

15-213, 20xx年秋季
数据实验室：操纵比特
指定日期：8月30日，到期日期：星期三，9月
12日晚上11时59分

哈里·博维克(bovik@cs.cmu.edu)是这项任务的负责人。

1 引言

本次作业的目的是更加熟悉整数和浮点数的位级表示.. 你将通过解决一系列编程“谜题”来做到这一点。这些谜题中的许多都是人造的，但你会发现自己在用你的方式处理它们时会更多地比特。

2 后勤

这是一个单独的项目。所有的手都是电子的。澄清和更正将张贴在课程网页上。

3 分发说明

网站-特殊：在这里插入一个段落，解释教师将如何向学生分发datalab-handout.tar文件。

首先，将datalab-handout.tar复制到Linux机器上的（受保护的）目录，您计划在其中执行您的工作。然后下达命令。

```
Unix>tarX VF数据处理。
```

这将导致在目录中解压缩许多文件。您将修改和输入的唯一文件是bit.c。

bit.c文件包含13个编程谜题中的每一个的骨架。您的任务是只使用整数拼图的直线代码（即没有循环或条件）和有限数量的C算理和逻辑运算符来完成每个函数骨架。具体来说，您只允许使用以下八个操作符：

! ~&^|+<<>>

一些函数进一步限制了此列表。此外，不允许使用超过8位的任何常量。有关详细规则和对所需编码样式的讨论，请参阅bit.c中的注释。

4 困惑

本节描述您将在bit.c中解决的谜题。

表1按难度从最容易到最难的顺序列出了谜题。“评级”字段给出拼图的难度等级（点数），“Maxops”字段给出允许使用的最大操作符数来实现每个函数。有关函数的期望行为的更多细节，请参见bit.c中的注释。您还可以参考test.c中的测试函数。这些函数被用作引用函数来表示函数的正确行为，尽管它们不满足函数的编码规则。

名字	说明	评级	最大行动
位Xor(x, y)	x y只使用&和~。	1	14
泰民()	最小的两个补整数	1	4
是Tmax(X)	只有当xx是最大的两个COMP时，才是正确的。整数。	1	10
所有奇数位(X)	只有当x中所有奇数位设置为1时，才为真。	2	12
否定(X)	使用操作员返回-x。	2	5
是AsciiDigit(X)	如果0x30≤x≤为真。	3	15
有条件的	和x一样？ y：z	3	16
较少或相等(x, y)	如果x≤y，则为true，否则为false	3	24
逻辑否定(X)	计算！x不使用！操作员。	4	12
多少比特(X)	敏。没有。在两个COMP中表示x的位。	4	90
浮法2(UF)	返回位级等值。 2* f.f.p. 阿格。 f.	4	30
浮子2Int(UF)	返回位级等值。 f.p的(int)f. 阿格。 f.	4	30
浮动功率2(X)	返回位级等值。 整数x的0^x。	4	30

表1：达达拉布拼图。对于浮点拼图，值f是具有与无符号整数UF相同位表示的浮点数。

对于浮点拼图，您将实现一些常见的单精度浮点操作..对于这些谜题，允许您使用标准控制结构（条件、循环），您可以同时使用int和无符号数据类型，包括任意无符号和整数常量。您不能使用任何联合、结构或数组。最重要的是，您不能使用任何浮点数据类型、操作或常量。相反，任何浮点操作数都将作为具有类型传递给函数

无符号，任何返回的浮点值都将是无符号类型.. 您的代码应该执行实现指定浮点操作的位操作。

包含的程序fshow帮助您理解浮点数的结构。去编译显示，切换到讲义目录并键入：

```
Unix>制造
```

可以使用fshow查看任意模式表示为浮点数：

```
Unix>/fshow2080374784
```

浮点值2.658455992e+36.

位表示0x7c000000，符号=0，指数=f8，分数=000000归一化。 $100000000 \times 2^{(121)}$

还可以给出fshow十六进制和浮点值，它将破译它们的位结构..

5 评价

你的分数将根据以下分布计算出最多67分：

36正确点。

26绩效分数。

5 风格要点。

正确点。 你必须解决的谜题被赋予了1到4之间的难度等级，因此它们的加权和总计为36。我们将使用btest程序评估您的功能，这将在下一节中描述。如果一个谜题通过了btest执行的所有测试，你将获得完整的信用，否则就没有信用。

业绩要点。 我们在这一点上的主要关注是，你可以得到正确的答案。然而，我们想给你灌输一种保持尽可能短和简单的感觉。此外，有些谜题可以用蛮力解决，但我们希望你更聪明。因此，对于每个函数，我们已经建立了允许您为每个函数使用的最大数量的运算符。这一限制非常慷慨，其目的只是为了捕捉极端低效的解决方案。对于满足运算符限制的每个正确函数，您将得到两个点。

风格要点。 最后，我们保留了5分，以主观地评估您的解决方案的风格和您的评论。你的解决方案应该尽可能干净和直截了当。你的评论应该是信息丰富的，但它们不需要广泛。

自动记录你的工作

我们已经在讲义目录中包含了一些自动编辑工具-btest、dlc和driver.pl-以帮助您检查工作的正确性。

- 测试：此程序检查bit.c中函数的功能正确性。要构建和使用它，请键入以下两个命令：

```
Unix>使Unix>。  
/测试
```

请注意，每次修改bit.c文件时，必须重新构建btest。

你会发现，这是有帮助的工作，通过功能一次，测试每一个你去。可以使用-f标志指示btest只测试一个函数：.

```
Unix>/btest-f位Xor
```

您可以使用选项标志-1、-2和-3：unix>/btest-f位Xor-14-25来为它提供特定的函

数参数

检查文件自述文件以获得运行btest程序的文档。

- 这是MITC IL K组中ANS IC编译器的修改版本，您可以使用它来检查每个拼图的编码规则是否符合要求。典型用法：.

```
Unix>/DLC比特。C
```

除非程序检测到一个问题，如非法运算符、太多运算符或整数拼图中的非直线代码，否则程序将默默运行。用-e开关运行：.

```
Unix>/DLC-e比特。c
```

使DLC打印每个函数使用的操作符数目。类型。/DLC-帮助用于命令行选项列表。

- 驱动程序.pl：这是一个驱动程序，使用btest和DLC计算解决方案的正确性和性能点。它不需要争论：

```
Unix>/driver.pl
```

您的教员将使用driver.pl评估您的解决方案。

6 手提指令

网站-特殊：在这里插入文字，告诉每个学生如何交他们的比特。c
你学校的解决方案文件。

7 建议

- 不要将<stdio.h>头文件包含在bit.c文件中，因为它混淆了DLC并导致一些非直观错误消息。您仍然可以在bit.c文件中使用printf进行调试，而不包括<stdio.h>头，尽管gcc将打印一个可以忽略的警告。
- dlc程序强制执行一种比c++或gcc更严格的c申报形式。特别是，任何声明必须在不是声明的任何声明之前出现在一个块中（你用大括号括起来的）。例如，它会抱怨以下代码：

```
国际货币基金组织 (INT X)
{
    =x ;
    a* =3 ;      / * 不是声明的声明* /intb=a ;    / * 错误：这里
    不允许声明*/
}
```

8 “击败教授”比赛

为了好玩，我们提供了一个可选的“击败教授”比赛，让你与其他学生和教练竞争，以开发最有效的谜题。目标是使用最少的操作符解决每个数据实验室难题。匹配或击败教练的操作员计算每个拼图的学生是赢家！

提交参赛作品，类型：.

```
Unix>/driver.pl-u`你的昵称`
```

昵称仅限于35个字符，可以包含字母数字、撇号、逗号、句号、破折号、下划线和安培数。你可以随时提交。您的最新提交将出现在实时记分板上，仅由您的昵称标识。您可以通过指向浏览器来查看记分板

```
http://$SERVER_NAME:$REQUESTD_PORT
```

标准：将\$SERVER_NAME和\$SERVER_NAMEREQUESTD_PORT替换为./contest/Contest.pm文件中设置的值。