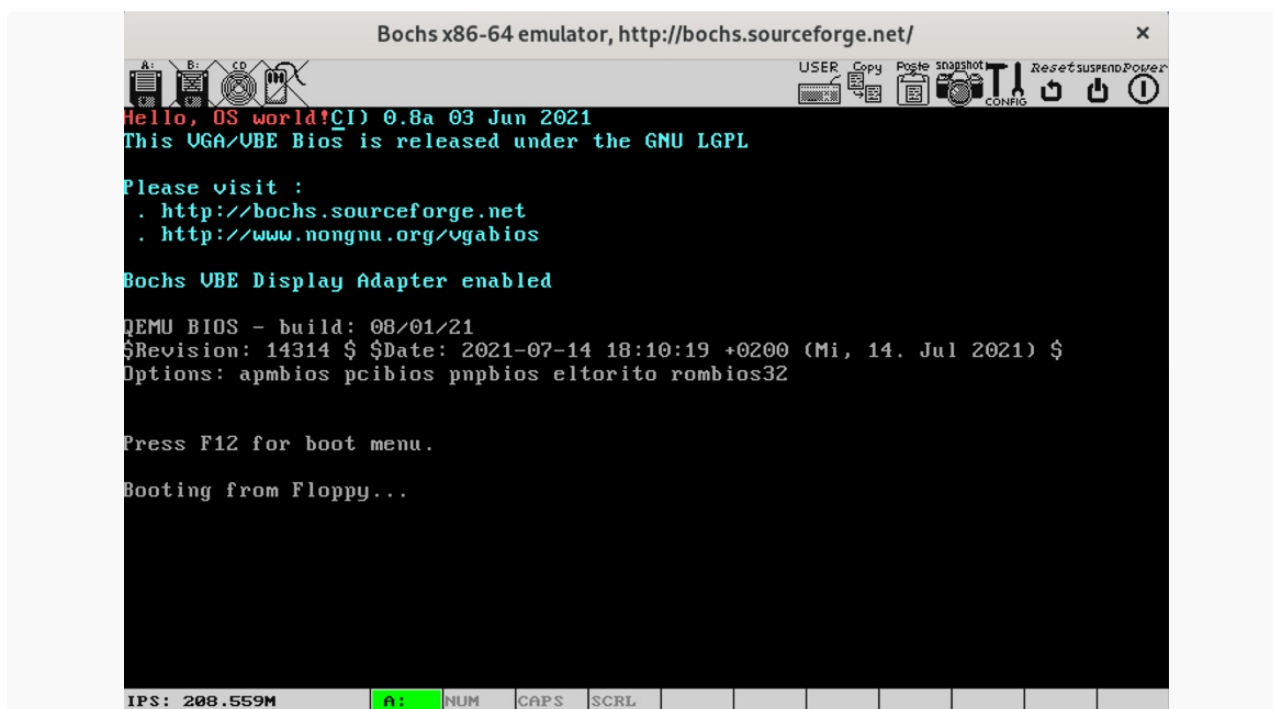


## 2022年秋季操作系统实验（一）

### 实验任务一：Hello OS（1分）

选择任意平台，参考讲义搭建NASM+Bochs实验平台，在该实验平台上汇编 `boot.asm` 并用Bochs虚拟机运行，显示“Hello OS world!”。最终运行结果如下图所示：



#### 具体要求

- 理解 `boot.asm`、`bochsrc` 中的内容；理解实验过程中涉及的所有命令、选项、参数的含义；
- 在Moodle上提交代码与运行截图。

#### 评分标准

- 在不依赖助教提示、查看讲义的情况下复现实验，得1分；
- 部分操作需要依赖助教提示或查看讲义，得0.5分；
- 大部分操作需要依赖助教提示或查看讲义，得0分。

### 实验任务二：整数加法乘法（6分）

使用汇编语言NASM实现大整数的加法和乘法。给出整数  $x, y$ ，计算  $x + y$  或  $x \times y$  的值。

#### 具体要求

- 输入输出格式：

- 输入格式：<input\_x><operator><input\_y>，两两之间没有空格，<input\_x>和<input\_y>分别代表  $x$  和  $y$ ，其中  $-10^{20} \leq x, y \leq 10^{20}$ ；<operator> 包含加法运算符 + 和乘法运算符 \*，若操作符后是负数，负号直接跟在操作符后；输入 q 结束程序；
- 输出格式：计算的结果；输出结果后继续接受下一个输入；
- 至少处理 1 种非法输入情况（如无效操作符，缺少操作数等），保证程序不崩溃。

Input	Output
1234+567	1801
1234*567	699678
1234+-567	667
-1234+-567	-1801
1234*-567	-699678
1234/567 1234+ 123.4+567	Invalid
q	程序终止

- 本实验在本机的 Linux/Windows/macOS 系统上完成，而不是在 Bochs 中完成；
- 只能使用 NASM 实现，不可使用其他语言；
- 在 Moodle 上提交代码与运行截图。

## 评分标准

- 完成  $0 \leq x, y \leq 10^{20}$  的部分即可得 6 分；
- 完成  $-10^{20} \leq x, y \leq 10^{20}$  的部分可获得 1 分的附加分，但本次实验总分不超过 10 分；
- 代码质量不高会酌情扣分；
- 无法解释代码视情况扣 1~6 分。

## 实验问题（3分）

在整个实验过程中，无论是编程还是查资料，请同学们注意思考以下问题，助教检查时会从中随机抽取若干题目进行提问，根据现场作答给出分数。请注意，我们鼓励自己思考和动手实验，如果能够提供自己的思考结果并辅助以相应的实验结果进行说明，在分数评定上会酌情考虑。

- 什么是复杂指令集？什么是精简指令集？80x86 采用的是哪种？
- 什么是小端存储？什么是大端存储？80x86 采用的是哪种？
- 8086 有哪 5 类寄存器？请分别举例说明其作用。
- 有哪些段寄存器，它们的作用是什么？
- 什么是寻址？8086 有哪些寻址方式？
- 什么是直接寻址？直接寻址的缺点是什么？
- 主程序与子程序之间如何传递参数？你的实验代码中在哪里体现的？

8. 如何处理输入和输出？你的代码中在哪里体现的？
  9. 通过什么寄存器保存前一次的运算结果？你的代码中在哪里体现的？
  10. 请分别简述 `MOV` 指令和 `LEA` 指令的用法和作用。
  11. 解释 `boot.asm` 文件中 `org 07c00h` 的作用。
  12. 解释 `boot.asm` 文件中 `times 510-($-$) db 0` 的作用。
  13. 解释 `bochsrc` 中各参数的含义。
  14. `boot.bin` 应该放在软盘的哪一个扇区？为什么？
  15. Loader 的作用有哪些？
- 

## 参考资料

- 《Orange's 一个操作系统的实现》
- NASM Docs <https://www.nasm.us/doc/>

使用 `man` 查询各种命令的用法。

如遇到实验相关问题，请在 Moodle 上发贴或向助教发邮件。