CSE/ISE 337 Assignment 2 (Spring 2014)

Due date: Tuesday, Mar .18, at 11:55pm

Important! Must read: (a) When writing programs, you **must** use the techniques that are described in the lecture notes and illustrated in the given samples. You may **not** use methods, modules, packages that were not covered in lectures. (b) You **must** first read the lecture slides "0-Course-Overview.pdf" available in Blackboard – Documents – Lecture Slides, especially those related to assignments on Slides 0-9 to 0-13, and follow them. (c) Start working on this assignment right away; you will **not** be able to finish it if you wait until the last day. (d) You must have "`use strict;`" and "`use warnings;`" at the start of your programs.

1. (5pts) Write a program that repeats a given string n times. Both the string and the number n are read from the standard input, i.e., from the user input, in that order. Preserve the newline character at the end of the given string.

2. (5pts) (a) Use the `for` loop to print a table of 32 rows and 3 columns. Each row contains the row number `i`, `2*i`, and `i*i` respectively. The row numbers start from 1, up to and include 32. Format the output so that each column takes up a 5-character width. (b) Repeat part (a), but this time use a `foreach` loop instead.

3. (10pts) Write a program that reads a number n from the standard input. It then reads n lines of text from user input, and prints these lines backwards, i.e., if n=5, it prints the 5th line first, the 4th line second, …, and the 1st line last. Your program must use the `do … until` loop at least once.

4. (10pts) Subroutine; scalar vs. list context. Define a subroutine that takes the radius of a circle as input, it returns the area of the circle if it is called in a scalar context; or a list that contains the area, the diameter and the circumference if called in a list context. Demonstrate the usage of this subroutine with a small calling program. **Hints**: (a) you may define a constant pi with value 3.141593. An example showing how to define a constant in Perl is "`use constant PI => 3.141593; print "PI is ", PI, "\n";`" (b) use the `wantarray()` function

5. (10pts) Write a program that reads the user input of words until ctrl-D is typed. There can be a single word or multiple words per input line. Use hash to count the number of occurrences of each word in the input. Print the content of the hash after the hash is built, with one word and its count per line. Separate the input from the output by a line that is made of 20 dashes.

6. (10pts) Write a program that takes the name of an input file as a command line argument. The input file contains some source code with the following format:

```
1 print( "Enter a number: " );
2 $number = <>;
3 chomp( $number );
4
5 print "$number is an even number\n" if $number %2==0;
```

   Your program removes all line numbers (and the space after each line number) and preserves the source code format including empty lines.

7. (10pts) Write a grouping program. It takes the name of a name-list file and a positive integer N from command line arguments. Each line of the name-list file contains a person's name.

The program goes through the input file from beginning to end, and forms groups of N members. For each group, it prints the group number, then the N names on N lines respectively. Note that the last group may contain <N members. To run the program, you may type "`perl q1.pl namelist 5`", in which q1.pl is the program that you write, namelist is the name-list file name, and 5 is the group size.

8. (Bonus 8pts) Build a list of three hashes. The first hash stores three first-year courses, the second hash stores three second-year courses, and the third hash stores three third-year courses. For each course, the key would be the course number, such as "110", and the value would be its course title, for example "Introduction to Computer Science". Choose any courses that you like. Write a subroutine that does the following: it takes three arguments, the first is a reference to the list described above, the second is a number indicating the year, 1 for the first-year, 2 for the second year, and so on. The third is a course number, e.g., 110. It then checks the list of hashes if it contains the given course in the given year, if so it returns the course title, if not an alert message "Unknown course!" will be returned. Finally call the subroutine a few times to verify its correctness.

Deliverables

Your assignment submission should include two files: (a) a printout of all programs that you write. Concatenate them into one **plaintext** file called "a2-printout.txt". Each program should be clearly labeled with its corresponding question/part numbers. (b) A **zip** file that includes all individual programs that you write. Name it "a2-source.zip". Be sure to name each program using its question and part number, e.g., "q2part_a", "q2part_b", and so on. You should include certain amount of program documentation, i.e., in-line comments, in your programs for important steps used. Do not repeat what the line of code says; rather write comments to help readers to understand your code.

Total: 60 points

Submission instructions

The handing-in will be through Blackboard Assignment. The submission instructions are at: http://it.stonybrook.edu/help/kb/creating-and-managing-assignments-in-blackboard.
You **must** read the submission instructions very carefully, and check to make sure your assignment has been submitted correctly **before** the deadline.

You can only submit once! However you can save your work by clicking "Save" as many times as you like. Only click "Submit" after you have checked and are certain that all requirements are followed.

Late submissions will not be accepted. The due date is **11:55pm on Tuesday, March 18.**