Charles Bennington
4/6/2025

# Write Up 6

**Difficulties:** This assignment was tough, and I ended up writing a shell script specifically for testing specific programs. The shell script is below if you are curious. Overall, I felt that this was equivalently difficult to hw4. Mostly there were a lot of bugs. The most difficult part would be the nested for loops.

I found various bugs about my other programs while coding.

```
mvn compile assembly:single
FILE_NAME=$1
printf "\n\e[40;35m======================MYPL CODE=================================\e[0m\n"
cat -n $FILE_NAME
printf "\n\e[40;36m======================VM CODE===================================\e[0m\n"
./mypl -m IR $FILE_NAME
printf "\n\e[40;32m======================OUTPUT====================================\e[0m\n"
if ./mypl $FILE_NAME > /dev/null 2>&1; then
    ./mypl $FILE_NAME
else
    printf "\e[31m"
    ./mypl $FILE_NAME
    printf "\e[0m\n"
fi
```

**Prog 1:** Prog1 demonstrates a pseudo random number generator. Typically these pseudo random number generators use the “System time clock” as the seed value, resulting in a more random value. In mypl we don’t have that so the seed must be provided manually. Tests the math capabilities of mypl

**Prog 2:** Provides a visual representation of why pseudo random number generators aren’t actually random. The screenshot below shows the visual representation of the rand(), which can be seen to have a very predictable pattern. Tests structs and nested for loops.