Tutorial 3

MATH10017

October 9, 2025

1 Reading

Read lecture slides on "Memory Allocation for Functions" carefully.

Answer the following questions (similar to previous exam questions).

- What is "stack" data structure?
- When a function is called, where is it placed in the stack memory?
- What is "local variable"? Why is it called "local"?

2 Coding

2.1 Call Stack

We are going to see how the stack is visualized.

- Copy and paste the `f1-f2-f3` example from the slides. Right click the editor and "format code" to nicely reformat your code into a more readable format.
- Add a break point to the beginning of `f1`, `f2` and `f3`.
- Look at the "call stack" pane on the left, are they the same as shown in the lecture slides?

2.2 Local Variable

Open `stack.c`, you should see the following code.

```
#include <stdio.h>
int addtwo(int a) {
    a = a + 2;
    return a;
}

int multiplytwo(int a){
    a = a * 2;
    return a;
}

void main() {
    int a = 0;
    printf("a: %d\n", a);

    addtwo(a);
    printf("a: %d\n", a);

    a = addtwo(a);
    printf("a: %d\n", a);

    a = multiplytwo(addtwo(a));
    printf("a: %d\n", a);

    a = multiplytwo(addtwo(a) / addtwo(a));
    printf("a: %d\n", a);
}
```

- Without running the code, what is the print out?

2.3 Fibonacci Sequence

Open `fib.c`. Write code that print out the first 10 numbers of a Fibonacci sequence.

2.4 Chain Rule (Challenging)

A neural network is essentially a composite function. For simplicity, let us define a neural network with depth n as a composite function g,

$$g(x) = f_1(f_2(\dots f_n(x))),$$

where $f_1 = f_2 \cdots = f_n = \sin$. For example, a two-layer neural network would be $\sin(\sin(x))$.

Open `chain.c`, both f and f' have been defined for you.

- Write a function `double g(double x, int n)`. It evaluates a neural network g with depth n at input x.
- Write a function `double dg(double x, int n)`. It evaluates the derivative of a neural network g with depth n at input x.

You can use loops. However, recursion would make things much simpler.