

Given: Fri Sep 30

Due: Thu Oct 6, 1 p.m.

1. (48%) Revise the class `Song` you created in the previous assignment by doing the following.
  - (a) Add a method `set(title, artist)` that sets the title and artist of the song to the given values. Both arguments are strings.
  - (b) Replace the `print` method by an output operator (`<<`).
  - (c) Replace the `read` method by an input operator (`>>`).
  - (d) Replace the `is_equal` method by an equality operator (`==`).

Write a test driver for the new method and operators. Organize your code into separate files as explained in Section 2.9 of the notes.

2. (12%) Add to the class `Time` an assignment operator that allows a time to be set to a given hour. For example, `t = 14` should cause `t` to be set to 14:00. The operator should not perform unnecessary conversions of integers to times. Add the operator to the version of `Time` included in Version 2.3 of the pay calculator (available on the course web site as `PayCalculator2.3`). Write a test driver for the new operator. Keep the code organized into separate files.
3. (20%) Create a function `print_double_spaced(cs)` that takes a C string as argument and prints the characters of that string separated by a space. For example, the string `hello world` should be printed as

h e l l o   w o r l d

Note that a total of three spaces are printed between the words `hello` and `word` in this example. Note also that no spaces are printed before the first character of the string and after the last one. Implement the function as efficiently as possible.

4. (20%) Create a function `count_lines(file_name, s)` that returns the number of lines in the file called `file_name` that contain the string `s`. Both arguments are C++ strings. Use appropriate string operations to keep your implementation as simple as possible.