Name: (as it would appear on official course roster)	
UCSB email address:	**@ucsb.edu**
Lab Section Time:	
Optional: name you wish to be called if different from above	
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone")	

Homework 05: Strings

Assigned: *Tuesday, November 3rd, 2020*
Due: *Monday, November 9th, 2020 by 11:59 PM*
Points: *100*

- You may collaborate on this homework with AT MOST one person, an optional "homework buddy".
- MAY ONLY BE TURNED IN ON **GRADESCOPE as a PDF file**. Instructions on How to Submit (applicable to ALL homework assignments in this class) are on Piazza.
- There is NO MAKEUP for missed assignments.
- We are strict about enforcing the LATE POLICY for all assignments (see syllabus).
- **IMPORTANT:** If you use code techniques we have NOT covered in class, you will **get a zero grade** on that problem. If you cheat, or have someone else do your work, you will **get an F in the class**.
  Only use the space provided for answers. Use clear and clean handwriting (or typing).

Reading: Chapter 8.1 and 8.2

1. (10 pts) Is there a difference between the output from these 2 groups of statements if the user enters "**Gaucho Gillian Gomez**" at the prompt each time? Explain why.

```
string name;                                 string name;
cout << "Enter name: ";                      cout << "Enter name: ";
cin >> name;                                 getline(cin, name);
cout << name;                                cout << name;
```

Name:

(as it would appear on official course roster)

2. (10 pts) Which of these is correct usage (syntax) of a stand-alone instruction (i.e. single statement) on a string variable called **`message`** and which of these is incorrect usage (and say ***very briefly*** WHY). Variables **`n`** and **`m`** are int types.

```
a. message.erase(n, m);
```

```
b. message = message.erase(n, m);
```

```
c. cout << message.find(n);
```

```
d. message.size() = n;
```

```
e. cout << message.rfind(“x”);
```

3. (20 pts) Given the declaration of a **<u>C-string</u>** variable, where **`MAX`** is a defined constant:

```
char mychars[MAX];
```

The C-string variable **`mychars`** has been assigned in code not shown here. For correct C-string variables, the following loop reassigns all positions of **`mychars`** the value 'z', leaving the length the same as before. Assume this code fragment is embedded in an otherwise complete and correct program. Answer the questions following this code fragment:

```
int position = 0;
while (mychars[position] != '\0'){
    mychars[position] = 'z';
    position++;
}
```

a. (10 pts) Explain how this code can destroy the contents of memory beyond the end of the array.

b. (10 pts) Modify this loop to protect against inadvertently changing memory beyond the end of the array.

Name:
(as it would appear on official course roster)

4. (20 pts) Show the output produced when the following code (entire program not shown) is executed. IF THERE'S AN ERROR IN THIS CODE, point it out and tell me why it is not correct. You are encouraged to also try to compile this in a program to verify your results.

```
string name = "Porcupine Tree";
cout << "NAME = " + name << endl;
cout << name.length() << endl;
name.erase(8, 6);
cout << name << endl;
name.append("Dean WD Morgan");
cout << name << endl;
name.insert(22, "@TWD");
cout << name << endl;
name.replace(23, 3, "The WD");
cout << name << endl;
cout << name.find("WD") << endl;
cout << name.rfind("WD") << endl;
cout << name.rfind("cupi") << endl;
for (int i = name.length(); i > 20; i--) {
      cout << name[i-1];
      cout << endl;
}
```

Name:
(as it would appear on official course roster)

5. (20 pts) Write the full definition of a function called **FunString()** that takes a string argument and does 2 things: it prints the string backwards and it reports on how many words the string has (a word is separated from others with *space* characters). For example, if the argument is “`Hello my friend!`”, the function should print out “`!dneirf ym olleH`” on one line and then the number **3** on the next line. Another example is: if the argument is “`fish-and-chips`”, then the output would be “`spihc-dna-hsif`” and then the number **1** on the next line. Assume the function will never have an empty string as argument.

Name:

(as it would appear on official course roster)

6. (20 pts) Write a full definition for a function called **`CheckIfNumbers()`** that takes in a string argument and checks if each character in the string is a numerical character (i.e. ‘0’, ‘1’, etc…). If all characters pass this test, then the function returns a Boolean true, otherwise, it returns false.