

C Sci 40 Assignments

1: Volume.cpp) Write a program that reads a radius and prints the volume and surface area of the sphere with that radius. ($V = 4/3 \pi r^3$; $S = 4 \pi r^2$; use $\pi = 3.14159$)

2: commas.cpp) Write a program that accepts a 7-9 digit integer and echoes the number with commas between every three digits from the right. Test it with the numbers 1234567, 987654321, 20300045, & 10000000.

3: Mississippi.cpp) Print the word "Mississippi" in block letters (e.g., drawn entirely with asterisks). Each letter should be printed by its own function.

4: Format.cpp) Write a program that gets two real values from the user, prints their sum and product in scientific notation, and prints their sum and product in fixed point notation. Print answers to three decimal places.

5: CheckLetter.cpp) Write a program that adds a check letter to an eight-digit number. The check letter should be computed as follows: Break the number up into 4 two-digit numbers. Add the four numbers together. Find the remainder after division by 26. The check letter is the letter in the alphabet that corresponds to the number just computed. (A=0, B=1, C=2, etc.) Print the original number followed by the check letter. Use at least one function that returns a value when you write this program.

6: forLoops.cpp)

a) Print the even numbers between 3 and 43.

b) Print every third letter starting with 'B'.

c) Print the numbers 1 to 20 in one column right next to another column with the numbers 20 down to 1.

d) Print every other character in the computer's character set.

Write this assignment as one program with 4 void functions.

7: Exams.cpp) Suppose a teacher weights the four exams he gives 10%, 25%, 30%, and 35%. Write a program that reads ten sets of four grades, prints the weighted average of each set, and prints the unweighted average of each test. The number of students should be in a global constant.

8: PGandE.cpp) Consider a gas and electric bill that sets two levels of use as follows:

	Gas	Electric
Tier 1 quantities	10.1 therms	558 kwh
Tier 1 price	\$1.415 /therm	\$.223 /kwh
Tier 2 Price	\$1.932 /therm	\$.281 /kwh

Write a program that takes starting and ending readings for gas and electric meters and calculates charges to the nearest cent. Use one function to calculate charges for both gas and electric so as to eliminate duplicate calculations.

9: perfect.cpp) A number is called perfect if the sum of its divisors is equal to the original number. A number is called deficient if the sum of its divisors is less than the original number. A number is called abundant if the sum of its divisors is more than the original number. Write a program that answers the following questions: What numbers below 5000 are perfect? What odd numbers below 5000 are abundant? What are the relative proportions of deficient, abundant, and perfect numbers? 5000 should be in a global constant.

10: numberWords.cpp) Write a program that reads a whole number of up to nine digits and prints it in words. For example, the input 13247 ought to produce "thirteen thousand two hundred forty seven".

11: series.cpp) Write a program that will add the terms of an infinite geometric series. The program should read the first term (a) and the common ratio (r) for the series. It should then compute the sum in two ways: by formula (

$$S = \frac{a}{1-r} \text{)}, \text{ and by adding the individual terms until the}$$

answer agrees with the formula to 7 significant digits. Then print the formula answer, the answer found by adding the

terms, and the number of terms that were added. Print the sums to at least ten places. Verify input with a while loop. Two real values are equal to n significant digits if the following condition is true:

$$|a-b| < |a| * 10^{-n}.$$

12: guess.cpp) Write a program that plays a guessing game with the user. The user should pick a letter, and the computer should try to guess the letter. After each guess, the user should tell the computer whether its guess was too high or too low. With this information, the computer should be able to guess the letter within five tries. The user should be able to give whole words or single upper or lower case letters as responses.

13: bigNumber.cpp) Write functions for doing arithmetic on 100-digit integers. Include functions for input, output, and addition. The input function should not print anything. The addition function should do no reading or printing. No function should make any assumptions about how the other functions or main program work. The number of digits should be in a global constant. The function prototypes should be as follows:

```
void readBig(int[])
void addBig(int[], int[], int[])
void printBig(int[])
```

Use the main program provided to test the functions.

14: files.cpp) Write a program that reads a file encoded with the "rot13" cypher, decodes it, and stores the decoded file. The decoded file name should be the original file name with the word "coded" attached.

15: points.cpp) Write a program that reads 10 pairs of Cartesian coordinates from a file called "points.txt" and sorts them by increasing x-values, decreasing y-values, and increasing distance from the origin. Use only one sorting routine. Use an enumerated data type to keep track of the field on which the list is being sorted. The number of pairs should be in a global constant.

16: linkedList.cpp) Write a program in three parts. The first part should create a linked list of 26 nodes. The second part should fill the list with the letters of the alphabet. The third part should print the contents of the linked list.

17: recursion.cpp) Write two recursive integer functions. The first function should calculate factorials using the definition $0! = 1$ and $n! = (n-1)! \times n$. In addition, if n is too large it should be able to detect integer overflow before it happens. The next function should calculate the greatest common factor (gcf) of two numbers. If the second number is zero, then the gcf is the other number. Otherwise, the gcf of **a** and **b** is the same as the gcf of **b** and **a mod b**.

18: bigNumber.h, bigNumber.cpp, bigNumberTest.cpp) Rewrite problem 13 as a BigNumber object with readBig, writeBig, and addBig member functions. The function prototypes from the header file should be:

```
void readBig();
void printBig();
bigNumber addBig(bigNumber);
```

The integer array should be a private data member. Submit the program in three files.

There will be written assignments in addition.