

COMPUTER SCIENCE 50100  
Computing for Science and Engineering  
FALL 2015  
ASSIGNMENT # 1 (35 points)  
August 24

**Due Wednesday, September 16 at 10:30 am** This assignment covers through Section 2.2 of the class notes.

The use of `continue` is not permitted.

Every loop is permitted to have only one exit. In particular, a `break` statement is permitted as an exit only for a loop with header “`while True:`”.

Do not use nested `ifs` when short circuit evaluation works.

1. (2 points) Let `process` be a program that runs from the bash command line. Suppose that `process` reads from the keyboard and writes output and error messages to the terminal window. By means of a bash command, show how can we use it to read from a file `in.dat`, write output to `out.dat`, and send error messages to the “bit bucket”.
2. (3 points) Write a bash script `envi.sh` that takes a single argument  $\langle string \rangle$  and prints each environment variable and its value for which at least one of them contains the string of characters  $\langle string \rangle$ . For example

```
$ ./envi.sh TERM
TERM_PROGRAM=Apple_Terminal
TERM=xterm-256color
TERM_PROGRAM_VERSION=343.7
TERM_SESSION_ID=99F60DEC-36AC-46B2-9FBD-03FA60530D04
```

You may assume that  $\langle string \rangle$  contains only alphanumeric characters and underscores.

3. (7 points) Write a script that, given a string as an argument, prints a count of the number of occurrences of each alphabetic character *present* in the string, without distinguishing between lower or upper case. Useful functions: `str.isalpha`, `str.lower`. For example,

```
$ alphacount.py 'One, two, 3!'
o: 2
w: 1
e: 1
t: 1
n: 1
```

The ordering of the output can be arbitrary.

4. (10 points) Write a script `addition.py` that tests the user’s knowledge of the addition table, as illustrated below:

```

$ addition.py
6 + 3 = ?
12
Incorrect; try again
9
Correct.
8 + 2 = ?
10
Correct.
1 + 8 = ?
Good bye.

```

Each of the operands is a random integer from 1 through 9. The user is given an unlimited number of chances to enter the correct answer. The user terminates the interaction by entering control-D.

5. (13 points) Write a script `cube2sphere.py` that extracts data from the file `input.dat` given on Piazza to produce the file `output.dat` given in Appendix A. The script constructs two tables, each table has columns corresponding to the methods `cubicBspline Taylor4,...`, `undecicBspline Taylor12` and rows for the cutoffs 7, 8, ..., 20. Each entry of the first table compares a CUBE grid result with the corresponding SPHERE grid result by giving

$$((t_{\text{calc for CUBE}}) - (t_{\text{calc for SPHERE}})) / (t_{\text{calc for SPHERE}})$$

in percentage rounded to the nearest integer. The second table has entries giving

$$((t_{\text{setup for CUBE}}) - (t_{\text{setup for SPHERE}})) / (t_{\text{calc for SPHERE}})$$

Note that the denominators are same in both cases. You may fully exploit the specific format of the file, except you may not embed any numerical values from the file into your program (other than 30006).

**Testing** Your code will be tested by testing programs. The plan is to make prototypes available on Piazza. Additional testing may be added to the testers though an effort is made to have the prototype catch all types of errors.

**Send your solutions electronically** You should submit one PDF file containing your answer to problem 1, as well as your source code for problems 2–5. Additionally, send your source code for problems 2–5 as separate files. PDF files can be created by scanning hard copy or by printing-to-PDF from a word processor or text editor or by saving-as or by using `pdflatex`.

1. Create a folder/directory containing all files and name the folder `hw1_XXXXX`, where `XXXXX` represents your Purdue career account name (e.g., `hw1_yfang`). Compress the folder to a .zip file and name it `hw1_XXXXX.zip`. To pack a set of files and directories into a single file, use

```
zip <name>.zip <list>
```

and to unpack

```
unzip <name>.zip
```

Note: you are required to follow this instruction to name your files, or else points may be deducted.

2. Log on to Blackboard at <http://www.itap.purdue.edu/tlt/blackboard/> (To use Blackboard, you have to have Java enabled for your web browser.)
3. Click on Course Content from the left panel. You will find the homeworks.
4. Go into a homework assignment, you will see a submission window on the page. Click the "Browse My Computer" Button. You will be able to submit the .zip file containing your homework answers there.
5. Submission will be closed after the due time. Please manage your time carefully.

**A: Output file output.dat**

```

-21  -21   1   0   0
-21  -21   1   0   3
-21  -21   1   0   3
-21  -20   1   0   3
-23  -20   4   0   1
-21  -12   1   0   0
-21   4    1   0   0
-21   4    1   0   0
-21   4    0   0   0
-21   4    1   0   0
-21   4    1   0   0
-21   4    1   0   0
-21   4    0   0  -1
-21   1    0   0  -1

  0   -5    1  16   0
  1   -3    2  -6   2
  2   -1    2  -3  12
  1   -2    2  -4  11
  2    0    3   8   9
  3    1    4  -2   3
  5    8    5   3   3
  8   11    8   3   5
  5    9    6   6   4
  8   13    9   9   5
 11   18   13  11   8
 15   24   18  13  10
 21   32   23  19  14
 15   27   19  16  11

```