

ICS Lab1-DataLab

Deadline: 2022-10-05 23:59:59

一、实验简介

CSAPP 第一章配套实验。

本实验的目的是加深同学们对整数和浮点数二进制表示的认识。同学们需要解出若干程序谜题，编写代码并通过正确性测试，最后提交代码和报告。希望同学们多加思考，在解题过程中能学到的远不止二进制本身，还能加深对位运算的理解，以及学到一些算法知识。

二、部署实验环境

(1) 下载

从 `elearning` 下载 `datalab-handout.tar`。这是一个 `tar` 文件，需要对其解包。

(ubuntu 虚拟机 or WSL) 打开终端，进入到上述文件对应的目录下，然后执行如下命令：

```
tar -xvf datalab-handout.tar
```

会在当前目录解包出一个 `datalab-handout` 文件夹，其中的内容就是本次实验用到的文件了。

(2) 准备工作

确保已安装了 gcc

在终端中检查是否安装了 gcc：

```
gcc -v
```

如果已安装，终端将会反馈版本信息，否则会反馈 `command not found`。

如未安装，尝试执行以下命令进行安装：

```
sudo apt-get install gcc
```

确保已安装了 make

检查是否安装 make，在终端输入：

```
make -v
```

同理，如未安装，尝试以此执行以下命令：

```
sudo apt-get update
sudo apt-get install make
sudo apt-get install libc6 libc6-dev libc6-dev-i386
```

确保实验文件能正常 make 并运行测试

在终端中进入 `datalab-handout/` 文件夹目录。

键入 `ls` 查看内容，你应当看到如下文件：

```
Driverhdrs.pm  Makefile  bits.c  btest.c  decl.c  driver.pl  ishow.c
Driverlib.pm   README    bits.h  btest.h  dlc      fshow.c   tests.c
```

在终端中依次执行如下命令，以生成可执行文件并执行：

```
make clean
make all
./btest
```

如果过程顺利，最后会输出测试结果，最后一行为 `Total points: 0/49`。

如果过程中出现如下报错：

```
/usr/bin/ld: cannot found -lgcc
```

那么尝试执行：

```
sudo apt install gcc-multilib
```

确保dlc (data lab compiler) 能正常执行

依然在 `datalab-handout/` 目录下，

执行：

```
./dlc bits.c
```

如果执行立即结束并且没有任何反馈，则没有问题。

三、实验提示与说明

(1) 如何入手

推荐阅读顺序：本文档 > `README` > `bits.c` 的注释部分。

`README` 文档中对实验文件做了较为详细的介绍。

`bits.c` 是同学们唯一需要编辑的文件，其中包含了谜题规则介绍，18 个谜题以及谜题内容、难度、分数等。

上面的两个文件请务必仔细阅读。

注意到，每一个谜题包含了如下信息：

- 能使用的运算符。
- 能使用的运算符总数量。
- 能使用的常数的值域范围。
- 变量类型。
- 能否使用控制语句（如 `if`）等。

(2) 测试

- 除了 `bits.c`，你不应该编辑任何其余文件。
- 完成谜题后，需要检验自己的代码是否正确：
 - 首先，执行下述指令，检查每个函数使用的运算符类型、数目是否符合要求：

```
./dlc -e bits.c
```
 - 如没有任何报错，按顺序执行下述指令，测试每个函数的正确性：

```
make clean
make all
./btest
```
 - `btest` 执行时会给出每个谜题（函数）是否通过测试（未通过时会给出测试数据），并且会计算你的最终得分。
- 你可以利用上 `./ishow` 和 `./fshow` 来帮助你调试（用法见 `README`）。

四、提交实验

(1) 内容要求

你需要提交至少两份文件，包含你的 `bits.c` 和一份实验报告。

实验报告应该包含以下内容：

- 实验标题，你的姓名，学号。
- 你在终端中执行 `./dlc -e bits.c` 后的截图。
- 你在终端中执行 `./btest` 后的截图。
- 描述你实现每个函数的思路，要求简结清晰。
- 如果有，请务必在报告中列出引用的内容以及参考的资料。
- 对本实验的感受（可选）。
- 对助教们的建议（可选）。

(2) 格式要求

可提交 `.md` 文件或者 `.pdf` 文件。不要提交 `.doc` 或 `.docx` 文件。

将你的 `bits.c` 和实验报告打包成 `tar` 文件。

可按如下流程操作：

1. 新建文件夹，以你的学号命名（例如 `19307130237`）。将你要提交的两份文件放入文件夹中。
2. 在终端中进入到目录 `19307130237/`，执行 `cd ..` 返回到上级目录，然后：

```
tar -cf 19307130237.tar 19307130237/
```

这将得到一个名为 `19307130237.tar` 的文件。（该学号仅为示例，请使用你的学号）

3. 在 elearning 上对应作业处提交打包完成的 `tar` 文件。

(3) 打分规则

- 实验谜题的正确性占比 70%，实验报告和代码风格占 30%。（C语言和C++不同，一个作用域下的所有局部变量的声明应该放在最上面）
- 实验报告的字数和打分无明显关系，报告的内容清晰即可，过于冗长的报告可能是减分项。
- 对每道谜题请先做思考，不要立即使用搜索引擎。
- 不允许抄袭其它人的代码，一旦发现，零分处理。

五、参考资料

- <http://csapp.cs.cmu.edu/3e/labs.html>（lab来源，本实验相较原文件有改动）
- 本文档编写时参考了20年的实验文档。