

CS23021 Computer Science I

Project #4

Due by 11:59pm on Thursday, April 17, 2008

You will *write two programs* for this project. The first program “removedups” eliminates double entries in an array of integers. The second program “checker” corrects simple syntax errors in English text.

1. Removedups. Write a program that accepts a sequence of integers (some of which may repeat) as input into an array. Write a function that processes the array so that any duplicate values are eliminated. Write an output function that prints out the values of the array and the sum of the values. You can assume that there are no more than 20 integers in the input. But there may be less. Zero signifies the end of input. The zero should NOT be printed.

For example, when the following input is provided to your program:

```
5 6 22 5 22 7 6 0
```

your program should print:

```
The sum of 5 6 22 7 is 40
```

- You must use an array to store the numbers that are input.
- Develop your program with the functions in a separate file called `dupfuncs.cpp` and the main in a file called: `removedups.cpp`
- EXTRA-CREDIT OPTION: for 10% extra credit you may prompt the user for a file name from which to receive the integer input, but your program must also offer the interactive input option as specified in the original description.
-

2. Checker. Write a program that reads in English text. The program should output the text with spacing corrected and with letters corrected for capitalization. In particular, in the output text all strings of two or more blank spaces should be compressed to a single blank. Each sentence (a letter starting the text and following a dperiod) should start with an uppercase letter. There should be a space after a comma or a period. No other formatting is required.

For example, the input:

```
the Answer to life,the Universe, and everything is 42 . that is all .  
Do , or do not . there IS No 'try' .
```

Should produce the following output:

```
The answer to life, the universe, and everything is 42. That is all.  
Do, or do not. There is no 'try'.
```

- An empty line signifies the end of input. This empty line should NOT be output.
- You must use a C++ string to store a line of input.
- Develop your program using appropriate functional decomposition, place the functions in a file called `checkerfuncs.cpp` and the main in a file called: `checker.cpp`
- EXTRA-CREDIT OPTION: for 10% extra credit you may prompt the user for a file name from which to read in the sentences (lines), but your program must also offer the interactive input option as specified in the original description.