_answer = russian_questions[index]
        bot.send_message(user_id, question)
        user_data[user_id]['correct_answer'] = correct_answer
    else:
        show_result(message)

def ask_english_question(message):
    user_id = message.chat.id
    index = user_data[user_id]['question_index']
    topic = user_data[user_id]['topic']
    if index < len(english_questions[topic]):
        question, options = english_questions[topic][index]
        correct_answer = options[0] # Первый вариант - правильный
        markup = types.ReplyKeyboardMarkup(one_time_keyboard=True)
        for option in options:
            markup.add(option)
        bot.send_message(user_id, question, reply_markup=markup)
        user_data[user_id]['correct_answer'] = correct_answer
    else:
        show_result(message)

@bot.message_handler(func=lambda message: user_data.get(message.chat.id, {}).get('state') in [MATH, RUSSIAN, ENGLISH, ENGLISH_TOPIC])
def handle_answer(message):
    user_id = message.chat.id
    correct_answer = user_data[user_id]['correct_answer']
    if message.text.lower() == correct_answer.lower():
        user_data[user_id]['score'] += 1
        user_data[user_id]['question_index'] += 1

    if user_data[user_id]['state'] == MATH:
        ask_math_question(message)
    elif user_data[user_id]['state'] == RUSSIAN:

```

```

        ask_russian_question(message)
    elif user_data[user_id]['state'] in [ENGLISH, ENGLISH_TOPIC]:
        ask_english_question(message)

def show_result(message):
    user_id = message.chat.id
    score = user_data[user_id]['score']
    total_questions = len(math_questions) if user_data[user_id]['state'] ==
MATH else len(russian_questions) if user_data[user_id]['state'] == RUSSIAN
else len(english_questions[user_data[user_id]['topic']])
    bot.send_message(user_id, f"Твой результат: {score} из
{total_questions}!")
    user_data[user_id]['state'] = START
    user_data[user_id]['state'] = RESULT
    bot.send_message(user_id, "Выберите предмет: /math, /russian, /english
или /stop для завершения.")

@bot.message_handler(commands=['stop'])
def stop_quiz(message):
    bot.send_message(message.chat.id, "Спасибо за участие! До свидания!")

# Добавляем возможность начать викторину заново
    bot.send_message(message.chat.id, "Чтобы начать заново, введите /start.")

bot.polling()

```

## Экранные формы с примерами выполнения программы

23 декабря



/start 13:31 ✓✓



Добро пожаловать! Выберите предмет: /math, /russian, /english 13:31



/english 13:31 ✓✓



Выберите тему: /fruits, /clothes, /animals 13:31



/clothes 13:31 ✓✓



Рубашка 13:31



Shirt 13:31 ✓✓



Штаны 13:31



Pants 13:31 ✓✓



Шляпа 13:31



Hat 13:31 ✓✓



Обувь 13:31



Shoes 13:31 ✓✓



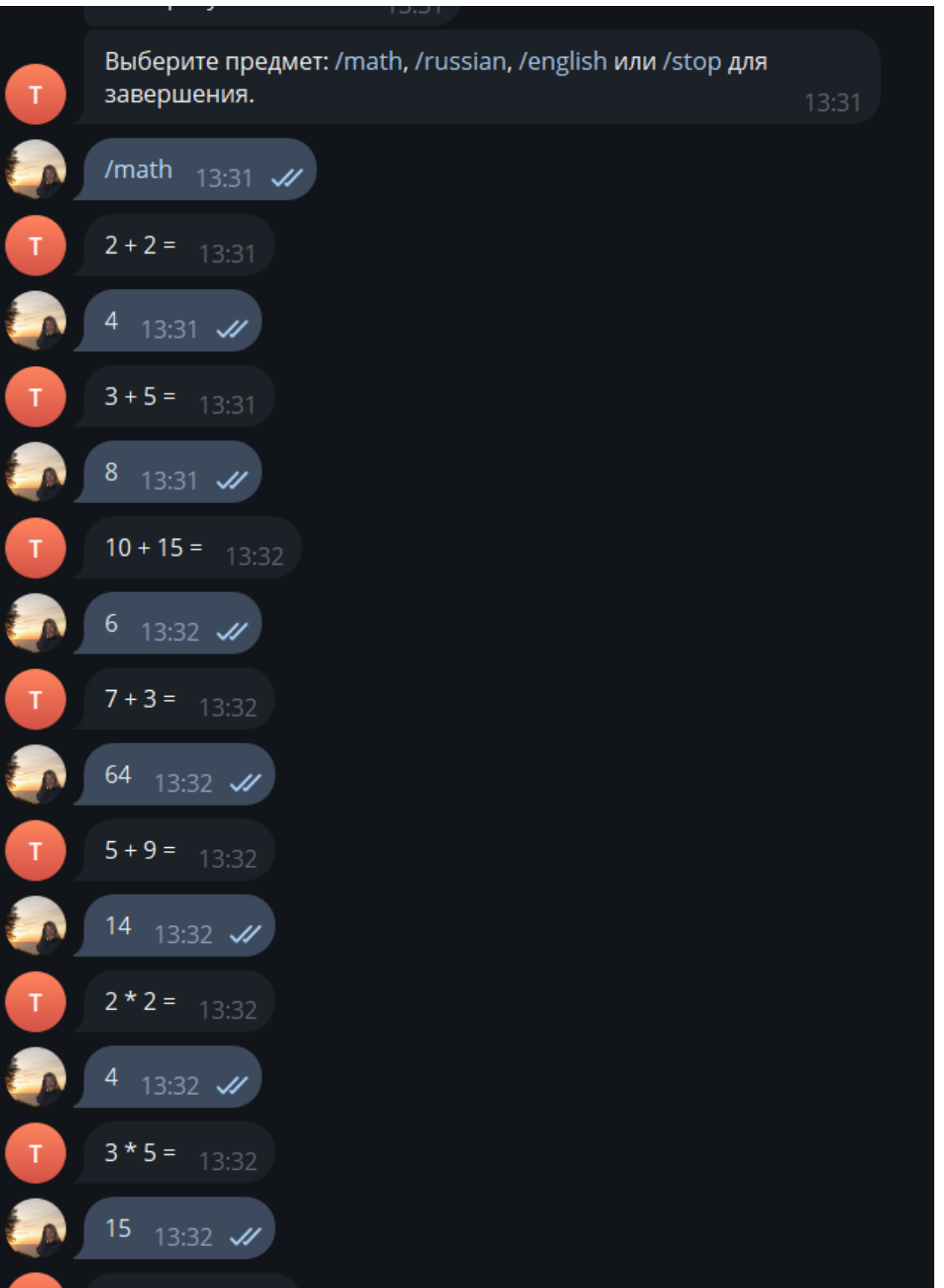
Платье 13:31



Dress 13:31 ✓✓



Твой результат: 5 из 5! 13:31



Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-ые).

Туча (1) нависшая (2) над высокими вершинами тополей (3) уже сыпала (4) морозящим дождем.

13:32

5 13:32 ✓✓

Укажите цифру(-ы), на месте которой(-ых) должна(-ы) стоять запятая(-ые).

Пробравшись (1) через мокрый папоротник и какую-то (2) стелющуюся растительность (3) выбираемся на едва приметную тропинку.

13:32

5 13:32 ✓✓

Твой результат: 3 из 10! 13:32

Выберите предмет: /math, /russian, /english или /stop для завершения.

13:32

/stop 13:33 ✓✓

Спасибо за участие! До свидания! 13:33