CS162 - Welcome!

- Introduction and Review
- Introduction:
  - Syllabus,
  - Objectives for the Course,
  - Project and Exams
  - Review Course Outline
- List areas you would like reviewed.....

CS162 - Project

- In CS162, there is one large project, turned in at 4 stages.
- The intent is to incorporate all that you learned in CS161 in solving a “real world” programming problem,
- And, apply some of what we learn this term in the assignment.
- It is large, so begin on it as soon as possible!

CS162 - Project

- One of the goals is to learn how to create a user friendly environment
- This means you should assume that the user doesn’t know anything about computer programming
- This means you will need to carefully prepare prompts, echo all input, provide labels for your output, and...

CS162 - Project

- Error check (i.e., the user may enter invalid data).
- The types of things to check for include:
  - User typing in too many characters.
    - make sure to throw these away using `cin.get`
  - User types in an incorrect option
    - you prompt for options 1-5 and they enter 99
  - User types in lower versus upper case
    - you should accept either! (Y, y, N, n, No, NO, YES, yes,...are all valid confirmation!)

CS162 - Project

In your project you will need to use:

- structures and array of structures
- classes (we will learn about these...)
- pass all objects of a structure or class by reference --- NEVER by value!
- no global variables are allowed (global constants are fine, however)
- external data files (fstream)
- your main program should be very small

CS162 - Project

In your project keep in mind:

- Use call by reference instead of call by value whenever possible to improve efficiency.
- Use iostream (and fstream) libraries. Do not use stdio.h for your I/O
- Display a menu of items the user can select from. Remember to allow the user to quit!

CS162 - Review

Let’s list the areas that you would like us to review this week:

- pass by reference vs pass by value?
- defining arrays of characters?
- reading strings using 3 argument cin.get?
- structures? arrays of structures?
- passing structures by ref vs by value?
- reading/writing external data files?
- others?

CS162 - Review

Why use call by reference?

- supply a value back to the calling routine
- more effective use of memory

Why use call by value?

- only when you need a spare and duplicate copy of the data or if passing fundamental data types (like an int, short, char)

Why use constant references?

- `void print(const float & data);`

CS162 - Review

How is an array passed to a function?

- what does the function call look like?
- what does the prototype look like?
- is there any way to pass an array to a function by value? vs. by reference?
- this term it is important to realize that the name of an array is a constant address of the first element in the array. It is that which is passed (by value)!!!!

cs162 slide 1-9

CS162 - Review

Reading in arrays of characters:

- what is the advantage/disadvantage of:

```
char s[20];cin >>s;
```

- what is the advantage/disadvantage of

```
cin.get(s,20, '\n');
```

- what does this do:

```
while (cin.get() != '\n');
```

or

```
cin.ignore(100,'\n');
```

CS162 - Review

Reading in arrays of characters:

- after using

  ```
  cin >> any_variable;
  ```

- what is left in the input buffer? what will it do to a subsequent call to:

  ```
  cin.get(s,20, '\n');
  ```

- remember, `cin.get` does not skip leading whitespace (nor does cin.getline).
- `cin.getline` should not be used this term, as some compilers will hang if the user types in more than the specified # of characters.

CS162 - Review

- What is the purpose of a function declaration (i.e., prototype)
  - to allow a function to be called even if it is defined (i.e., implemented) later or in some other file.

- What about defining arrays, can the size be variable? (no!)
  - remember to allow 1 character in a “string” for the ‘\0’ (terminating nul)