

《计算机系统 III》项目作业一要求

一、项目作业准备

1. 在 <http://csapp.cs.cmu.edu/3e/students.html> (或 <http://csapp.cs.cmu.edu/3e/code.html>) 找到:

Chapter 12: Network Programming

- The Tiny Web server ([tiny.tar](#)).
Expands with `tar xvf tiny.tar` into a directory called `./tiny` that contains everything you need to test the Tiny web server, including sample HTML files, GIFS, CGI scripts, and the `csapp.c` and `csapp.h` files. Compiled and tested on Linux boxes.
- The authoritative list of [MIME types](#).

下载 `tiny.tar` 并将包中的 `tiny` 目录解压到“~/姓名/hw1”目录下, 用 `dir` 命令查看; 进入~/姓名/hw1/tiny 目录, 用 `ls` 命令查看。

其中, README 文件内容如下:

Tiny Web server

Dave O'Hallaron

Carnegie Mellon University

This is the home directory for the Tiny server, a 200-line Web server that we use in "15-213: Intro to Computer Systems" at Carnegie Mellon University. Tiny uses the GET method to serve static content (text, HTML, GIF, and JPG files) out of `./` and to serve dynamic content by running CGI programs out of `./cgi-bin`. The default page is `home.html` (rather than `index.html`) so that we can view the contents of the directory from a browser.

Tiny is neither secure nor complete, but it gives students an idea of how a real Web server works. Use for instructional purposes only.

The code compiles and runs cleanly using `gcc 2.95.3` on a Linux 2.2.20 kernel.

To install Tiny:

Type `"tar xvf tiny.tar"` in a clean directory.

To run Tiny:

Run `"tiny <port>"` on the server machine,
e.g., `"tiny 8000"`.

Point your browser at Tiny:

static content: `http://<host>:8000`

dynamic content: `http://<host>:8000/cgi-bin/adder?1&2`

Files:

`tiny.tar` Archive of everything in this directory

tiny.c	The Tiny server
Makefile	Makefile for tiny.c
home.html	Test HTML page
godzilla.gif	Image embedded in home.html
README	This file
cgi-bin/adder.c	CGI program that adds two numbers
cgi-bin/Makefile	Makefile for adder.c

2. 使用 make 命令进行构建。

(1) 生成前

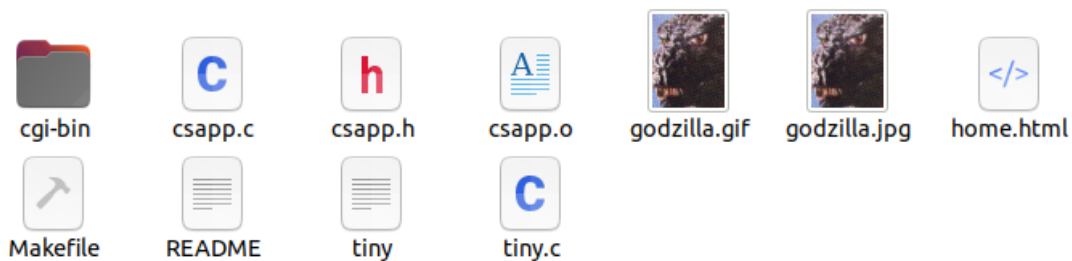
```
likecure@likecure-virtual-machine:~/Desktop/test$ tar xvf tiny.tar
tiny/
tiny/csapp.c
tiny/csapp.h
tiny/godzilla.gif
tiny/cgi-bin/
tiny/cgi-bin/adder.c
tiny/cgi-bin/Makefile
tiny/README
tiny/Makefile
tiny/home.html
tiny/godzilla.jpg
tiny/tiny.c
likecure@likecure-virtual-machine:~/Desktop/test$
```

(2) 生成

在终端下进入 tiny 目录：

```
$make
```

生成后 tiny 目录文件：



3. 启动 tiny web 服务

在终端下进入 tiny 目录：

```
$./tiny 8080
```

```
likecure@likecure-virtual-machine:~/Desktop/test/tiny$ ./tiny 8080
```

此时 tiny 在 8080 端口下启动服务，等待客户端访问。

二、项目作业任务

1. 单选题（共 6 小题，每个小题 5 分，共 30 分）

(1) 关于 TINY 的主程序，下列说法不正确的是（ ）

- A. TINY 程序的合法参数共 2 个，其中第二个参数为端口，TINY 监听这个端口上的连接请求

- B. TINY 执行典型的无限服务器循环，不断接受连接请求，执行事务
 - C. 主函数调用 `open_listenfd` 函数，用来打开一个监听套接字
 - D. 由于 TINY 是一个迭代服务器，因此它不需要被关闭
- (2) `doit` 函数的作用是 ()
- A. 处理多个 HTTP 事务
 - B. 处理一个 HTTP 事务
 - C. 处理 HTTP 请求
 - D. 处理 HTTP 响应
- (3) 关于 `void doit(int fd)`，其中参数 `fd` 为 ()
- A. 主动套接字(active socket)
 - B. 监听套接字(listening socket)
 - C. 已连接描述符(connected descriptor)
 - D. 未连接描述符(unconnected descriptor)
- (4) `read_requesthdrs` 函数的作用是 ()
- A. 读取 http 请求报文的请求行并输出
 - B. 读取 HTTP 请求报文的首部行部分并输出
 - C. 读取 HTTP 请求报文的实体体部分并输出
 - D. 读取 HTTP 请求报文所有部分并输出
- (5) 关于 `parse_uri` 函数，下列说法不正确的是 ()
- A. 其形参 `uri` 表示传入的 http 请求行中的 url 字段值
 - B. 若为动态内容请求，返回 1；若为静态内容请求，返回 0
 - C. 如果客户端请求的 URL 为：`http://localhost:8080/`，则参数 `filename` 的最终值为“`./home.html`”
 - D. 如果客户端请求的 URL 为：`http://localhost:8080/cgi-bin/adder?2&3`，则参数 `cgiargs` 的最终值为“`2&3`”
- (6) 关于 `serve_dynamic` 函数，下列说法错误的是 ()
- A. 该函数通过派生一个子进程并在子进程上下文中运行一个 CGI 程序，来提供动态内容响应
 - B. 该函数中将 `cgiargs` 设置为环境变量，以便在子进程中读取和使用
 - C. 该函数中调用的 `Dup2` 函数，用于将子进程的标准输出到已连接文件描述符上，这样子进程写到标准输出上的任何内容都会直接送到客户端进程，不会受到来自父进程的干涉
 - D. http 响应报文中的状态行，也是由子进程负责发送给客户端的

2. 问答题 (共 4 小题，共 70 分)

(1) 结合 HTTP 请求报文格式，以及 TINY 中 `doit` 函数源码 (代码及行号见教材图 11-30)，回答如下问题：(共 25 分)

- 1) 源码中哪一行用来读取一个 http 请求报文中的“请求行”？(5 分)
- 2) 结合 HTTP 请求报文格式，说明源码第 14 行的语义是什么？(10 分)
- 3) 源码第 15-19 行的语义是什么？(5 分)
- 4) 源码第 23 行的语义是什么？(5 分)

(2) 结合 HTTP 请求报文格式，以及 TINY 中 `read_requesthdrs` 函数源码 (代码及行号见教材图 11-32)，回答如下问题：(共 10 分)

1) 源码第 6-9 行的语义是什么? (5 分)

2) 若 buf 值为"\r\n"时, 结合 http 请求报文格式, 请说明当前读取到了 http 请求报文的什么位置? (5 分)

(3) 结合 HTTP 响应报文格式, 以及 TINY 中 `serve_static` 函数源码 (代码及行号见教材图 11-34), 回答如下问题: (共 15 分)

1) 源码中哪一行用来设置 http 响应报文中的“状态行”? (5 分)

2) 源码第 9-12 行的语义是什么? (5 分)

3) 源码第 13 行的语义是什么? (5 分)

(4) 结合 `get_filetype` 函数源码 (代码及行号见教材图 11-34), 回答如下问题: (共 20 分)

1) TINY 能够解析的静态文件类型有哪些? (5 分)

2) 如何需要修改 TINY, 使之能够处理 .htm 的文件类型, 请问需要修改哪个函数的代码? 请给出具体修改方案, 并给出相应的运行截图。(15 分)