

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное учреждение высшего образования  
«Чувашский государственный университет И.Н. Ульянова»  
Факультет информатики и вычислительной техники  
Кафедра вычислительной техники

Кросс-платформенные средства разработки программного обеспечения  
«Nucleus(Чат-бот telegram)»

**Выполнил:**

Студент группы ИВТ-41-20  
Галкин Д.С.

**Проверил:**

Ковалев С.В.

Чебоксары, 2023

## Цель работы:

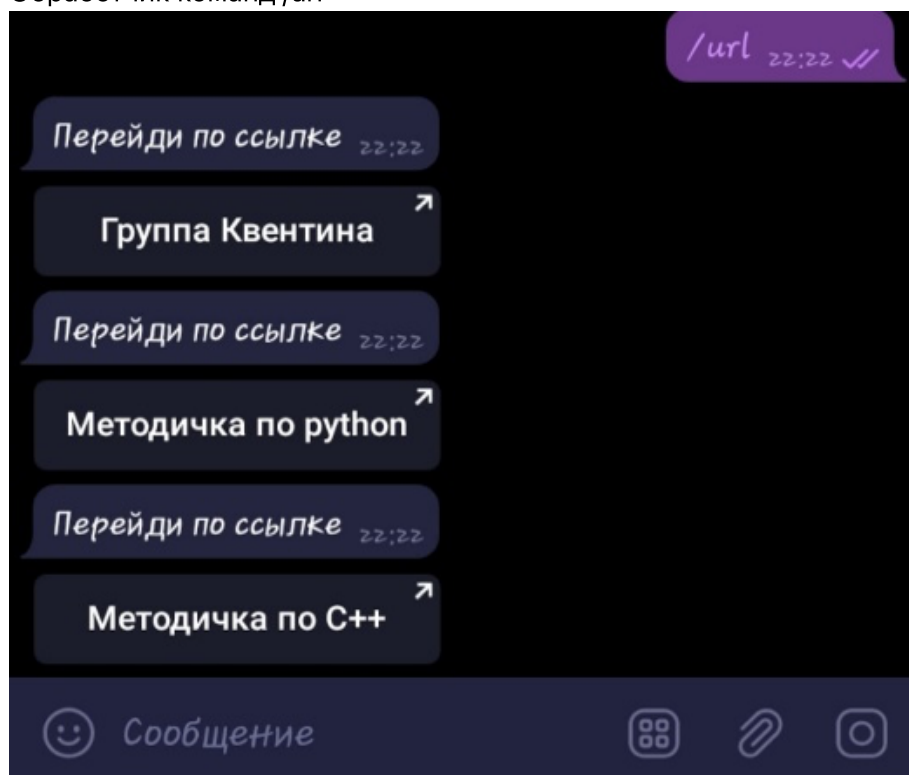
1. Выбрать среду разработки и продумать интерфейс использования
2. Подобрать подходящие библиотеки под разные задачи
3. Выбрать одну из главных модулей поддерживающие структуру для написания Telegram-бота
4. Разместить на хостинге для бесперебойной работы бота
5. Проверить правильность работы бота

## Разработка приложения:

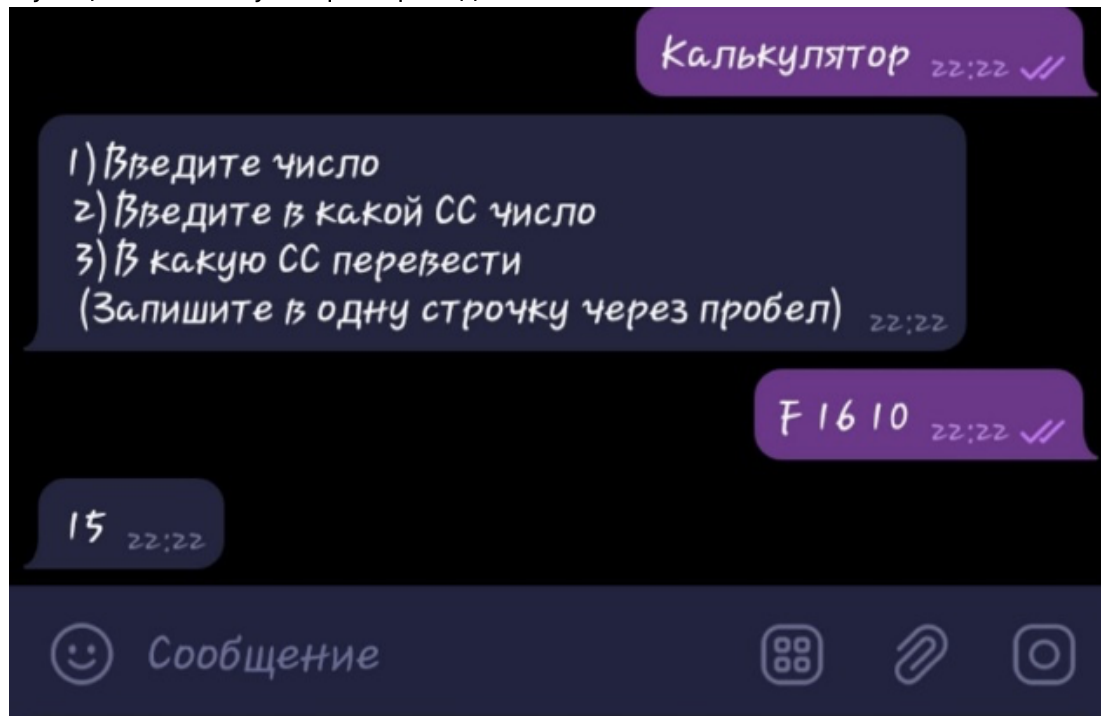
- Создание бота - перед написанием функционала бота, его необходимо создать при помощи главного официального бота Telegram BotFather
- Проектирование интерфейса - главный функционал, построен с помощью ReplyKeyboardMarkup, который позволяет создавать пользовательские клавиатуры для комфортного пользования
- Разработка функционала - обработка различных команд, поступающих боту, происходит с помощью специальных "Хендлеров" - конструкции, которые принимают команду разных типах, обрабатывают и производят действие при помощи функций.

## Интерфейс:

- Обработчик команд /url



- Функционал калькулятора перевода СС



#### ⚠ Внимание!

Построения функционала калькулятора перевода систем счисления

Такая функция очень востребована в некоторых заданиях ЕГЭ по информатике. Был специально написан алгоритм для вычисления подобных арифметических действий

## Деплой на Heruko:

Установить расширение при помощи командной строки под названием Heroku CLI, для удобной работы с платформой. Также создать 2 дополнительных файла:

1. Profile - файл в котором находится команда для запуска бота
2. Requirements - содержит описание всех модулей

## Полный текст программы:

Класс BotHandler

```
class BotHandler:

    def __init__(self, token):
        self.token = token
        self.api_url = "https://api.telegram.org/bot{}/".format(token)

    def get_updates(self, offset=None, timeout=30):
        method = 'getUpdates'
        params = {'timeout': timeout, 'offset': offset}
        resp = requests.get(self.api_url + method, params)
        result_json = resp.json()['result']
        return result_json

    def send_message(self, chat_id, text):
        params = {'chat_id': chat_id, 'text': text}
        method = 'sendMessage'
        resp = requests.post(self.api_url + method, params)
        return resp

    def get_last_update(self):
        get_result = self.get_updates()

        if len(get_result) > 0:
            last_update = get_result[-1]
        else:
            last_update = get_result[len(get_result)]

        return last_update
```

## Функция арифметических операций СС

```
# Функции
m16 = [10, 11, 12, 13, 14, 15]
c16 = ['A', 'B', 'C', 'D', 'E', 'F']
k = []
def TenSS(x1,x2):
    # Начальная система счисления 10, преревод с любыми СС(Кроме 16)
    k = []
    b = x1
    c = x2
    while b != 0:
        if b % c in m16:
            for i in range(len(m16)):
                if m16[i] == b % c:
                    k.append(c16[i])
                    break
            else:
                k.append(str(b % c))
        b //= c
    k.reverse()
    s = ''.join(k)
    return s

def SSTen(x1, x2):
    # Начальная система счисления x, преревод в 10-ую СС
    b = x1
    c = x2
    s = 0
    st = 1
    for i in range(1, len(b) + 1):
        if b[-i] in c16:
            for j in range(len(c16)):
                if c16[j] == b[-i]:
                    s += int(m16[j]) * st
                    break
            else:
                s += int(b[-i]) * st
        st *= c
    return s

@bot.message_handler(commands=['help'])
def help(message):
    bot.send_message(message.chat.id, "/start – Отправка заданий/n /help – Помощь/n /url – Ссылки на важную информацию")
def mess(b,d,c):
    flag = True
    while flag:
        kol = 0
        for i in b:
            if i in c16:
                for j in range(len(c16)):
                    if i == c16[j]:
                        i = m16[j]
                        break
```

```
        if int(i) >= d:
            return ('Число некорректное, введите число еще раз')
            break
        kol += 1
    if kol == len(b):
        flag = False
if d == 10:
    b = int(b)
    return TenSS(b, c)
elif c == 10:
    return SSTen(b, d)
else:
    return TenSS(SSTen(b, d), c)
```

## Ссылки на сайты

```
# Ссылки на сайты
@bot.message_handler(commands=['url'])
def url(message):
    markup001 = types.InlineKeyboardMarkup()
    btn_may_group = types.InlineKeyboardButton(text="Группа Квентина",
url="https://vk.com/chbquentin")
    markup001.add(btn_may_group)
    bot.send_message(message.chat.id, "Перейди по ссылке", reply_markup=markup001)
    markup002 = types.InlineKeyboardMarkup()
    site_python = types.InlineKeyboardButton(text="Методичка по python",

url="https://docs.python.org/3/tutorial/index.html")
    markup002.add(site_python)
    bot.send_message(message.chat.id, "Перейди по ссылке", reply_markup=markup002)
    markup003 = types.InlineKeyboardMarkup()
    site_C = types.InlineKeyboardButton(text="Методичка по C++",
url="https://www.rulit.me/books/c-bazovyj-
kurs-read-271738-1.html")
    markup003.add(site_C)
    bot.send_message(message.chat.id, "Перейди по ссылке", reply_markup=markup003)
```

## Действующие кнопки

```
# Действующие кнопки
@bot.message_handler(commands=['start'])
def send_welcome(message: Message):
    bot.send_message(message.chat.id, "Добро пожаловать", reply_markup=Keyboard())
# отправка методички
@bot.message_handler(content_types=['text'])
def text(message):
    if message.text == "Калькулятор":
        bot.send_message(message.chat.id, "1)Введите число\n 2)Введите в какой СС  
число\n 3)В какую СС перевести\n (Запишите в одну строчку через пробел)")
    elif len(message.text.split(" ")) == 3:
        line = message.text
        b = line.split(" ")[0]
        d = int(line.split(" ")[1])
        c = int(line.split(" ")[2])
        bot.send_message(message.chat.id, mess(b, d, c))
    elif message.text == "Задания":
        bot.send_message(message.chat.id, "Выберите часть: ",
reply_markup=Keyboard1())
    elif message.text == "Часть 1":
        bot.send_message(message.chat.id, "Выберите задания: ",
reply_markup=Keyboard2())
    elif message.text == "Задания 1-11":
        bot.send_message(message.chat.id, " Выберите задание: ",
reply_markup=Keyboard3())
    elif message.text == "Задание 10":
        img1 = ('https://i.ibb.co/XLKhxzd/10-a.jpg')
        img2 = ('https://i.ibb.co/zmqvCr6/10.png')
        media = [types.InputMediaPhoto(img1, "1"), types.InputMediaPhoto(img2, "2")]
        bot.send_media_group(message.chat.id, media)
    elif message.text == "Задание 9":
        img3 = ("https://i.ibb.co/xgFbTYy/9-a.jpg")
        img4 = ('https://i.ibb.co/QDj3RPB/9.jpg')
        img5 = ('https://i.ibb.co/3WFvbWH/9.jpg')
        media = [types.InputMediaPhoto(img3, "1"), types.InputMediaPhoto(img4, "2"),
types.InputMediaPhoto(img5, "3")]
        bot.send_media_group(message.chat.id, media)
    elif message.text == "Задания 12-22":
        bot.send_message(message.chat.id, " Выберите задание: ",
reply_markup=Keyboard4())
    elif message.text == "Часть 2":
        bot.send_message(message.chat.id, "Выберите задания: ",
reply_markup=Keyboard5())
    elif message.text == "Задание 16":
        img6 = ("https://i.ibb.co/FJvLNxr/zadanie-16-1.jpg")
        img7 = ("https://i.ibb.co/VTN1kVz/zadanie-16-2.jpg")
        img8 = ("https://i.ibb.co/W20GjLD/zadanie-16-3.jpg")
        img9 = ("https://i.ibb.co/V9R4XtG/zadanie-16-4.jpg")
        media = [types.InputMediaPhoto(img6, '1'), types.InputMediaPhoto(img7, '2'),
types.InputMediaPhoto(img8, '3'),
types.InputMediaPhoto(img9, '4')]
        bot.send_media_group(message.chat.id, media)
    elif message.text == "Задание 26":
        img10 = ('https://i.ibb.co/RCG89G0/26-a.jpg')
```



```
img11 = ('https://i.ibb.co/kyfyVpL/26.jpg')
img12 = ('https://i.ibb.co/PzhRCHX/26.jpg')
img13 = ('https://i.ibb.co/bsPHLYQ/26.jpg')
img14 = ('https://i.ibb.co/jhf6b3d/26.jpg')
img15 = ('https://i.ibb.co/fq30dn4/26.jpg')
img16 = ('https://i.ibb.co/FX7qszy/26.jpg')
img17 = ('https://i.ibb.co/M2RVH1y/26.jpg')
img18 = ('https://i.ibb.co/3BvKsKV/26.jpg')
media = [types.InputMediaPhoto(img10, "1"), types.InputMediaPhoto(img11, "2"),
          types.InputMediaPhoto(img12, "3"),
          types.InputMediaPhoto(img13, "4"), types.InputMediaPhoto(img14, "5"),
          types.InputMediaPhoto(img15, "6"),
          types.InputMediaPhoto(img16, "7"), types.InputMediaPhoto(img17, "8"),
          types.InputMediaPhoto(img18, "9")]
bot.send_media_group(message.chat.id, media)
elif message.text == "Задание 18":
    img19 = ('https://i.ibb.co/dJFsJmQ/ddfa3fdfe08fc21b94edd988f806b714-0.jpg')
    img20 = ('https://i.ibb.co/K6rxKGh/ddfa3fdfe08fc21b94edd988f806b714-1.jpg')
    img21 = ('https://i.ibb.co/7RzP94s/ddfa3fdfe08fc21b94edd988f806b714-2.jpg')
    img22 = ('https://i.ibb.co/XCz876b/ddfa3fdfe08fc21b94edd988f806b714-3.jpg')
    img23 = ('https://i.ibb.co/x1CgNJV/ddfa3fdfe08fc21b94edd988f806b714-4.jpg')
    media = [types.InputMediaPhoto(img19, "1"), types.InputMediaPhoto(img20, "2"),
              types.InputMediaPhoto(img21, "3"),
              types.InputMediaPhoto(img22, "4"), types.InputMediaPhoto(img23, "5")]
    bot.send_media_group(message.chat.id, media)
elif message.text == "Стартовое меню":
    bot.send_message(message.chat.id, "Добро пожаловать", reply_markup=Keyboard())
```