

Assignment #4 – structs/classes

Points: 30

Instructions (3 points)

1. Write a comment at the top that contains the following information

```
// Your Name  
// CS 52  
// Assignment #4
```

2. Properly indent, format and comment your code as necessary

Warning: - No late submission is accepted!
- Points will be deducted for entirely commented files!
- Name your files *problem{#}.cpp*, for example *problem1.cpp*, *problem2.cpp*, etc.
- Canvas may append -1, such as *problem2-1.cpp* which is okay!

Problem 1

Recursion

7 points

Implement a program that lets the user enter a number and then prints the result calculated by the following function. Write a recursive function that $n * n/3 * n/9 * \dots$ where $n/x > 0$ and n is a nonnegative integer. For example:

```
27: 27 * 9 * 3 * 1 = 729  
8: 8 * 2 = 16  
100: 100 * 33 * 11 * 3 * 1 = 108900
```

The function prototype should be: `int recursive(unsigned int n);`

Example:

```
Input: 27  
Result: 729
```

Problem 2

Date Class

10 points

Design a class called Date. The class should store a date in three integers month, day, and year. Declare all member variables private and all member functions public. Each instance variable should have getter and setter functions to get and set each value. There should be three member functions to print the date in the following forms:

```
12/25/2018      December 25, 2018      25. December 2018
```

Implement a program that lets the user enter a date using your Date class. The user enters a day, month and year. You will then create an object of the Date class passing in the three values to the constructor. Call each function to print the three different date forms as shown above.

Problem 3

structs

10 points

Create a *struct* called Employee. The *struct* should have the following members:

- firstName: **string**
- lastName: **string**
- pay: **int**

Create a dynamic array of n employee variables where n is a number entered by the user. Then loop through the array and let the user input the data for each employee. Create a function that expects a pointer to the array as input and prints the elements in the array. Remember that arrays are *passed by reference*, that is, any modifications inside the function will be reflected in the caller's array.

Example: Number of employees: **2**
 Employee 1
 First Name: **Allen**
 Last Name: **Harper**
 Pay: **1000**
 Employee 2
 First Name: **John**
 Last Name: **Doe**
 Pay: **500**

 Employees:
 Name: Allen Harper (1000)
 Name: John Doe (500)