

CS150 Exam #3 Review Topics

1. Nested Loops

Write a nested loop that asks the user to enter a value between 1 and 10. Print the following pattern based on the user's input.

```
1
22
333
4444
...
```

2. Functions

- Why use functions?
 - Function prototype versus function definition
 - Function arguments versus function parameters
 - Pass by value versus pass by reference
 - Function body
 - Scope of a variable
 - Rules for parameter lists and passing arguments to functions
- ### 3. What is the output?

```
void changeIt (int &i, int j, int &k)
{
    ++i;
    k = i + j++;
}
int main ()
{
    int i = 1, j = 2, k = 3;
    changeIt (i, j, k);
    cout << i << j << k << endl;
    changeIt (k, i, j);
    cout << i << j << k << endl;
    changeIt (i, i, i);
    cout << i << endl;
    return EXIT_SUCCESS;
}
```

4. Arrays

- Declaration
- index
- element
- bounds checking

- e. initializer list
 - f. implicit versus explicit array sizing
 - g. parallel arrays
 - h. passing arrays and regular variables to functions
 - i. How are arrays passed ... by value or reference?
5. Problem (Easy):
- a. Write a function **switchIt** that accepts an integer array and the number of elements in the array.
 - b. Switch the first and last values in the array. For example, if the array is `int values[]={1,2,3,4}`, then after calling `switchIt`, the array would be 4, 2, 3, 1.
 - c. Show what a call to your function would look like.
6. Problem (Hard):
- a. Write a bool function **containsDuplicates** that accepts an integer array and the number of elements in the array.
 - b. Return true if there are duplicates in the array; otherwise, return false
 - c. Show what a call to your function would look like.
7. Problem: A data file `students.txt` exists containing the following student information:
- ```
lastName firstName age gender
```

Assuming an unknown number of students exist (but at most 100), write a C++ program that:

- a) reads in all student information into parallel arrays by calling a function **readStudentData**
  - b) prints the average of all students ... call **computeAvgAge** from the main function
  - c) prints all students of a particular gender ... call a **functionprintStudentGender** to do this
8. Help videos:
- a. Functions:  
<http://zeus.cs.pacificu.edu/PacificCSVideos/c++/functions.html>
  - b. Arrays:  
<http://zeus.cs.pacificu.edu/PacificCSVideos/c++/array1d.html>