2023秋《操作系统》课程实验报告

实验二

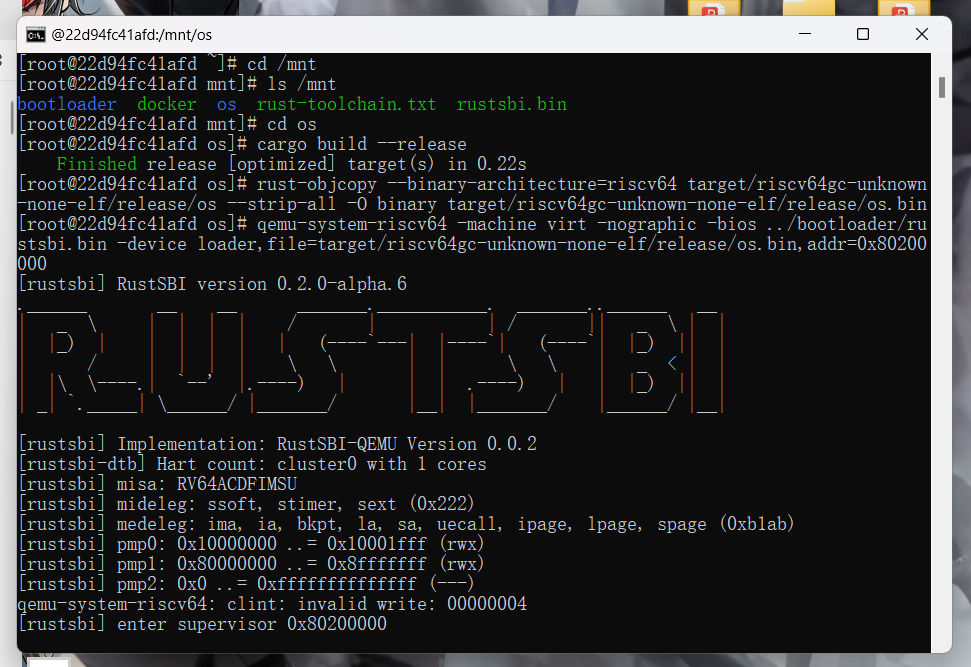
21301161 齐家昕

1. 实验步骤

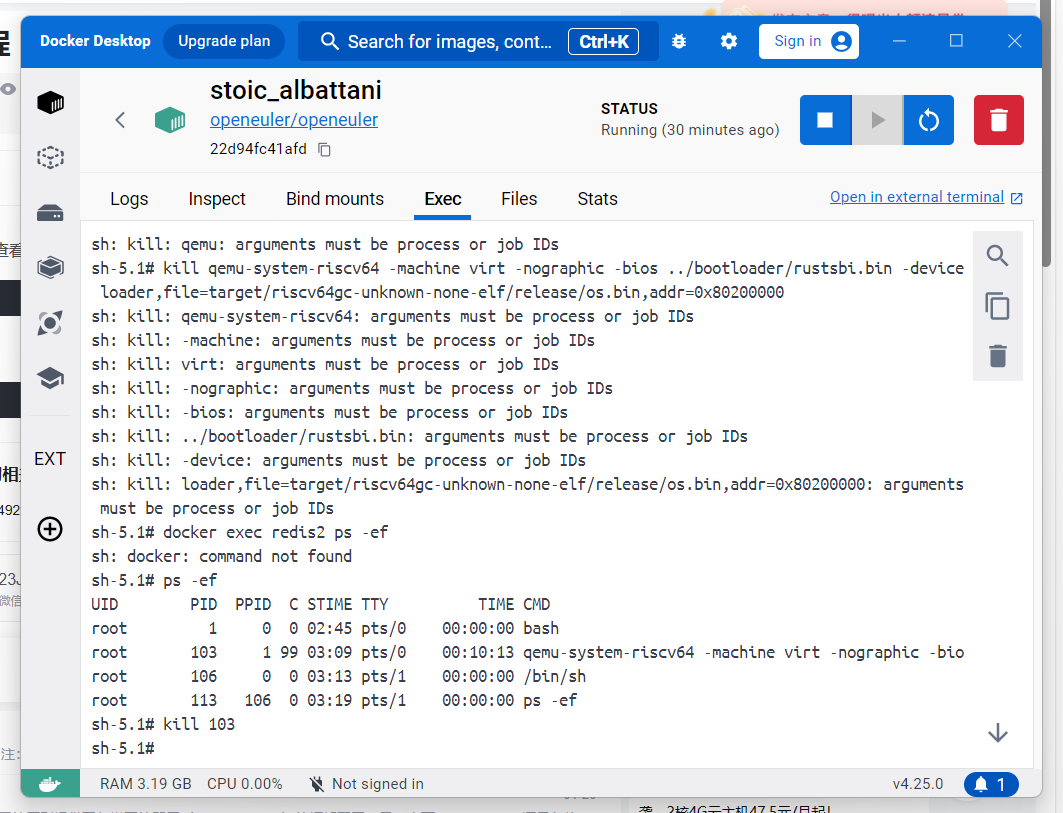
本实验的主要目的是实现裸机上的执行环境以及一个最小化的操作系统内核。

1. 编译生成内核镜像

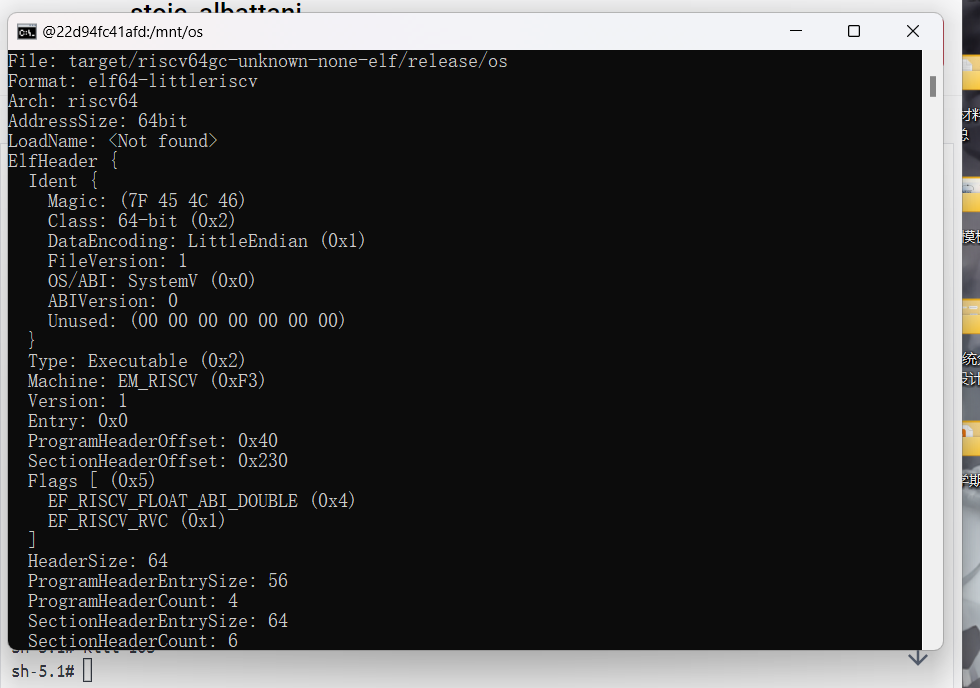
这时候运行会进入死循环，原因是操作系统的入口地址不对！



注意：退出qemu可以通过Docker Desktop里的容器的命令行杀死qemu的进程。



对于os ELF执行程序，通过rust-readobj分析，看到的入口地址不是约定的 0x80200000。

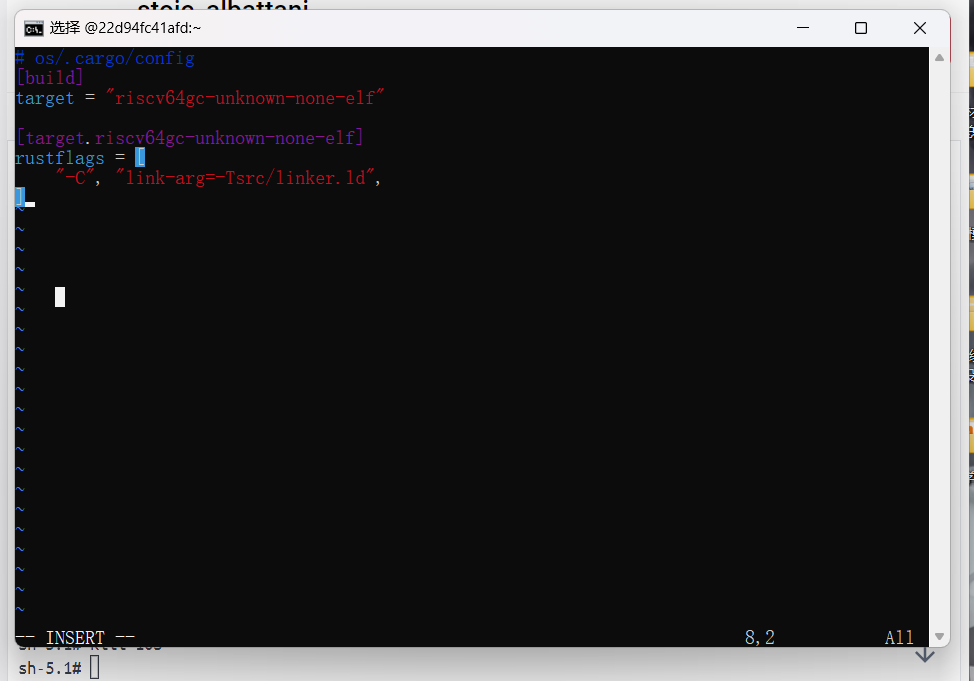


因此，我们还需要修改 os ELF执行程序的内存布局。

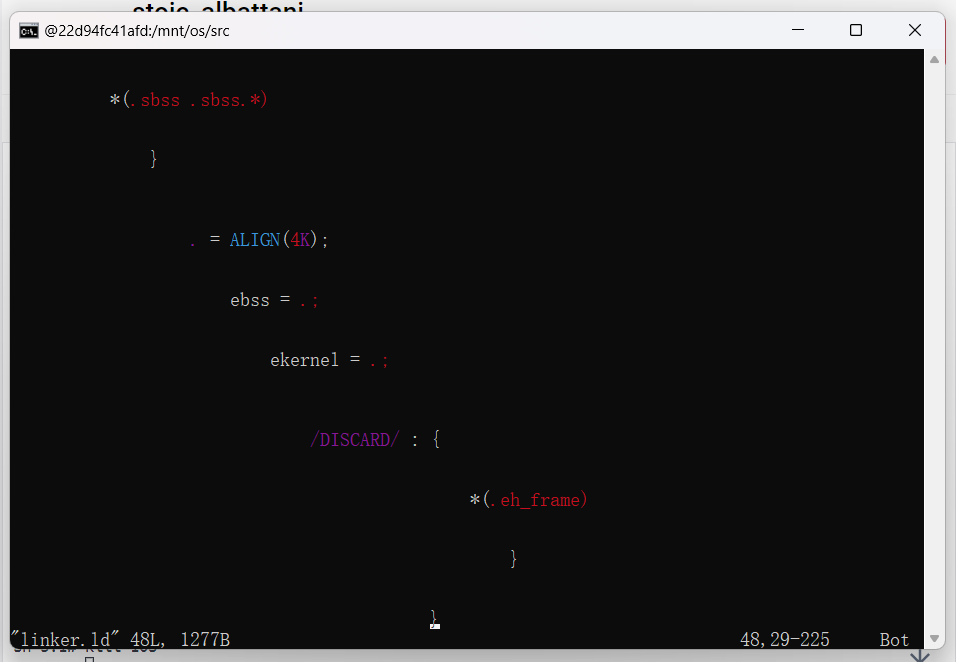
2. 指定内存布局

通过链接文件linker.ld可以实现指定可执行文件的内存布局。同时，我们还需要修改Cargo的配置文件来使用我们的链接脚本而不是默认的内存布局。

首先，修改os/.cargo/config，增加如下内容：



链接脚本文件os/src/linker.ld的内容如下：

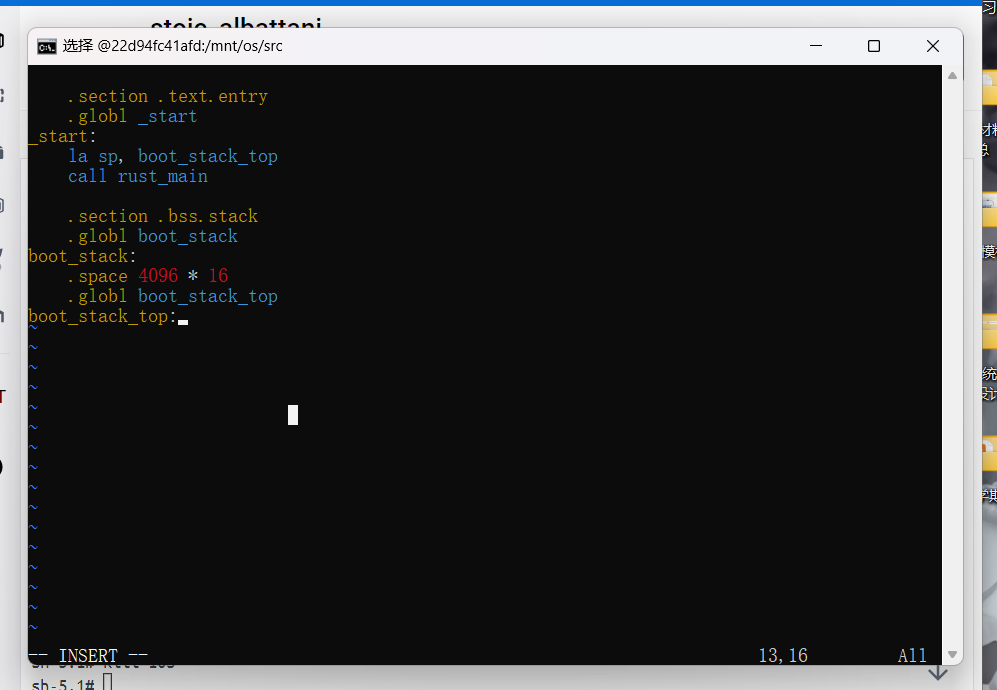


3. 配置栈空间布局

为了程序能够正确的执行，我们还需要设置正确的栈空间。

栈空间的通过汇编entry.asm来建立，文件目录为：os/src/entry.asm。

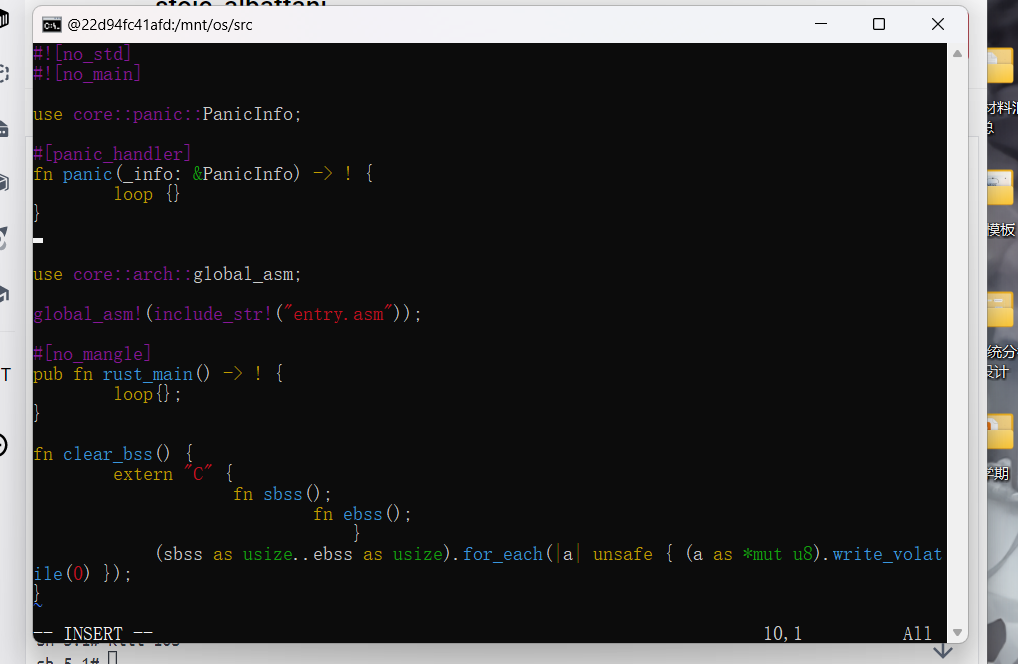
文件内容具体如下：



然后，我们还需要在 main.rs 中嵌入这些汇编代码并声明应用入口 rust\_main。

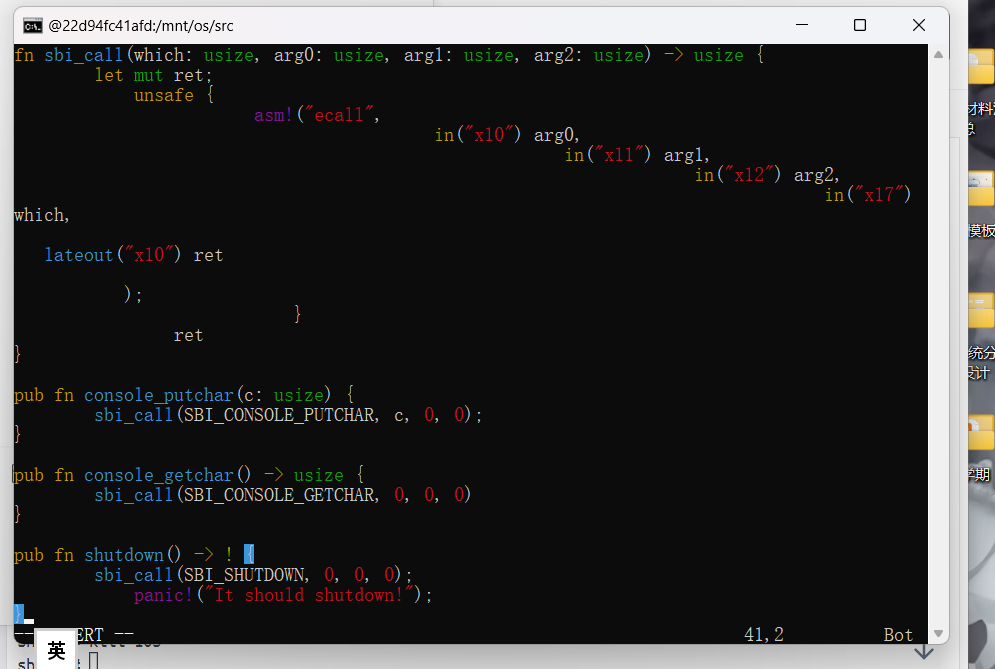
4. 清空bss段

为了保证内存的正确性，我们还需要撰写代码清空.bss段。在main.rs中增加如下代码：



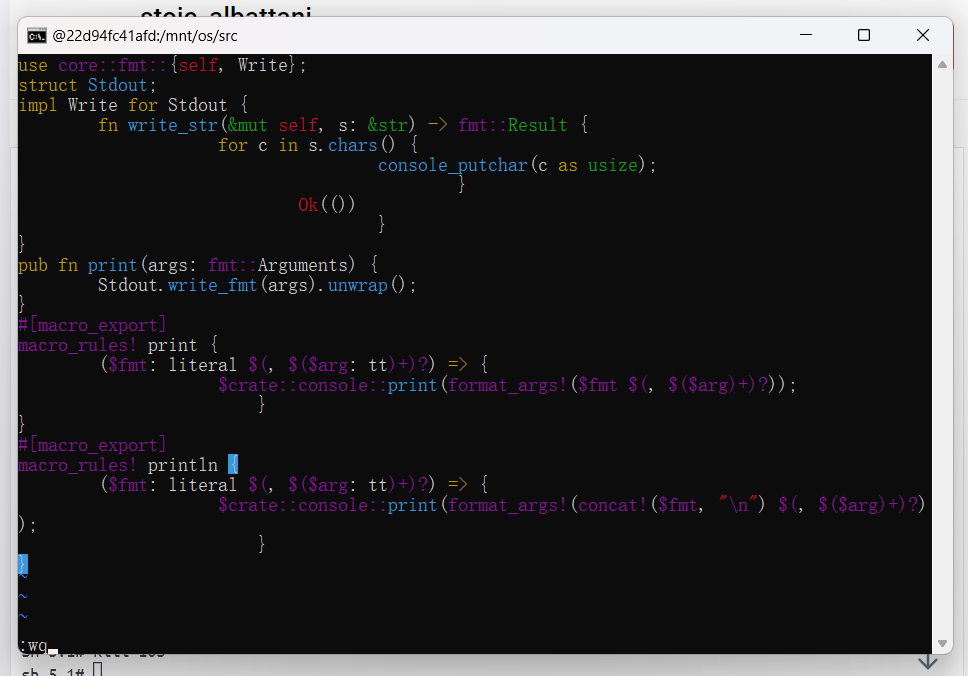
5. 实现裸机打印输出信息

为了实现在裸机上能够打印信息，我们需要把之前的系统调用改成sbi调用即可实现。同时，我们还可以调用sbi提供的接口实现关机的功能。

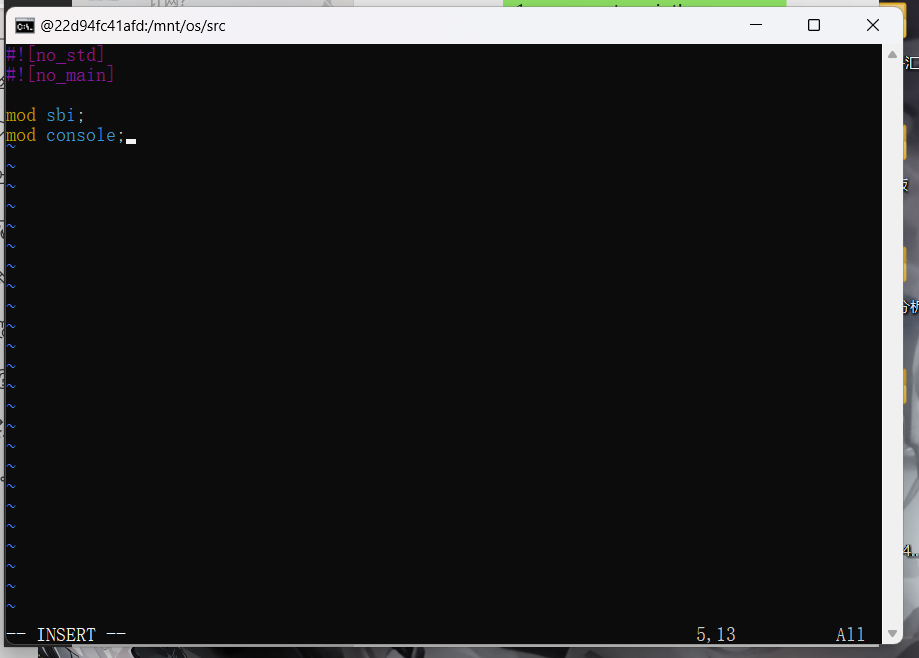


在sbi.rs提供接口的基础上，根据前一节print函数的实现，我们实现裸机上的print函数。

具体在os/src/console.rs中，其内容具体如下：



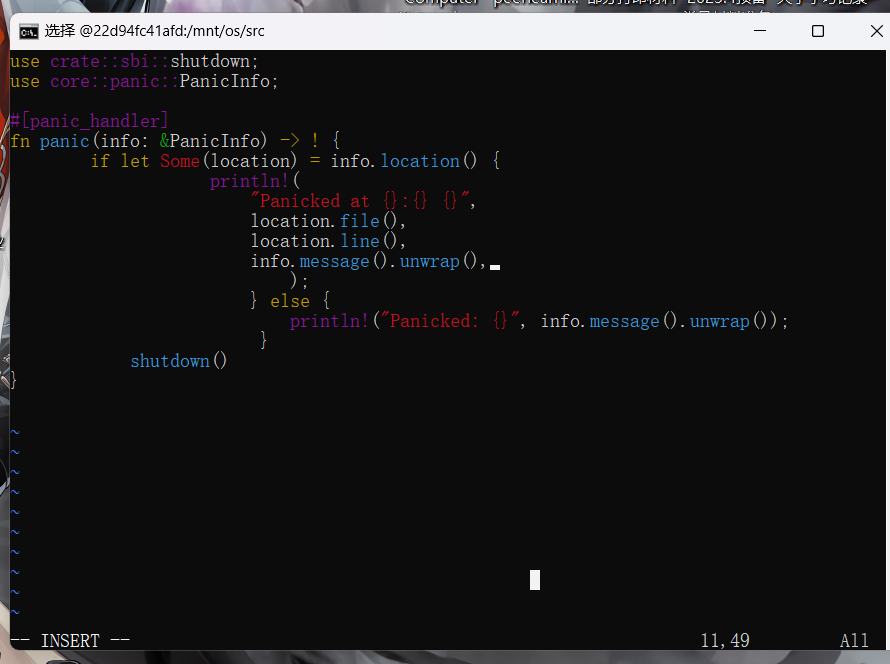
修改完上述代码后，注意需要删除main.rs中的内容，并且增加调用sbi和console两个模块。



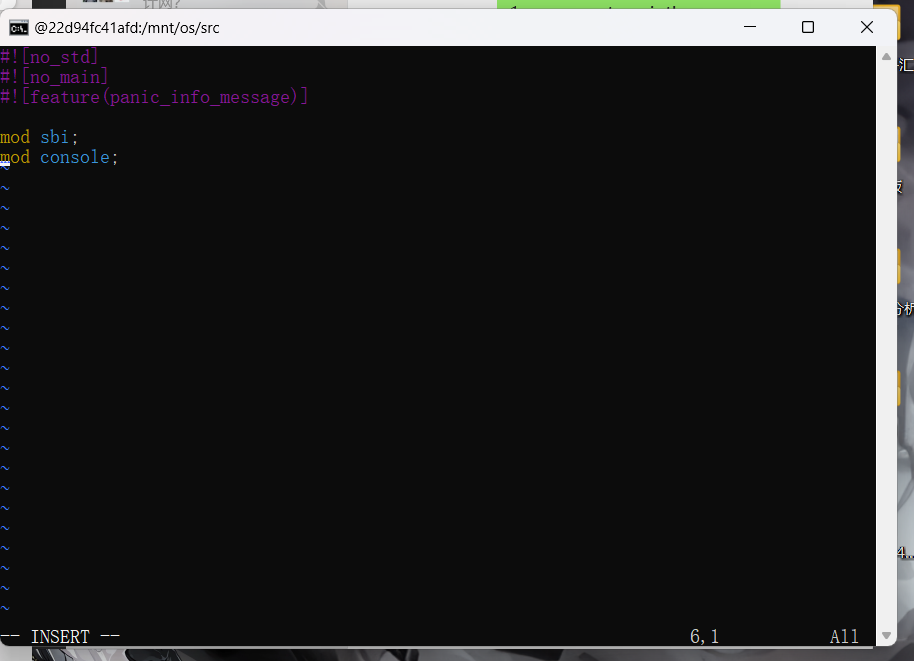
6. 给异常处理增加输出信息

最后，再给异常处理函数panic增加输出显示，以便我们更好的了解程序的执行情况。

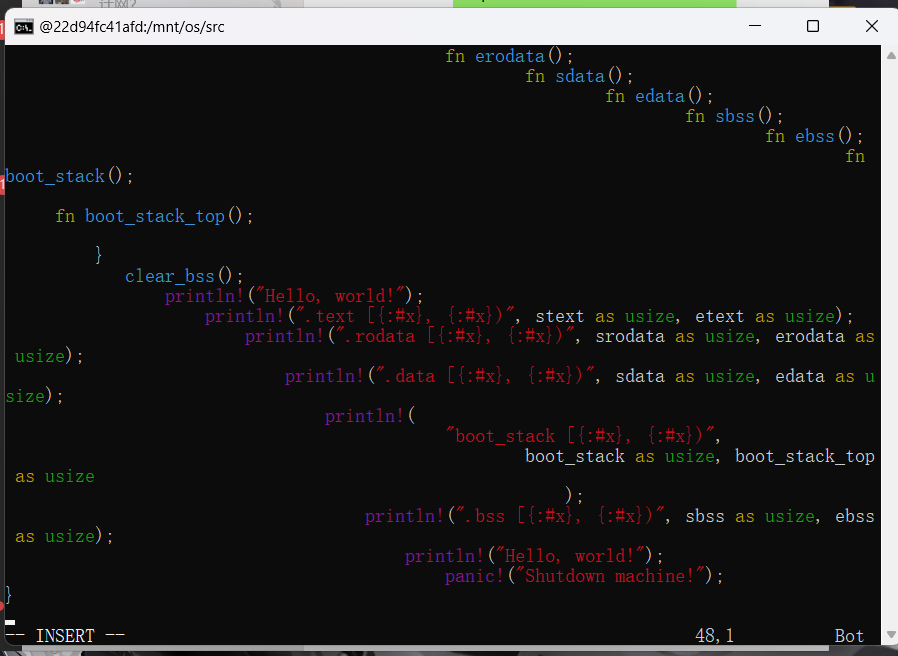
实现os/src/lang\_items.rs



注意需要在 main.rs 开头加上 #![feature(panic\_info\_message)] 才能通过 PanicInfo::message 获取报错信息。



7. 修改main.rs输出测试信息



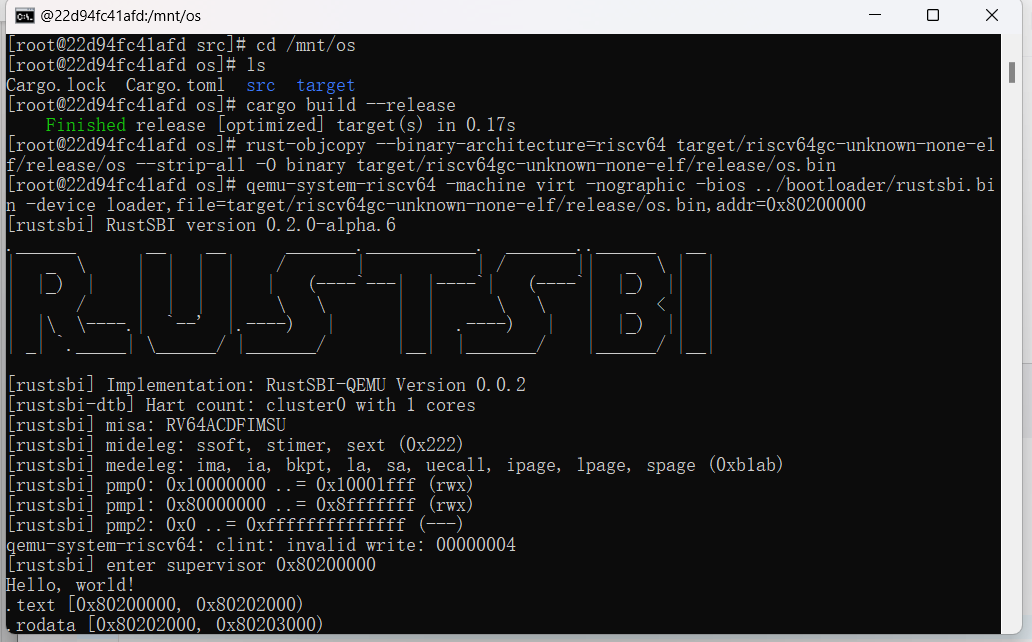
然后，重新编译以及生成二进制文件。具体步骤为：

（1）编译

cargo build --release

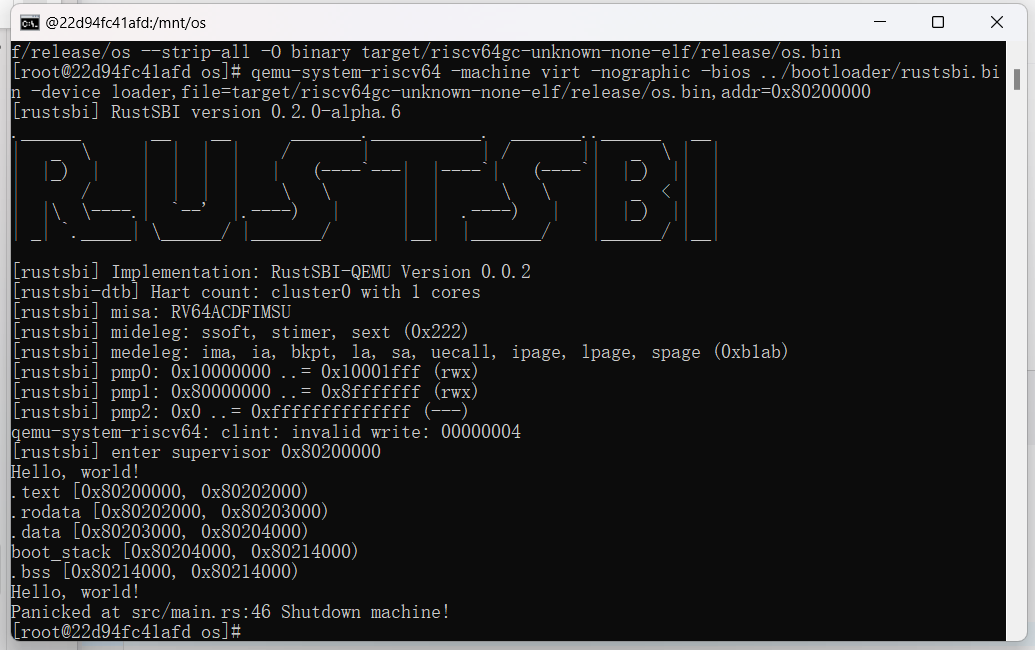
（2）生成二进制文件

rust-objcopy --binary-architecture=riscv64 target/riscv64gc-unknown-none-elf/release/os --strip-all -O binary target/riscv64gc-unknown-none-elf/release/os.bin



（3）运行

qemu-system-riscv64 -machine virt -nographic -bios ../bootloader/rustsbi.bin -device loader,file=target/riscv64gc-unknown-none-elf/release/os.bin,addr=0x80200000



同时，为了更加方便地编译运行，还可以编写一个Makefile文件。

1. 思考问题

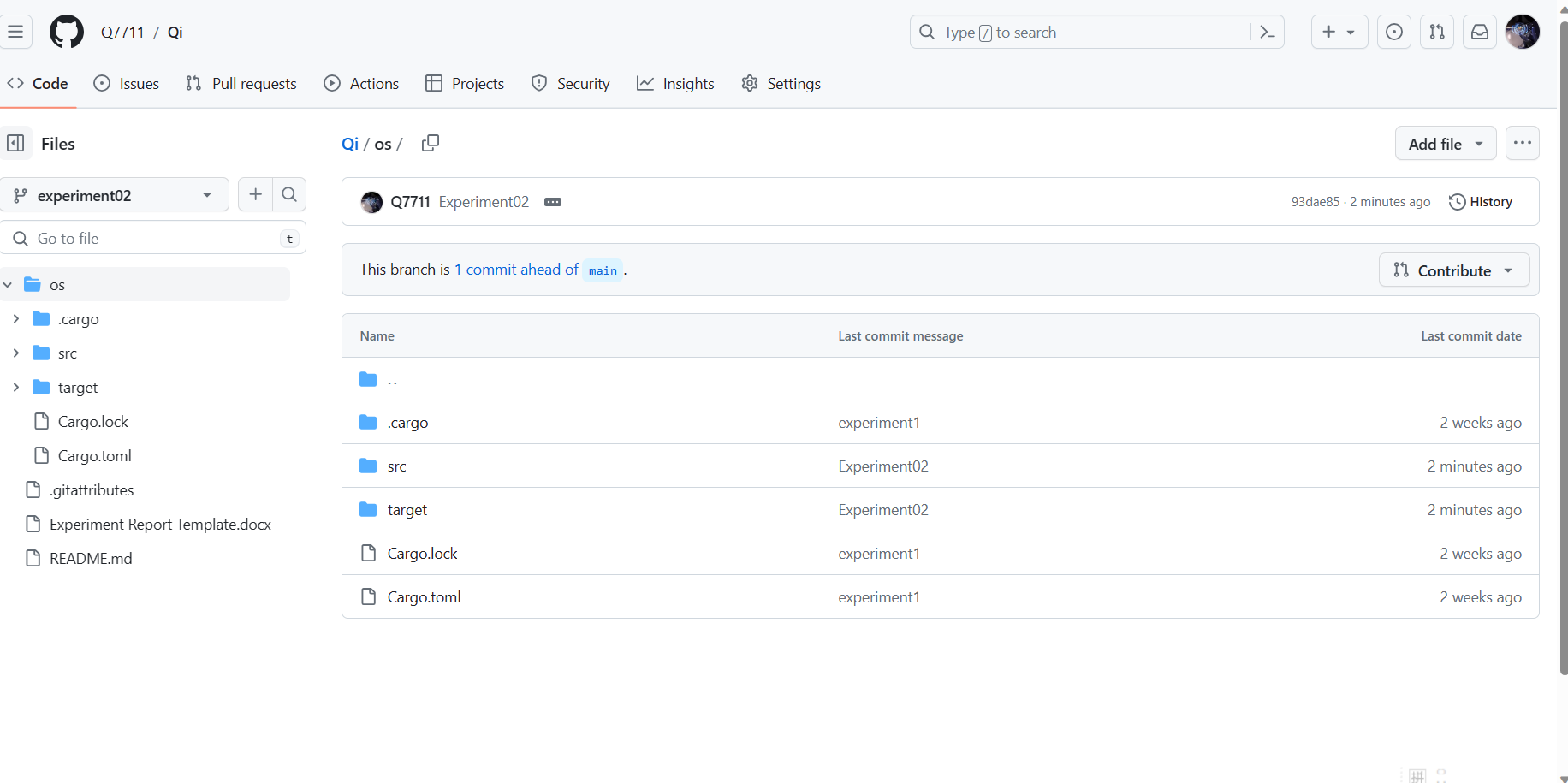
（1）分析linker.ld和entry.asm所完成的功能；

linker.ld指定输出文件的架构和程序的入口点,定义程序在内存中加载的起始地址基地址为 0x80200000，定义各个段的布局和属性，实现指定可执行文件的内存布局。entry.asm 文件设置了程序的入口点 \_start，初始化了栈空间，并调用 rust\_main 函数，设置正确的栈空间，为程序启动和初始化创建条件。

1. 分析sbi模块和lang\_items模块所完成的功能；

sbi模块定义了一系列SBI相关的常量和函数，以及内联函数sbi\_call。把之前的系统调用改成sbi调用，同时调用sbi提供的接口实现关机的功能。为实现在裸机上能够打印信息创造条件。lang\_items模块中，定义了panic函数和panic\_handler异常处理函数，给异常处理增加输出信息。

1. Git提交截图



1. 其他说明