



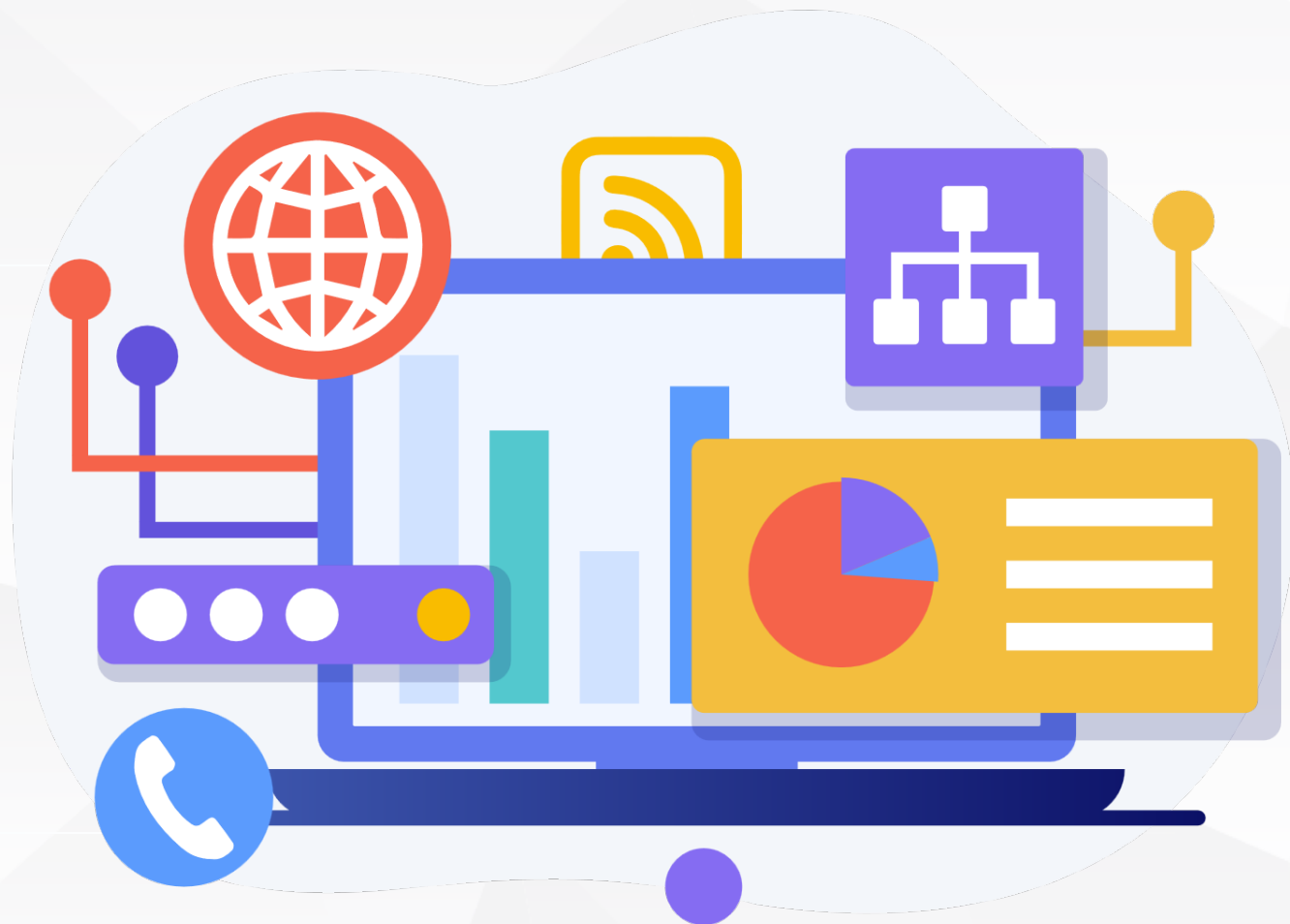
中国科学技术大学

University of Science and Technology of China

计算机网络 实验2说明

GBN Sender Lab

2023.10.18





中国科学技术大学
University of Science and Technology of China

目录

01 实验概述

02 实验准备

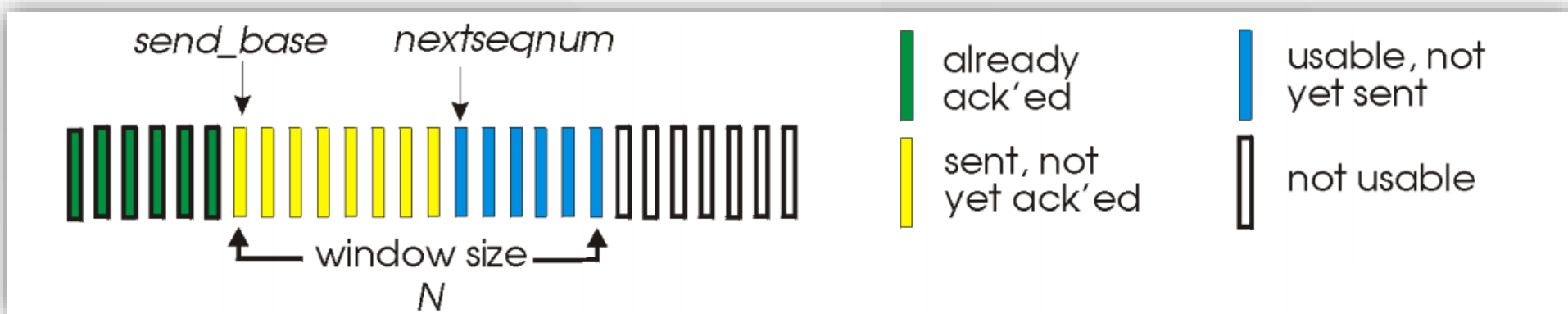
03 实验任务

04 实验提交



■ 实验背景

- Go-Back-N是自动重复请求(ARQ)协议的一个特定实例,在该协议中,即使没有收到来自接收方的确认(ACK)包,发送过程也会继续发送由窗口大小指定的分组数。它是一般滑动窗口协议的一种特殊情况,发送窗口大小为N,接收窗口大小为1。当收到接收方的确认时,发送方将窗口向后移动,并且发送其他可以发送的分组。



GBN具体相关知识,请仔细阅读PPT《第三章 传输层》第3.4节51页~54页,以及教材《计算机网络—自顶向下方法》145页~147页。



■ 实验目标

- 本实验需要实现一个搭建在OpenNetLab上的使用GBN协议的发送端。这个发送端需要将一个字符串中的每个字符封装为分组发送给接收端，并且遵循GBN协议：
 - 开始时发送滑动窗口内所有可以发送的分组。
 - 每发送一个分组，保存该分组在缓冲区中，表示已发送但还未被确认。
 - 缓冲区使用一个定时器，当定时器超时的时候，重新发送缓冲区中的所有分组。
 - 当收到接收方的确认后，判断该确认是否有效。如果无效的话，什么也不做；如果有效的话，采取累计确认，移动滑动窗口，将已经被确认的分组从缓冲区中删除，并且发送接下来可以发送的分组，重置定时器。



你可以通过 [GBN交互演示](#) 来帮助你更好地理解GBN的过程



■ 实验环境准备

- 注册OpenNetLab在线测评平台账号，网址: [[OpenNetLab](#)]。
- 实验环境：Linux操作系统，Python版本3.6以上
 - 推荐1：学校VLAB实验平台创建虚拟机，网址: [[VLAB实验中心](#)]。
 - 推荐2：VMware Workstation或VirtualBox安装Linux操作系统
- 在Linux系统中安装onl库：

```
$ pip install onl-edu
```



OpenNetLab的具体使用请参考《网络教学平台学生使用手册》



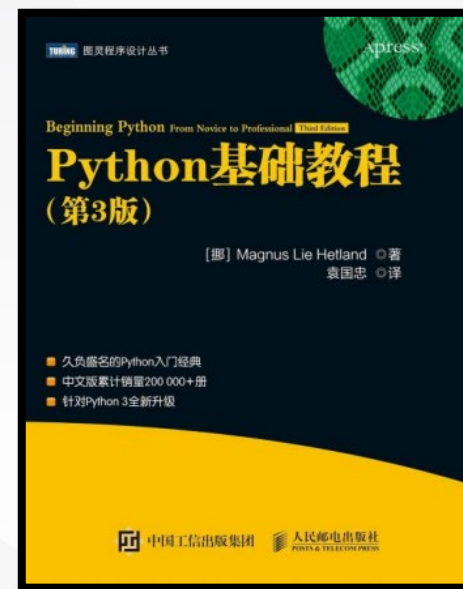
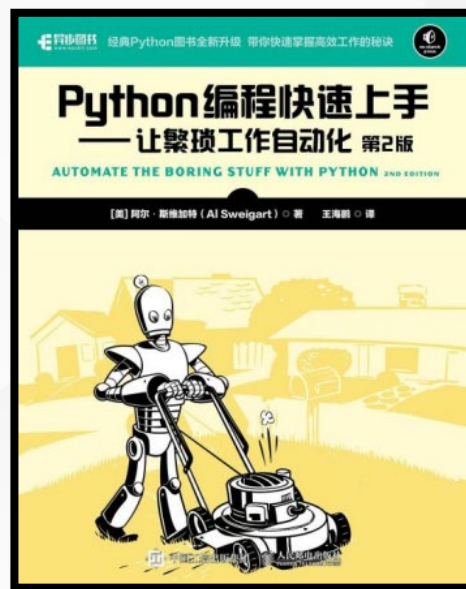
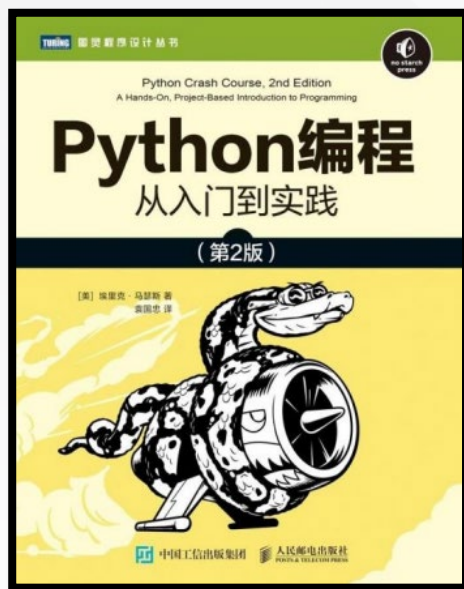
实验准备



中国科学技术大学
University of Science and Technology of China

■ 编程准备（对于Python零基础的同学）

■ Python入门书籍



■ 菜鸟教程，网址：[\[Python教程 \]](#)。

■ bilibili、YouTube上各种Python入门课程。

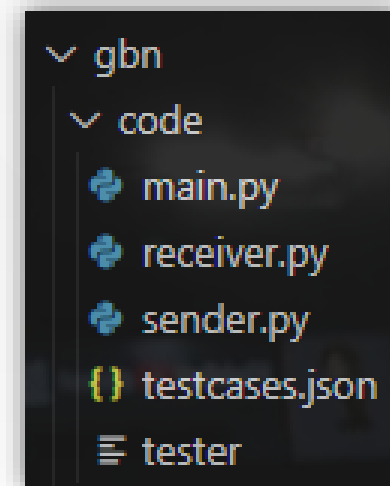


■ 实验相关资源

■ GBN实验代码下载链接: [[gbn.zip](#)]，解压后进入gbn文件夹，其中包含code文件夹：

➤ code文件夹：包含基础的实验代码模板。

- main.py：本地调试运行文件。
- receiver.py：接收方文件，无需修改。
- sender.py：发送方文件，TODO部分待编写。
- testcases.json：本地评测配置文件。
- tester：本地评测运行文件，具体使用见说明文档。



■ OpenNetLab实验相关说明文档链接: [[OpenNetLab Documentation](#)]，包含完成该实验所需阅读的相关文档。



■ 学生需要完成的工作

- 学习GBN相关基础知识，仔细阅读GBN实验手册: [[GBN Lab Manual](#)]。
- 完成编写sender.py代码模板中 TODO 部分的代码片段。
- 进行本地测试。
- 将sender.py代码提交到在线平台进行远程评测。
- 完成实验报告，准备实验结果展示。



实验任务



中国科学技术大学
University of Science and Technology of China

伪代码参考

```
function run() {
    # send available packets, add these packets to buffer
    while (sequence number is available) {
        encapsulate messages into new packets
        send new packets
        add these packets to buffer
    }
    reset timer
    Check whether to terminate the sending process
}

function put(packet) {
    # Process packets received from the receiver
    if ackno is valid {
        remove acked packets from the buffer
    }
    send the following available packets
    if all packets are sent and acked {
        Inform to terminate the sending process
    }
}

function timeout_callback() {
    for all packets in the buffer {
        send packet
    }
}
```

TODO函数模板

```
1  def run(self, env: Environment):
2
3      """
4      TODO:
5          (1) 检查滑动窗口是否已满，来产生分组并发送（发送滑动窗口内所有可以发送的分组）
6          (2) 每发送一个分组，保存该分组在缓冲区中，表示已发送但还未被确认
7          (3) 记得在规定的时机重置定时器: self.timer.restart(self.timeout)
8      """
9
10     """
11     通过`self.finish_channel.get()`获取状态
12     即当`self.finish_channel.put(True)`时发送端模拟结束
13     """
14     yield self.finish_channel.get()
15
16     def put(self, packet: Packet):
17         """从接收端收到 ACK"""
18         ackno = packet.packet_id
19
20         """
21         TODO:
22             (1) 检查收到的 ACK
23             (2) 采取累积确认，移动滑动窗口，并发送接下来可以发送的分组
24             (3) 重置定时器: self.timer.restart(self.timeout)
25             (4) 检查是否发送完 message，若发送完毕则告知结束: self.finish_channel.put(True)
26         """
27
28     def timeout_callback(self):
29         self.dprint("timeout")
30
31         """
32         TODO:
33             (1) 超时重传所有已发送但还未被确认过的分组
34             (2) 注意这个函数结束会自动重置定时器，不用手动重置
35         """
```



■ 进行本地测试

- 在本机运行 `main.py` 程序。`main.py` 程序会使用本地的一个测试用例对GBN发送端的正确性进行评测并输出运行日志。

```
$ python3 main.py
```

- 完成程序基本功能的调试后，可运行执行测试文件`tester`进行多个测试用例评测，更详细的使用说明见相关文档。

```
$ ./tester
```



调试时，可在自己的代码里使用已经定义好的`dprint`函数，输出运行过程相关信息，例：

```
self.dprint("timeout")
```



实验任务



中国科学技术大学
University of Science and Technology of China

■ 平台提交代码进行远程评测

- 在OpenNetLab上选择要提交的实验
- 输入并提交代码

❗ 把编写完成的 `sender.py` 文件里的所有内容复制进去

#	Title	Sender
	DNS Server Lab	DNS Sender
	GBN Sender Lab	GBN Sender
	SR Sender Lab	SR Sender

点击GBN Sender

点击Labs

Code - 1 sender.py

Language: Python3 Theme: Solarized Light

1

← 添加代码

提交代码 →

Submit

✓ You have submitted a solution.



实验任务



中国科学技术大学
University of Science and Technology of China

■ 平台提交代码进行远程评测

■ 查看评估结果

Lab Submissions							Status ▾	<input type="radio"/> All	<input type="text" value="Search Author"/>	<input type="text" value="Search Labs"/>	<input type="button" value="Refresh"/>
When	ID	Status	Lab	Score	Language	Author					
2023-9-14 22:14:18	e9c585fdf5bc	Judging	GBN Sender	0	Python3	rootz					
2023-9-14 22:13:20	713765bad752	PROGRAM TIMEOUT	GBN Sender	0	Python3	rootz					
2023-9-14 22:12:49	eb255cecb3b9	ALL PASSED	GBN Sender	100	Python3	rootz					
2023-9-14 22:12:32	d7bc79f4a06e	Judging	DNS Sender	0	Python3	rootz					

Overview

Announcements

Labs

Submissions

点击Submissions



■ 1.在OpenNetLab上提交代码 截止时间：2023.12.31 23:59

- 代码提交次数与测试样例通过率会影响实验最终评分。
- 推荐进行本地调试，不鼓励根据远程测试样例结果进行调试。

禁止抄袭代码！单人独立完成！



临近DDL平台容易崩溃导致提交失败，请勿踩点提交。

不接受以踩点失败、提交失败为理由的补交。



■ 2. 完成实验报告 截止时间：2023.12.31 23:59

- 在bb系统作业区提交实验报告，包括
 - 设计结果：对TODO代码片段进行解释说明
 - 实现结果：本地测试结果截图、OpenNetLab评估结果截图
- 实验报告单人独立完成！
- 报告文件命名格式：学号_姓名_实验二.pdf/docx
- 格式要求：PDF或Word

■ 3. 实验结果展示 展示时间：待定

- 进行实验结果展示，包括
 - TODO代码片段解释说明
 - 程序本地运行说明



联系方式



中国科学技术大学
University of Science and Technology of China

■ 实验相关问题联系助教

■ 实验2 GBN内容相关问题:

- 钉钉群搜索@：助教-郑博文
- 邮箱：bwzheng@mail.ustc.edu.cn

■ 实验评测平台相关问题:

- 钉钉群搜索@：助教-杨帆
- 邮箱：yangfan@mail.ustc.edu.cn



中国科学技术大学
University of Science and Technology of China

End

