Assignment 1

Jason Zhao u7493671

1 The Bourne Again Shell (Bash)

1.1 Bash environment and basics

1. Bash is an interactive command interpreter and script language allowing users to interact with the operating system by commands.

2. Both .bashrc and .bash_profile are configuration files for Bash. .bashrc is executed every time when opening a new terminal window (interactive non-login shells) and commonly used to define aliases, functions, and shell customisations. .bash_profile is executed only when logging in to a system (login shells), commonly used to set environment variables like PATH or to run startup scripts.

3.
   ```
   echo $PATH
   ```

4. The PATH variable indicates where to look for the corresponding executable files when typing a command in the shell. It is convenient in the sense that without it the full path to every command would have to be specified every time.

5. File descriptors are process-unique identifiers, usually non-negative integers used to access files. FD0 normally refers to the standard input, the default channel Bash uses to receive input when typing something.

6.
   ```
   ls * wrong path or file > shell.out 2>&1
   ```

1.2 Bash scripts basics

1.
   ```
   #!/bin/bash
   echo $HOME
   echo $USER
   ```

2.
   ```
   nonsense=’This is such a nonsense!’
   echo $nonsense
   ```

3.
   ```
   echo $nonsense > nonsense.txt
   echo "nonsense.txt has been created/overwritten"
   ```

4.
   ```
   nonsense_=${nonsense// /_}
   echo $nonsense_ >> nonsense.txt
   ```

5.
   ```
   nonsense5=${nonsense/’such a’/’not’}
   nonsense5+=’ It’s Bash’
   echo $nonsense5 >> nonsense.txt
   ```

6. See script.

7.
```
echo $HOME | tr -d ’/’
```

1.3 More advanced Bash scripting

1.
```
#!/bin/bash
for i in {1..20}; do
  echo $i
done
echo
```

2.
```
for i in {1..20}; do
  echo -n "$i "
done
echo
```

3.
```
for i in {1..20}; do
  printf "%02d, " "$i"
done
echo
```

4.
```
for i in {1..20}; do
  if [ "$i" -eq 20 ]; then
    printf "%02d" "$i"
  else
    printf "%02d, " "$i"
  fi
done
echo
```

5.
```
for item in /*; do
  echo "Listing item in root: $item"
done
```

2 Useful shell commands

1.

2.
```
wget http://www.mso.anu.edu.au/˜chfeder/teaching/astr_4004_8004
/material/mM4_10048_pdfs/EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data
```

3.
```
grep ’sigma’ EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data
```

4.
```
grep -A 1 ’sigma’ EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data
```

5.
```
grep -A 1 ’sigma’ EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data | grep -v ’sigma’
```

6.
```
bash --help
```

7.
   ```
   bash --help | grep -A 1 ’^Usage’
   ```

8.
   ```
   bash --help | grep -A 1 ’^Usage’ | wc -w
   ```

9.
   ```
   echo "Number of words: $(bash --help | grep -A 1 ’^Usage’ | wc -w)"
   ```

10.
    ```
    awk ’NR <= 11’ EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data > header.txt
    ```

11.
    ```
    awk ’NR > 11 {print $1}’ EXTREME_hdf5_plt_cnt_0050_dens.pdf_ln_data > column1.txt
    ```