CSE 333 - OPERATING SYSTEMS
Programming Assignment # 1
DUE DATE: 05/11/2019 - 23:59

1. (20 pts) A palindrome is a word, a phrase, or sequence that reads the same backwards as forwards. Write a shell script that will take one command line argument as string and decide if it is a palindrome. For example:

   ```
   Ex:
   $ ./myprog1.sh "apple"
   apple is not a palindrome

   Ex:
   $ ./myprog1.sh "ey edip adanada pide ye"
    ey edip adanada pide ye is a palindrome
   ```

   Your program should work on both lowercase and uppercase letters.

   ```
   Ex:
   $ ./myprog1.sh "Madam"
      Madam is a palindrome
   ```

2. (15 pts) Write a shell script that takes an optional pathname as parameter. If your program is run with no parameters, it will create a directory named **cprogs** under current working directory and move all of the files whose name ends with .c to this directory. If your program is run with an argument, it will create a directory names **cprogs** under given path and move all of the files whose name ends with .c under given path to this directory. For example:

   ```
   Ex:
   $ ./myprog2.sh
   A  directory  named  cprogs  is  created  under  current  working
   directory.
   All  the  files  whose  name  ends  with  .c  in  the  current  working
   directory are moved into cprogs directory.

   Ex:
   $ ./myprog2.sh /home/zuhal/Desktop
   A directory named cprogs is created under /home/zuhal/Desktop.
   All the files whose name ends with .c in /home/zuhal/Desktop are
   moved into cprogs directory.
   ```

3. (15 pts) Write a shell script that will take two integer arguments. The first argument must be greater than the second one. The difference between the two arguments must be an even number. Then your program will draw a hollowed square with the first argument being the side of outer square and the second one being the side of the inner square. For example:

   ```
   Ex:
   $ ./myprogr3.sh 7 5
   *******
   *     *
   *     *
   *     *
   *     *
   *     *
   *******
   ```

```
Ex:
$ ./myprogr3.sh 8 2
*******
*******
*******
***  ***
***  ***
*******
*******
*******
```

4. (20 pts) Write a shell script that takes a wildcard and an optional pathname as an argument. If your program is run only with the wildcard as an argument, then it will search for the words that matches with this wildcard inside all the files whose name ends with .txt under current working directory and substitute them with their uppercase versions. If your program is run with two arguments, one being the wildcard and the other being the pathname, then your program search for the words that matches with this wildcard inside all the files whose name ends with .txt under given pathname and substitute them with their uppercase versions. Note that all of the occurrences of the same word will also be changed. For example:

```
Ex:
$ ./myprog4.sh "ek*"
 The word "ekin" inside test.txt is substituted with "EKIN".
 The word "ekmek" inside test.txt is substituted with "EKMEK".
 The word "ekleme" inside deneme.txt is substituted with "EKLEME".

Ex:
$ ./myprog4.sh "an*e" /home/zuhal/Desktop
  The word "anne" inside /home/zuhal/Desktop/aile.txt is substituted
with "ANNE".
  The word "anne" inside /home/zuhal/Desktop/aile.txt is substituted
with "ANNE".
 The word "antre" inside ev.txt is substituted with "ANTRE".
```

5. (20 pts) Write a shell script that takes an optional pathname as an argument and a -R option. If your program is run with no arguments, then it will find all the files whose size is zero under current working directory and ask the user to delete them one by one. If your program is run with a pathname as an argument, then it will find all the files whose size is zero under given pathname and ask the user to delete them one by one. If -R option is given, then your program will work recursively. For example:

```
Ex:
$ ls -l
-r---w---- 1 std std 13107 Jun 20 2005 cask-of-amontillado.txt
-rw------- 1 std std 0 Jun 20 2005 french.txt
drwx------ 14 std std 456 May 25 2007 shakespeare
-rw------- 1 std std 0 Jun 20 2005 trees-and-other-poems.txt

$ ls -l shakespeare
-rw------- 1 std std 456 Jun 20 2005 barleby-scrivener.txt
-rw------- 1 std std 0 Jun 20 2005 calaveras-county.txt

$ myprogr5.sh -R
Do you want to delete french.txt? (y/n): y
1 file deleted

Do you want to delete trees-and-other-poems.txt? (y/n): n

Do you want to delete calaveras-county.txt? (y/n): y
```

```
        1 file deleted

        Ex:
        $ ls -l /home/zuhal/Desktop
        -rwx------ 1 std std 0 Jun 20 2005 alice-in-wonderland.txt
        -rw------- 1 std std 496769 Jun 20 2005 hawthorne.txt
        -rw------- 1 std std 172541 Jun 20 2005 looking-glass.txt
        drwx------ 14 std std 0 May 25 2007 stories
        -r-------- 1 std std 0 Jun 20 2005 song-of-hiawatha.txt

        $ myprogr5.sh /home/zuhal/Desktop
        Do you want to delete alice-in-wonderland.txt? (y/n): y
        1 file deleted

        Do you want to delete song-of-hiawatha.txt? (y/n): y
        1 file deleted
```

Note that the directories are not deleted. Note also that you can use option and argument at the same time.

- **Bonus:** You will get 10% extra credit if your program supports a *Menu* including all questions above. Example:

```
  $ ./myprog.sh

Please select an option:

    1. Check for palindromes
    2. Move .c files
    3. Draw hollowed square
    4. Uppercase conversion
    5. Delete files
    6. Exit
```

These options must be printed inside a loop until "Exit" option is selected. When user enters one choice, you should ask for the arguments if that option requires any.

Notes:

- You are required to consider all necessary error checking for the programs.

- No late homework will be accepted.

- In case of any form of **copying and cheating** on solutions, all parties will get **ZERO** grade. You should submit your own work. In case of any forms of cheating or copying, both giver and receiver are equally culpable and suffer equal penalties.

- You have to work with a partner. Your partner will not be changed throughout the semester.

- (10 pts) You are required to submit a minimum 2-pages report and commented code (via e-mail: cse333.projects@gmail.com).

- Your report should include explanations about implementations with screenshots of your sample executions. Your implementation detail should be provided in the source code comment.

- Please put your COMMENTED source codes and project report in a zip file and make sure that your zip file name contains your student IDs! Ex: 150713852_150713853_Project1.zip

- If after 3 projects, your overall grade is higher than 100, it will be considered as 100.