МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
им. Н.Э. Баумана

Факультет “Информатика и системы управления”  
Кафедра “Системы обработки информации и управления”

A logo with a gold star and blue ribbon

Description automatically generated

Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по лабораторной работе №5  
“Разработка простого бота для Telegram с использованием языка Python.”

**Выполнил:**  
Студент группы ИУ5-34Б  
Савинский А.Ю.  
**Преподаватель:**  
Нардид А.Н.

Москва 2025

Задание лабораторной работы[​](https://iu5edu.ru/wiki/cpp2/docs/labs/lab6/#%D0%B7%D0%B0%D0%B4%D0%B0%D0%BD%D0%B8%D0%B5)

1. Разработайте бота для Telegram. Бот должен реализовывать конечный автомат из трех состояний.

Текст программы

bot.py

import telebot

from telebot import types

import random

import os

from dotenv import load\_dotenv

load\_dotenv()

TOKEN = os.getenv("BOT\_TOKEN")

bot = telebot.TeleBot(TOKEN)

duels = {}

stats = {}

rpg\_stats = {}

rpg\_battles = {}

@bot.message\_handler(commands=['start'])

def start(message):

markup = types.ReplyKeyboardMarkup(resize\_keyboard=True)

btn1 = types.KeyboardButton("🎲 Бросить кубик")

btn2 = types.KeyboardButton("🔢 Случайное число")

btn3 = types.KeyboardButton("🪙 Монета")

btn4 = types.KeyboardButton("❓ Помощь")

btn5 = types.KeyboardButton("📊 Статистика")

btn6 = types.KeyboardButton("🗡 RPG-атака")

markup.add(btn1, btn2, btn3, btn4, btn5, btn6)

bot.send\_message(

message.chat.id,

"Привет! Я RandomGameBot. Выбери действие 👇",

reply\_markup=markup

)

@bot.message\_handler(commands=['duel'])

def duel\_command(message):

chat\_id = message.chat.id

chat\_type = message.chat.type

if chat\_type == "private":

bot.send\_message(chat\_id, "Дуэли доступны только в групповых чатах 🙂")

return

challenger = message.from\_user

opponent = None

if message.reply\_to\_message:

opponent = message.reply\_to\_message.from\_user

if not opponent and message.entities:

for ent in message.entities:

if ent.type == "text\_mention" and ent.user:

opponent = ent.user

break

if ent.type == "mention":

username = message.text[ent.offset: ent.offset + ent.length]

try:

chat = bot.get\_chat(username)

opponent = chat

break

except Exception:

pass

if not opponent:

bot.send\_message(

chat\_id,

"Чтобы начать дуэль, ответь на сообщение соперника командой /duel\n"

"или напиши /duel и выбери его через упоминание."

)

return

if challenger.id == opponent.id:

bot.send\_message(chat\_id, "Нельзя вызвать на дуэль самого себя)")

return

if opponent.is\_bot:

bot.send\_message(chat\_id, "Нельзя вызвать меня на дуэль!")

return

duels[chat\_id] = {

"challenger\_id": challenger.id,

"opponent\_id": opponent.id,

"challenger\_name": challenger.first\_name or "Игрок 1",

"opponent\_name": opponent.first\_name or "Игрок 2",

"rolls": {}

}

bot.send\_message(

chat\_id,

f"⚔️ Дуэль!\n"

f"{duels[chat\_id]['challenger\_name']} вызывает {duels[chat\_id]['opponent\_name']}.\n\n"

f"Нажмите на кнопку «🎲 Бросить кубик»."

)

@bot.message\_handler(commands=['rpg'])

def rpg\_command(message):

chat\_id = message.chat.id

chat\_type = message.chat.type

if chat\_type == "private":

bot.send\_message(chat\_id, "RPG-сражения доступны только в групповых чатах 🙂")

return

challenger = message.from\_user

opponent = None

if message.reply\_to\_message:

opponent = message.reply\_to\_message.from\_user

if not opponent and message.entities:

for ent in message.entities:

if ent.type == "text\_mention" and ent.user:

opponent = ent.user

break

if ent.type == "mention":

username = message.text[ent.offset: ent.offset + ent.length]

try:

chat = bot.get\_chat(username)

opponent = chat

break

except Exception:

pass

if not opponent:

bot.send\_message(

chat\_id,

"Чтобы начать RPG-бой, ответь на сообщение соперника командой /rpg\n"

"или напиши /rpg и выбери его через упоминание."

)

return

if challenger.id == opponent.id:

bot.send\_message(chat\_id, "Нельзя сражаться с самим собой")

return

if opponent.is\_bot:

bot.send\_message(chat\_id, "Нельзя вызвать меня на RPG-бой!")

return

if chat\_id in rpg\_battles:

bot.send\_message(chat\_id, "В этом чате уже идёт RPG-бой! Сначала завершите его.")

return

hp\_start = 30

rpg\_battles[chat\_id] = {

"p1\_id": challenger.id,

"p2\_id": opponent.id,

"p1\_name": challenger.first\_name or "Игрок 1",

"p2\_name": opponent.first\_name or "Игрок 2",

"hp": {

challenger.id: hp\_start,

opponent.id: hp\_start,

},

"turn": challenger.id,

}

bot.send\_message(

chat\_id,

f"🗡 Начинается RPG-бой!\n"

f"{rpg\_battles[chat\_id]['p1\_name']} VS {rpg\_battles[chat\_id]['p2\_name']}\n"

f"У каждого по {hp\_start} HP.\n\n"

f"Ход {rpg\_battles[chat\_id]['p1\_name']}. Нажмите кнопку «🗡 Атака»."

)

@bot.message\_handler(func=lambda m: True)

def handle\_message(message):

text = message.text or ""

chat\_type = message.chat.type

if text.startswith("/"):

return

if chat\_type == "private":

if text == "🎲 Бросить кубик":

roll = random.randint(1, 6)

bot.send\_message(message.chat.id, f"Выпало: {random.randint(1, 6)} 🎲")

elif text == "🔢 Случайное число":

bot.send\_message(message.chat.id, f"Случайное число: {random.randint(1, 100)} 🔢")

elif text == "🪙 Монета":

bot.send\_message(message.chat.id, f"Монета: {random.choice(['орёл', 'решка'])}")

elif text == "❓ Помощь":

bot.send\_message(

message.chat.id,

"Что я умею:\n"

"🎲 Бросить кубик\n"

"🔢 Выдать случайное число\n"

"🪙 Подбросить монету\n"

)

else:

bot.send\_message(message.chat.id, "Такой команды не найдено. Выберите действие по кнопке ниже")

elif chat\_type in ("group", "supergroup"):

chat\_id = message.chat.id

user\_id = message.from\_user.id

name = message.from\_user.first\_name or "Игрок"

if text == "🎲 Бросить кубик":

roll = random.randint(1, 6)

bot.send\_message(chat\_id, f"{name} бросил(а) кубик: {roll} 🎲")

state = duels.get(chat\_id)

if state and user\_id in (state["challenger\_id"], state["opponent\_id"]):

if user\_id in state["rolls"]:

return

state["rolls"][user\_id] = roll

if len(state["rolls"]) == 2:

c\_id = state["challenger\_id"]

o\_id = state["opponent\_id"]

c\_name = state["challenger\_name"]

o\_name = state["opponent\_name"]

c\_roll = state["rolls"][c\_id]

o\_roll = state["rolls"][o\_id]

result\_text = (

"⚔️ \*Итоги дуэли:\*\n\n"

f"{c\_name}: {c\_roll}\n"

f"{o\_name}: {o\_roll}\n\n"

)

if c\_roll > o\_roll:

winner\_id = c\_id

winner\_name = c\_name

elif o\_roll > c\_roll:

winner\_id = o\_id

winner\_name = o\_name

else:

winner\_id = None

if winner\_id is None:

result\_text += "🤝 Ничья!"

else:

result\_text += f"🏆 Победитель: \*{winner\_name}\*!"

if winner\_id not in stats:

stats[winner\_id] = 0

stats[winner\_id] += 1

bot.send\_message(chat\_id, result\_text, parse\_mode="Markdown")

duels.pop(chat\_id, None)

elif text == "🗡 RPG-атака":

state = rpg\_battles.get(chat\_id)

if not state:

bot.send\_message(chat\_id, "Сначала начните RPG-бой командой /rpg (по реплаю или с упоминанием).")

return

if user\_id not in (state["p1\_id"], state["p2\_id"]):

bot.send\_message(chat\_id, "Ты не участвуешь в этом сражении.")

return

if state["turn"] != user\_id:

current\_name = state["p1\_name"] if state["turn"] == state["p1\_id"] else state["p2\_name"]

bot.send\_message(chat\_id, f"Сейчас ход {current\_name}!")

return

attacker\_id = user\_id

defender\_id = state["p2\_id"] if attacker\_id == state["p1\_id"] else state["p1\_id"]

attacker\_name = state["p1\_name"] if attacker\_id == state["p1\_id"] else state["p2\_name"]

defender\_name = state["p2\_name"] if defender\_id == state["p2\_id"] else state["p1\_name"]

dmg = random.randint(5, 15)

state["hp"][defender\_id] -= dmg

if state["hp"][defender\_id] < 0:

state["hp"][defender\_id] = 0

attacker\_hp = state["hp"][attacker\_id]

defender\_hp = state["hp"][defender\_id]

text\_out = (

f"🗡 {attacker\_name} атакует {defender\_name} и наносит {dmg} урона!\n"

f"HP {attacker\_name}: {attacker\_hp}\n"

f"HP {defender\_name}: {defender\_hp}\n"

)

if defender\_hp <= 0:

text\_out += f"\n💀 {defender\_name} пал в бою.\n🏆 Победитель: {attacker\_name}!"

winner\_id = attacker\_id

if winner\_id not in rpg\_stats:

rpg\_stats[winner\_id] = 0

rpg\_stats[winner\_id] += 1

bot.send\_message(chat\_id, text\_out)

rpg\_battles.pop(chat\_id, None)

else:

state["turn"] = defender\_id

text\_out += f"\nТеперь ход {defender\_name} (нажми «🗡 RPG-атака»)."

bot.send\_message(chat\_id, text\_out)

elif text == "❓ Помощь":

bot.send\_message(

chat\_id,

"Это групповой режим.\n"

"⚔️ Дуэль 1х1:\n"

" • Ответь на сообщение соперника командой /duel\n"

" • Оба жмите «🎲 Бросить кубик».\n\n"

"🗡 RPG-бой:\n"

" • Ответь на сообщение соперника командой /rpg\n"

" • Или используй /rpg с упоминанием.\n"

" • Затем по очереди жмите «🗡 RPG-атака», пока у кого-то не кончится HP."

)

elif text == "🔢 Случайное число":

bot.send\_message(message.chat.id, f"Случайное число: {random.randint(1, 100)} 🔢")

elif text == "🪙 Монета":

bot.send\_message(message.chat.id, f"Монета: {random.choice(['орёл', 'решка'])}")

elif text == "📊 Статистика":

user\_id = message.from\_user.id

duel\_wins = stats.get(user\_id, 0)

rpg\_wins = rpg\_stats.get(user\_id, 0)

bot.send\_message(

chat\_id,

f"📊 Ваша статистика:\n"

f"• Побед в дуэлях: {duel\_wins}\n"

f"• Побед в RPG-боях: {rpg\_wins}"

)

else:

return

bot.infinity\_polling()

Экранные формы с примерами выполнения программы













