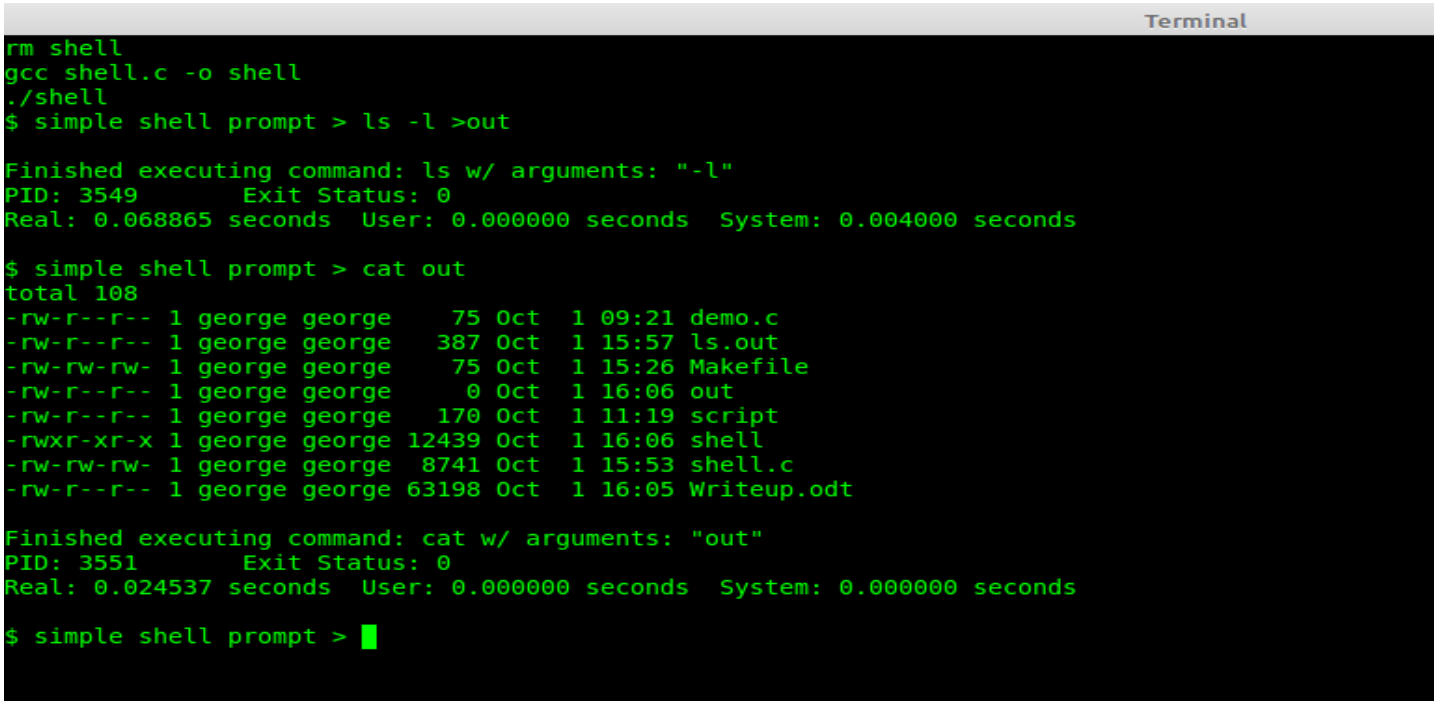
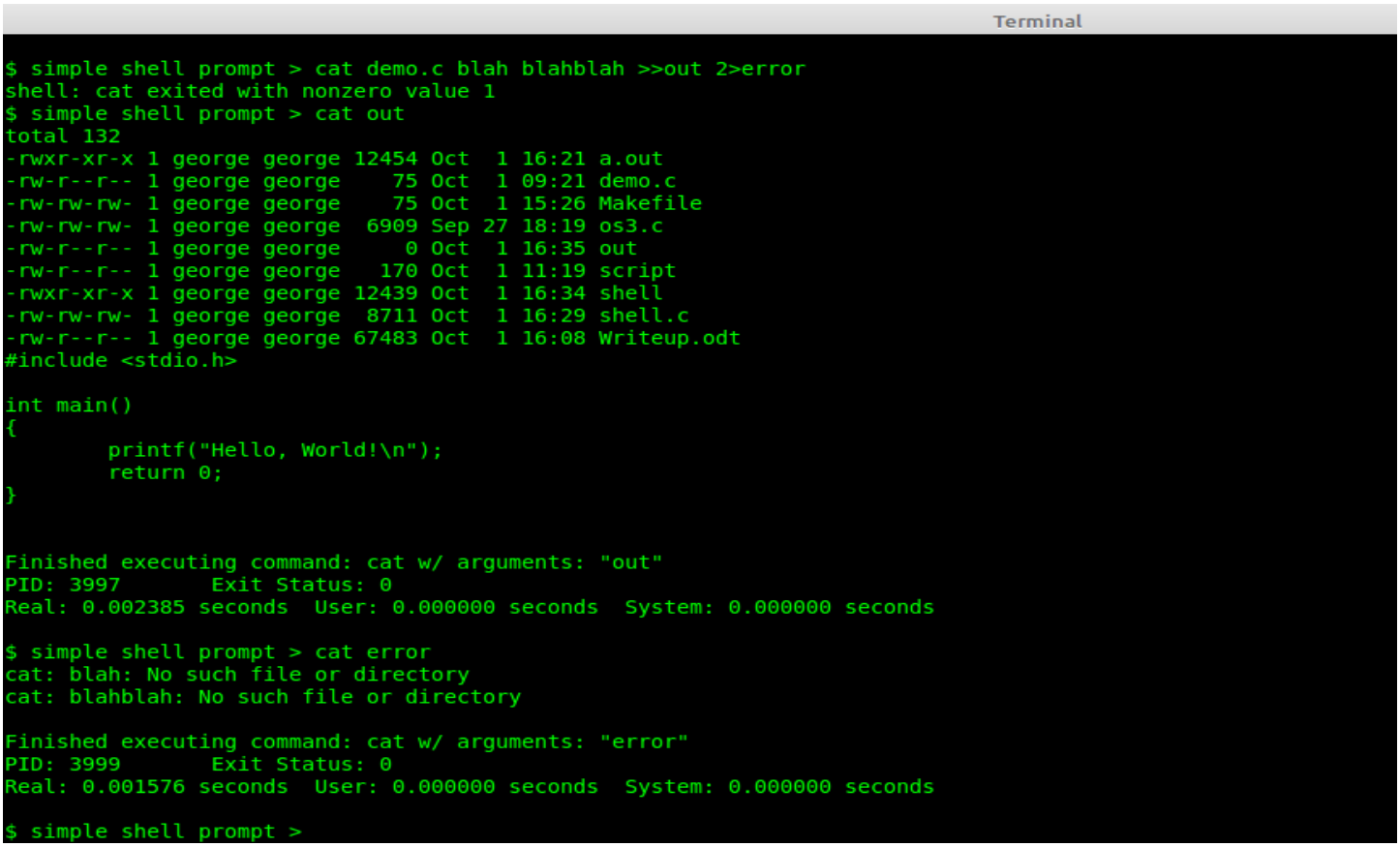
Stanley George
ECE 460: Computer Operating Systems
Professor Jeff Hakner

Problem Set 3: The Shell

Screenshot #1: shell ignores comment and runs command `ls` with argument "-l". The output is redirected to file `out`. The process stats are reported. Then, to verify that the output of `ls` really did go to `out`, run command `cat`. Again, process stats for `cat` are reported.



Screenshot #2: shell runs `cat` on three files: `demo.c`, `blah`, and `blahblah` and redirects output to `out` (created from above). Error messages are redirected to `error`. Then run `cat` to show contents of `out` and `error`



As shown in the screenshot, `demo.c` was a valid file and its contents were appended to the end of `out`. Since `blah` and `blahblah` are not existing files, error messages were generated by `cat` and sent to `error`.

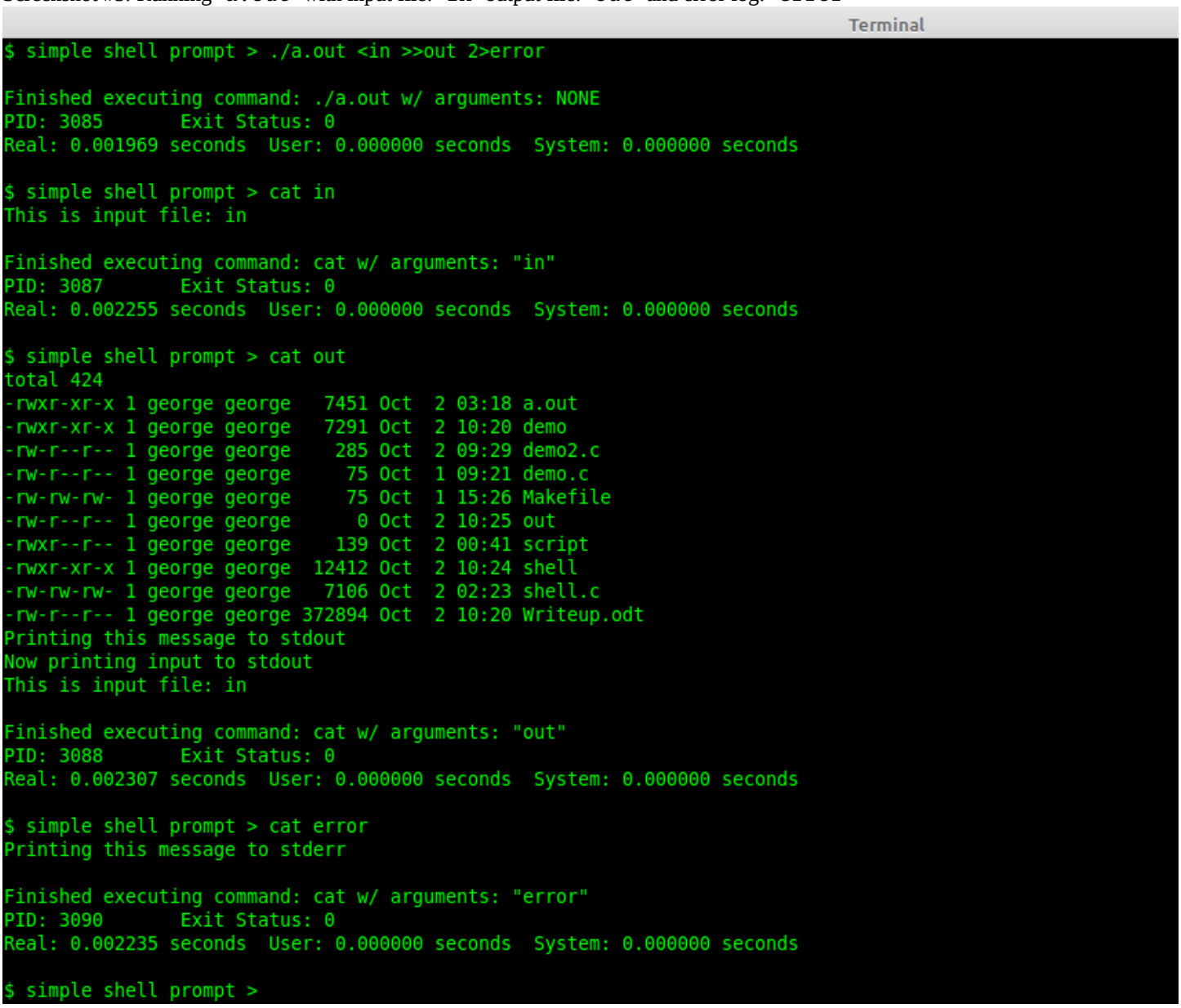
Screenshot #3: Running shell as an interpreter for script file `script` from within shell:

```
./shell
$ simple shell prompt > ./script
This is a script file called script...

Finished executing command: echo w/ arguments: "This"  "is"  "a"  "script"  "file"  "called"  "script..."
PID: 4806       Exit Status: 0
Real: 0.002012 seconds  User: 0.000000 seconds  System: 0.000000 seconds

#!/home/george/Desktop/ECE460/PS3/shell
echo This is a script file called script...
cat <script
gcc demo.c -o demo
./demo
rm demo

Finished executing command: cat w/ arguments: NONE
PID: 4807       Exit Status: 0
Real: 0.002060 seconds  User: 0.000000 seconds  System: 0.000000 seconds


Finished executing command: gcc w/ arguments: "demo.c"  "-o"  "demo"
PID: 4808       Exit Status: 0
Real: 0.074227 seconds  User: 0.056000 seconds  System: 0.008000 seconds

Hello, World!

Finished executing command: ./demo w/ arguments: NONE
PID: 4813       Exit Status: 0
Real: 0.000706 seconds  User: 0.000000 seconds  System: 0.000000 seconds


Finished executing command: rm w/ arguments: "demo"
PID: 4814       Exit Status: 0
Real: 0.000834 seconds  User: 0.000000 seconds  System: 0.000000 seconds



Finished executing command: ./script w/ arguments: NONE
PID: 4805       Exit Status: 0
Real: 0.082156 seconds  User: 0.056000 seconds  System: 0.008000 seconds

$ simple shell prompt >
```

Screenshot #4: Running shell as an interpreter for script file `script` from `bash` terminal

```
Terminal
george@george-Vostro-1400 ~/Desktop/ECE460/PS3 $ ./shell script
This is a script file called script...

Finished executing command: echo w/ arguments: "This"  "is"  "a"  "script"  "file"  "called"  "script..."
PID: 4862       Exit Status: 0
Real: 0.002000 seconds  User: 0.000000 seconds  System: 0.000000 seconds

#!/home/george/Desktop/ECE460/PS3/shell
echo This is a script file called script...
cat <script
gcc demo.c -o demo
./demo
rm demo

Finished executing command: cat w/ arguments: NONE
PID: 4863       Exit Status: 0
Real: 0.002118 seconds  User: 0.000000 seconds  System: 0.000000 seconds


Finished executing command: gcc w/ arguments: "demo.c"  "-o"  "demo"
PID: 4864       Exit Status: 0
Real: 0.071000 seconds  User: 0.048000 seconds  System: 0.012000 seconds

Hello, World!

Finished executing command: ./demo w/ arguments: NONE
PID: 4869       Exit Status: 0
Real: 0.000794 seconds  User: 0.000000 seconds  System: 0.000000 seconds


Finished executing command: rm w/ arguments: "demo"
PID: 4870       Exit Status: 0
Real: 0.000977 seconds  User: 0.000000 seconds  System: 0.000000 seconds


george@george-Vostro-1400 ~/Desktop/ECE460/PS3 $
```

Screenshot #5: Running `a.out` with input file: `in` output file: `out` and error log: `error`



The following is `demo2.c`, the source file used to compile `a.out` which was invoked in the above screen shot:

```
#include <stdio.h>

int main() {
   int c, i = 0;
   char buf[1024];
   while((c = getchar()) != '\n') {
      *(buf + i) = c;
      i++;
   }
   *(buf + i) = '\0';
   fprintf(stderr, "Printing this message to stderr\n");
   printf("Now printing our input to stdout:\n");
   printf("%s\n", buf);
   return 0;
}
```

This was the script used for the 2 previous screenshots:

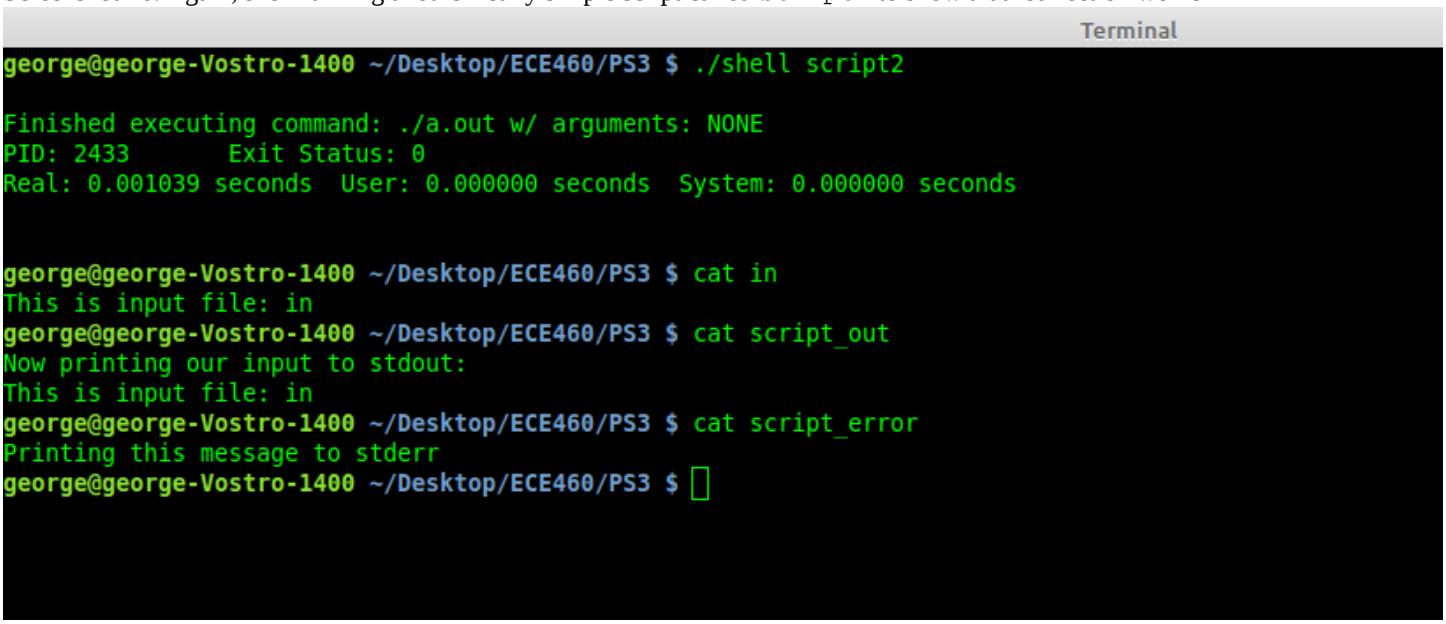
```
#!/home/george/Desktop/ECE460/PS3/shell
echo This is a script file called script...
cat <script
gcc demo.c -o demo
./demo
rm demo
```

This is `demo.c` that was compiled in the above script:

```
#include <stdio.h>

int main()
{
   printf("Hello, World!\n");
   return 0;
}
```

**IMPORTANT: I did have a bug in my program where if the last line in my script file was not `'\n'`, then the last command would not be executed.

Screenshot #6: Again, shell running another really simple script called `script2` to show that redirection works



The following is `script2`:

```
#!/home/george/Desktop/ECE460/PS3/shell
./a.out <in >script_out 2>script_error
```

`a.out` is the same as before: `demo2.c`

The following is the Makefile used to build shell:

```
shell: clean shell.c
   gcc shell.c -o shell
   ./shell
clean:
   clear
   rm shell
```