

Плюсы:

1. Использование классов
2. Применение стандартных структур данных
3. Классы не сильно связаны
4. Отсутствие глобальных переменных

Минусы:

1. Присутствуют закомментированные участки кода

```
private static void Tr_MessagePhoto(object sendr, MessagePhoto<User, Chat> e)
{
    //Console.WriteLine("ID сообщения:{0}\nID отправителя:{1}\nИмя отправителя:{2}\nИмя:{3} Фамилия:{4}\nДата:{5}\nКомментарий к фото: {6}",
    //    e.message_id, e.from.id, e.from.username, e.from.first_name, e.from.last_name, e.date, e.caption);
    //for (int i = 0; i < e.photo.Count; i++)
    //{
    //    Console.WriteLine("Фото №{0}", i + 1);
    //    Console.WriteLine("ID файла:{0}\nРазмер файла:{1}байт\nШирина:{2} Высота:{3}\n",
    //        e.photo[i].file_id, e.photo[i].file_size, e.photo[i].width, e.photo[i].height);
    //}
    //MethodBot m = new MethodBot(Settings.Default.Token);
    //m.SendMessage("Вы нам прислали фотографию", 243746390);
    //m.SendPhotoLink(243746390, e.photo[e.photo.Count - 1].file_id, e.caption);
}
```

2. Отсутствуют комментарии в теле методов

```
public void SendMessage(string message, int ChatID)
{
    using (WebClient webClient = new WebClient())
    {
        NameValueCollection pars = new NameValueCollection();
        pars.Add("chat_id", ChatID.ToString());
        pars.Add("text", message);
        webClient.UploadValues(LINK + _token + "/sendMessage", pars);
    }
}
```

### 3. Методы большого размера с повторяющимися кусками кода

```
public void GetUpdates()
{
    while (true)
    {
        using (WebClient webClient = new WebClient())
        {
            string response = webClient.DownloadString("https://api.telegram.org/bot" + _token + "/getupdates?offset=" + (LastUpdateID + 1));
            if (response.Length <= 23)
                continue;
            var N = JSON.Parse(response);
            foreach (JSONNode r in N["result"].AsArray)
            {
                string _type = r["message"].ToString();
                _type = WhatsType(_type).Replace("\\", "");
                LastUpdateID = r["update_id"].AsInt;
                #region SWITCH
                switch (_type)
                {
                    case "text":
                    {
                        try
                        {
                            MessageText.Method.Name.ToString();
                            Notification();
                            Console.WriteLine("Мы получили текстовое сообщение");
                            Notification(false);
                        }
                        catch (Exception)
                        { break; }
                        GetMessageText(r);
                        break;
                    }
                    case "sticker":
                    {
                        try
                        {
                            MessageText.Method.Name.ToString();
                            Notification();
                            Console.WriteLine("Мы получили стикер");
                            Notification(false);
                        }
                        catch (Exception)
                        { break; }
                        GetMessageSticker(r);
                        break;
                    }
                    case "photo":
                    {
                        try
                        {

```

### 4. Используются некорректные названия переменных (названия переменных начинаются то с большого, то с маленького регистра)

```
Stack<JSONNode> stack = new Stack<JSONNode>();
JSONNode ctx = null;
int i = 0;
StringBuilder Token = new StringBuilder();
StringBuilder TokenName = new StringBuilder();
bool QuoteMode = false;
bool TokenIsString = false;
```

## 5. Нет переносов строк, приходится использовать прокрутку

```
public void SaveToCompressedStream(System.IO.Stream aData)
{
    throw new Exception("Can't use compressed functions. You need include the SharpZipLib and uncomment the define at the top of Si
}

public void SaveToCompressedFile(string aFileName)
{
    throw new Exception("Can't use compressed functions. You need include the SharpZipLib and uncomment the define at the top of Si
}

public string SaveToCompressedBase64()
{
    throw new Exception("Can't use compressed functions. You need include the SharpZipLib and uncomment the define at the top of Si
}
```

Итоговая оценка 4 из 10