CS 2750 **System Programming and Tools** Test 2
Name: Spring 2018 Max Pts: 58

Important: This is an open book test. You can use any books, notes, or paper, but not exchange anything with other students. You are not allowed to use any electronic/communication device, including a calculator and e-books. *Do not log into the computer during the test. Change your cell phones to silent mode. Any other device with an* ON-OFF *switch should have its switch in the* OFF *position.* Any calculations and rough work can be done on the backside of the test pages. If there is a syntax error in any program segment, just point it out and you will get full credit for the problem. Please write legibly; if I cannot read what you wrote, I'll give you a zero. You will lose five points for not writing your name.

1. [6 pt] Look at the following two commands in `bash` and their output [the commands are preceded by the prompt `$`]:

```
$ echo { 1, 2, 3}
1 2 3
$ echo [[:digit:]]
[[:digit:]]
```

The output of the first command is what was expected but I thought the second command should print all digits from `0` to `9`. Why don't I observe what I expected?

`[[:digit:]]` is a character group used to match any digit. It does not provide a sequence of characters. The `echo` statement will attempt to match any filenames in the current directory that may be of single digit, lacking which, it simply prints the pattern itself.

2. [10 pt] Write the awk command to print the number of words in each line of a file named `textfile`, followed by a space, followed by the first 25 characters of the line. At the end, you should also print the number of words in the entire file. Make sure that the number of words is printed in two characters, for example, 7 will be printed as a space followed by the number 7.

```
awk ’BEGIN { nwords = 0 }
           { nwords += NF; printf("%02d %s\n", NF, substr( $0, 0, 25 )) }
     END   { printf ( "Total words: %d\n", nwords ) }’ textfile
```

3. [6 pt] What would be printed by the execution of following code:

```
#include <stdio.h>

int main()
{
    printf ( "What’s \new?\n" );

    return ( 0 );
}
```

```
What’s
ew?
```

4. [6 pt] What is the output produced by executing the following script?

```
#!/bin/bash

x=( a b )
y=( 1 2 3 )

for (( i=0; i<${#x[@]}; i++ ))
do
    for (( j=0; j<${#y[@]}; j++ ))
    do
        echo ${x[${i}]}"--"${y[${j}]}
    done
done
```

```
a--1
a--2
a--3
b--1
b--2
b--3
```

5. [6 pt] Why is it recommended that you should not use the `return` statement to return a value from a function in bash?

   The use of `return` limits the output integer range from 0 to 255. Also, the output may not be assigned to another variable but has to be extracted from `$?`.

6. [6 pt] What is a derived file when you compile some code? Give at least two examples of different types of derived files.

   A derived file is one that is created as a result of compilation or linking. Two examples of derived file are an object file and an executable.

7. [6 pt] What is the output produced by the following code? Assume that the user types `George Herbert Walker Bush` when prompted.

```
#include <stdio.h>

int main()
{
   char fn[30], ln[30];

   printf ( "Please enter your name:" );
   printf ( "\n%d\n", scanf ( "%s%s", fn, ln) );

   return ( 0 );
}
```

   2

8. [6 pt] What is printed by the following code?

```
#include<stdio.h>

void swap (char * str1, char * str2 )
{
  char * tmp = str1;
  str1 = str2;
  str2 = tmp;
}

int main()
{
  char * str1 = "John";
  char * str2 = "Doe";
  swap ( str1, str2 );
  printf ( "String1: %s\nString2: %s\n", str1, str2 );

  return ( 0 );
}
```

```
String1: John
String2: Doe
```

9. [6 pt] In the following program, what will you put in place of `?` so that the code prints `World`?

```
#include <stdio.h>

int main()
{
    char arr[] = "Hello World";
    printf( "%s", ? );
    return 0;
}
```

```
arr + 6
```