

CS 161

Introduction to CS I

Lecture 10

- Design documents
- Practice proficiency demo
- String functions
- Review for Midterm 1



Assignment 3 – Design Document

- What is the goal of a design document?
 1. Clear statement of the goals/problem and any assumptions
 2. High-level concept for how to solve the problem
 3. Test cases to check whether the problem has been solved
- Design should be transferable, not specific to C++
 - A good design could be used by another person to implement a solution in Java, Python, Ruby, etc.



Practice proficiency demo: Week 5 Lab

- **I want you all to ace this!**
- Duration: 50 minutes (the rest of the lab is a typical but shorter lab)
- Our laptop, not yours
- No Internet access
- Editor: vi, vim, or emacs
- Example question:
 - "Write a do-while loop that asks the user to enter three numbers. The program should determine which of the three numbers is largest and print it out. The user should be asked if he or she wishes to perform the operation again. If so, the loop should repeat; otherwise it should terminate."
- Grade: 1, 3, or 5 points
 - If you get a 1, consider this a red flag. Don't panic, but do come to TA or instructor office hours and work to gain experience so you will pass the final Proficiency Exam (week 10).

What happens when you call a function?

- Function definition:

```
int sum(int a, int b)
{
    return a + b;
}
```

- Function call:

```
int main()
{
    int r = 3, s = 5;
    int sum_rs = sum(r, s);
    return 0;
}
```

- \Rightarrow means "evaluates to"
- $\text{sum}(r, s) \Rightarrow \text{sum}(3, 5)$
- $\text{sum}(3, 5) \Rightarrow$
 - $a \Rightarrow 3, b \Rightarrow 5$
 - $a + b \Rightarrow 3 + 5 \Rightarrow 8$
 - return 8
 - 8
- $\text{int sum_rs} = ?$

String functions

- Get string length: `s.length()`
- Get character at position `i`: `s.at(i)`
- Get character at position `i` (indexing): `s[i]`
- Read in a full line (to `\n`): `getline(cin, s)`
- Reference: <http://www.cplusplus.com/reference/string/string/>

Midterm 1: Friday 1/31

- **I want you all to ace this too!**
- Covers material through the **end of week 3** (no functions)
- You cannot use cell phones, calculators, tablets, laptops, or other devices, notes, books, Internet access, friends, etc.
- You will be required to sign a Statement of Academic Integrity on the exam for it to be graded
- If you need scratch paper, raise your hand (it will be collected)
- **Thursday: Evening review – 6-7 p.m. in KEC 1001**
- **Friday: Bring your ID to exam (otherwise we cannot accept it) and writing utensils**

Midterm 1 format

- Duration: 50 minutes
- True/false questions
- Multiple choice questions
- Short answer questions
- 2 example midterms, with solutions, are posted on the course website (and will be used during the Thursday review)

Midterm 1: Data storage (1)

- Variable, literal, const variable
 - Give an example of each
 - `int soccer_score;`
 - `cout << 5 << endl;`
 - `const float pi = 3.14159;`
 - Which ones can be modified later in the program?
 - Only variables

Midterm 1: Data storage (2)

- Variable declaration, initialization, assignment
 - Give an example of use of an **undeclared** variable

```
cout << "Value is " << my_val << endl;
```
 - Give an example of a use of an **uninitialized** variable

```
int soccer_score;  
cout << "Value is " << soccer_score << endl;
```
 - Which ones cause compiler errors?
 - Undeclared (but not uninitialized)

Midterm 1: Data types

- What C++ primitive types have we discussed?
`int, double, char, long, float, bool, short`
- What type have we used that is not a primitive type?
`string`
- Order these data types from smallest (#bits) to largest:
`int double char long float short`
`char, short, int/float, long/double`
- Does using `unsigned` increase or decrease the largest representable value?
`increase`
- What is the largest value that can be stored in a (signed) `short`? (Equation is fine)
 $2^{(b-1)}-1$ or $2^{(16-1)}-1$ or 32,767

Midterm 1: Operators and expressions

- Operators: Arithmetic, relational, logical
 - Give an example of each:
Arithmetic: +, -, *, /, ++, -- Relational: <, <=, >, >=, ==, != Logical: &&, ||, !
 - Higher precedence: && or || ? &&
 - Expressions: What do these evaluate to?
 - $7 - 9$ -2
 - $(6 + 4)\%3 * 13$ 13
 - `int x = 4; bool z = true && (x = 3);` `z = true`
 - `int x = 4; bool z = true && (x == 3);` `z = false`
 - $14/5$ 2

Midterm 1: Conditional statements

- if-then, switch
- Can conditional statements be nested?
- What does `break` do in a switch statement?

Yes

Prevent execution of next case

```
switch (n) {
  case 1: s += 4; break;
  case 2: s -= 4; break;
  default: s = 0;
}

switch (n) {
  case 1: s += 4;
  case 2: s -= 4;
  default: s = 0;
}
```


Midterm 1: Loops

- for, while, do-while
- How many times will this loop execute:

$$3 * 4 = 12$$

```
for (int s = 0; s < 5; s += 2)
{
    for (int t = -3; t >= -6; t--)
    {
        cout << s + t << endl;
    }
}
```

Midterm 1: Loops

- for, while, do-while
- What does this loop do:

```
bool alive = true;
while (alive)
{
    if (rand()%5 == 0)
    {
        alive = false;
    }
}
```


Midterm 1: Strings

- How do you declare a string?
- What do you need to #include?
- How do you read user input into a string?
- How do you append to a string?
- How do you concatenate two strings?

```
string str;
```

```
<string>
```

```
cin >> str;
```

```
str += new_str;
```

```
str + new_str;
```

Midterm 1: Design

- Design documents
 - Show understanding of the problem
 - Plan: pseudocode or flowchart
 - Test cases
- Debugging
 - Syntax, logic, and runtime errors
 - How can you detect errors?
 - How can you locate the errors?

Midterm 1: Linux

- How do you...
 - Enter a directory called "dir"?
 - Create a directory called "dir"?
 - Delete a file called "f1"?
 - Rename a file from "f1" to "f2"?
 - Print the current directory?
 - Go "back" or "up" a directory
 - Go to your home directory

```
cd dir
```

```
mkdir dir
```

```
rm f1
```

```
mv f1 f2
```

```
pwd
```

```
cd ..
```

```
cd ~      or      cd
```

What other questions do you have?

Week 4 continues

- Prepare for Midterm 1 (Friday, Jan. 31) – review past midterms and answers, ask questions on Piazza, come to office hours
- (Optional) Attend Midterm 1 Review (Thursday, 6-7 p.m. in KEC 1001)
- Attend lab 4 (laptop required)
- Continue working on **Assignment 3 design** (due **Sunday, Feb. 2**)

See you Friday! **Bring your OSU student ID!**