SCC.111: GETTING STARTED WITH C

THE GOAL

This week's lab aims to:

- The goal for today is to get you really familiar with starting simple C projects and C compiler toolchain.
- There are three problems of increasing difficulty. There is also a more advanced 'hacker edition' if you are a more experienced programmer or complete all these exercises in good time.
- As a side effect, you will also be getting more familiar with the lab environment we introduced last week. If you missed this, then you're already behind: *make sure you've done the UNIX tutorial from last week FIRST*

Instructions on how to use the virtual lab environment (VDI) and/or install the compiler on your own machine are available at the course's moodle page.

TODAY'S TASKS

Problem 1: "I love SCC.111"

Your task is to create a derivative of the classic 'hello world' program, and get it compiling, fixing any errors on the way.

1. Start a new program (create a new text file on UNIX, e.g `code helloworld.c`, or by launching `code` and saving as `helloworld.c`).

```
Don't forget: files outside `h-drive` may be wiped overnight, so keep your files
↪ safe!
```

2. Type **and do not copy & paste** any program code. Typing helps you remember and understand the rules/ syntax of the language. *Making typing mistakes and fixing the resulting errors is essential to your learning on this course!*

```c
#include <stdio.h>


int main()
{
    printf("Hello world\n");
}
```

3. Save the program

4. Ensure that your program compiles (fix any errors flagged by the compiler using the editor, and compile again until no errors are generated). **You should make notes in your lab book so that you remember how to do this.**

   You can compile from the terminal with something like:

   ```
   gcc -o hello helloworld.c
   ```

   - `hello` is the output file, specified with `-o`. If you forget this, you'll get `a.out` as the executable file
   - `helloworld.c` is the name of the source file, notice the lower case `.c`

   When it compiles with no errors, `hello` is created. You can run this with `./hello` where `./` is *the relative path* to where the file is (the current folder).

5. Once that works, adapt the program to print 'I love SCC111'. *You'll need to save and recompile every time you've changed your program source code!.*

PROBLEM 2: "SOLVING OUR FIRST PROBLEM"

Start a new program (create a new file and choose a meaningful filename for your program). *Remember that the filename of C source code always needs to end in* `.c`.

VS Code (and many other editors) will also recognise you're editing C code and highlight the syntax of what you type in colour, which is nice! (you may need to select 'C' rather than 'plain text' using the menu bar at the bottom right).

The first problem we're going to solve is to create a program, that:

1. Given the length of a leg bone, estimates how tall the person was. This is commonly done in forensics when human remains are found or exhumed (read more here).

2. Note that the estimates are based on regression analysis of European population data, so is biased by where this sample is (i.e. would likely change for other parts of the world).

3. Given the length of the femur (hip to knee), calculate the estimated height and print out the result. Helpful code:

   ```
   float femurLength = 0.0;
   ```

   Declare a floating point variable `femurLength` (note the camelCase) and set it some value (`0.0`).

   ```
   scanf("%f", &femurLength);
   ```

   Input a floating point number `%f` and put the value in our variable specified.

   ```
   printf("Femur length = %f\n", femurLength);
   ```

   Print out to the terminal the string (using *print with format string*, or `printf`). Note how `%f` shows the type (floating point) and position of where to print the number. `\n` means go to the next line after printing the string. You will have too many decimal places, read up on `printf` to learn how to change `%f` so that it prints to just two decimal places. **You should get used to looking up functions to see how they work and what their options are**.

4. Extend your program to output the height based on another bone length, e.g. the Tibia.

5. Extend your program to print the upper and lower bounds of the height range using the confidence intervals specified in the conversion table.

PROBLEM 3: "SQUARES"

Our third and final problem for today is to print out a square using text characters. You should set a side length in your program and print out an appropriately sized square. A side length of 3 would output:

```
***
***
***
```

whereas a side length of 5 would yield:

```
*****
*****
*****
*****
*****
```

For additional credit and challenge, create a version of your program that prints a 'hollow' square, thus:

```
***
* *
***
```

REFERENCES

For a description of the stdio library see either http://en.wikipedia.org/wiki/C_file_input/output or the textbook, though note that there is *way more detail* than you need listed on this page!

SCC121: SETS

This week's lab activities will cover the topic of Sets. Set theory plays a fundamental role as it provides a mathematical foundation for dealing with collections of data and solving various computational problems. It offers a foundation for defining, manipulating, and reasoning about sets and their operations. Its applications in CS include data structures, algorithms, databases, and formal reasoning, making it an essential topic for computer scientists and software engineers.

THE GOAL

Using the knowledge of the lecture material from Week 2, you are expected to work through the following questions. You should work out the answers *with a pen and paper* - **no calculators nor computers**!

By the end of this task you should be able:

- ☐ To understand what is a Set.
- ☐ To define a Set.
- ☐ To apply operation between sets, including performing unions, intersections and set differences.

TODAY'S TASKS

1. Determine whether each of the following pairs of sets is equal.

   - {1, 3, 5} and {5, 3, 1}
   - {1, 3, 5} and {5, 1, 6}

2. Let A = {1, 2, 3, 4, 5} and B= {0, 3, 6}. Find:

   - $A \cap B$
   - $A \cup B$
   - $A–B$
   - $B - A$

3. Let A = {0, 2, 4, 6, 8, 10}, B = {0, 1, 2, 3, 4, 5, 6} and C = {4, 5, 6, 7, 8, 9, 10}. Find:

- $A \cap B \cap C$
- $A \cup B \cup C$
- $(A \cup B) \cap C$
- $(A \cap B) \cup C$

4. If A = {1, 2, 3, a}, B = {1, 2, 3, 4, 5}, C = {a, b} evaluate:

- $A \cup (B \cap C)$
- $(A \cup B) \cap C$
- $C–A$
- $(A–B)–C$
- $A–(B–C)$
- $(A \cap C) \cup B$
- $A \cap (C \cup B)$

5. Let A be the set of students who live within one mile of school and B the set of students who walk to classes. Describe the students in each of the following sets, in English.

- $A \cap B$
- $A \cup B$
- $A–B$
- $B–A$

6. Let A = {a, b, c} and B = {y, z}. Find

- A x B
- B x A

7. How many different elements does A x B have if A has m elements and B has n elements?

8. List the members of the following sets

- {x | x is a positive integer less than 12}
- {x | x is the square of an integer and x < 100}

9. For each of the following sets, determine if 2 is an element of that set

- $\{ x \in R \mid$ x is an integer greater than 1$\}$
- $\{x \in R \mid$ x is the square of an integer$\}$

10. Identify 2 sets whose cardinality is 1 and whose intersection's cardinality is also 1.

11. Which of the following is not a proper subset of set A if A = {4, 5, 6, x, y, z}?

- {4, x, y}
- {3, 4, 5}
- {4, 5, 6}
- Ø

12. Which is a subset of set A? A = {a, {b}, c, {c}, {x, y}}

- {a, {b}, c}
- {a, b, c}
- {c, x, y}
- {x, y}

13. Which is false?

- $1, 2 \subseteq 1, 2, 3$
- $1, 2 \subseteq 2, 3, 4$
- $\emptyset \subseteq 1, 2, 3$

14. Let Universal set U = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}, and its 3 sets: E = {2, 4, 6, 8, 10}, O = {x | x is an odd number}, and T = {3}.

- Enumerate the elements of set O
- What operation can be applied to set E and U to get set O?
- Define set E through its elements' property
- Which of the sets above are subsets of set O?
- Which of the sets above are proper subsets of set O?
- Which of the sets above are proper supersets of empty set?

SCC.131: NUMBER SYSTEMS

In the second week of the module, you were introduced to number systems. A **number system**, also known as a numeral system or base, is a mathematical notation system used to express numbers. Different number systems use different symbols or digits to represent numbers, and they also have different bases or radixes, which determine how numbers are organized and counted. The most commonly used number systems include:

- *Decimal System (Base 10)*: The decimal system is the one most familiar to people. It uses ten digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9. Each digit's position represents a power of 10, starting from the right with 10^0 (1), 10^1 (10), 10^2 (100), and so on.
- *Binary System (Base 2)*: The binary system uses only two digits, 0 and 1. It's commonly used in computer science because it's the basis for digital representation in computers. Each digit's position represents a power of 2, starting from the right with 2^0 (1), 2^1 (2), 2^2 (4), and so on.
- *Hexadecimal System (Base 16)*: The hexadecimal system uses sixteen digits: 0-9 and A-F (where A stands for 10, B for 11, C for 12, D for 13, E for 14, and F for 15). It's often used in computer science for compactly representing binary data. Each digit's position represents a power of 16, starting from the right with 16^0 (1), 16^1 (16), 16^2 (256), and so on.

THE GOAL

Using the knowledge of the lecture material from Week 2, you are expected to work through the following questions. These exercises aim to improve your experience working with binary number. You should work out the answers *with a pen and paper* - **no calculators nor computers**! In case of overflow, just ignore the overflow bit.

TODAY'S TASKS

This is a worksheet for the SCC.131 week 2 lab practical. Work through the following questions and ask us if you need help.

By the end of this task you should be able:

- [ ] Convert numbers between decimal and binary systems.
- [ ] Be able to perform basic arithmetic operations between number of any numerical system.

1. Convert the following unsigned binary numbers to decimal:

   - 00001001_2
   - 00101011_2
   - 01001111_2
   - 01111110_2
   - 10000110_2

2. Convert the following decimal numbers to unsigned 8-bit binary:

   - 22_{10}
   - 19_{10}
   - 84_{10}
   - 120_{10}
   - 93_{10}

3. Add the following pairs of unsigned 8-bit binary numbers. Give your answer in unsigned 8-bit binary.

   - $00000100_2 + 00000011_2$
   - $00000110_2 + 00001101_2$
   - $00001111_2 + 00010101_2$
   - $11101001_2 + 01011110_2$
   - $11100000_2 + 00110000_2$

HACKER EDITION

This section contains activities for those looking for an additional challenge. You are are not expected to complete them during the week, but we are happy to provide feedback or discuss how you can approach to solve them.

SCC.111: PASCAL TRIANGLE

If you've completed the other three tasks and want more, try this:

- Instead of printing a square, create a new program that prints out Pascal's triangle (this and the origins of the triangle in ancient Persia and China, and elsewhere in Europe can be found on Wikipedia).

There are a number of solutions to this, for added challenge try creating your own before emulating these. Note that some of the properties are as follows:

- The numbers result from the coefficients in the expansion of $(a+b)^n$ for $n = 0, 1, 2, 3, ...$
- Each row sums to an increasing power of 2
- Each digit is calculated as the sum of the left and right parents in the row above

Your program should 'pad' the output using spaces, i.e. a triangle of height 4 should look like:

```
   1
  1 1
 1 2 1
1 3 3 1
 ... etc.
```

i.e. if you assume any blank is a zero `0`, the resulting digit is the sum of the digits up and left and up and right in the previous row.

Naturally, there are lots of implementations of this online. The challenge is *not* to find one of them in the first instance…! :smile:

SCC.121: ADVANCED SET EXERCISES

1. Let sets A and B be defined as:

   - $A = \{a \in Z \mid a = 2k, \text{ for some integer } k\}$
   - $B = \{b \in Z \mid b = 2j - 2, \text{ for some integer } j\}$

   Are these equal sets? Prove your answer.

2. Prove the following statement: *The empty set is a subset of every set.*

3. Let's us have the sets $A = \{1, 2, 3\}$, $B = \{x, y\}$, and $C = \{u, v\}$. Let $\mathscr{P}(A)$ denote the power set of A, or the set which includes all the subsets of set A. Find each of the following:

   - $\mathscr{P}(A \cup B)$
   - $\mathscr{P}(A \times B)$
   - $\mathscr{P}(\mathscr{P}(C))$

SCC.131: ARBITRARY NUMBER SYSTEMS

1. Number systems are not limited to a binary, decimal or hexadecimal base, but you can also consider other bases. For an added challenge, consider how to answer the following conversions.

   - Define a base-5 number system. Explain how it works and how it differs from a base-10 (decimal) system.
   - Which symbols would be appropriate to use in a base-5 number?
   - Convert the following base-5 numbers to their decimal (base-10) equivalents: 123_5
   - Convert the following decimal numbers to their base-5 equivalents using the symbols provided: 9_{10}
   - Perform the following mathematical operations in the base-5 system:
     - $3_5 + 4_5$
     - $20_5 + 12_5$