# Lecture 4

4 September 2018

Admin Matters
Unit 8: **If Else**
Unit 9: **Logical Expression**
Unit 10: **Assertion**

# How to do badly in CS1010

# 1.

# Ignore what Wei Tsang advised or announced during lecture

# 2.

# Do not read / pay attention to the guides and notes posted on CS1010 website

# 3.

# Visit Piazza only when you have a question to ask yourself

# 4.

# Wait for others to ask questions

# 5.

# Wait for solutions from UDLs instead of trying the problems yourself.

# 6.

# Expect assignments / tests / exams to be similar to problem sets / exercises

# 7.

# Copy-paste sample code from lecture instead of writing it out yourself

# Tutorial 3

Problem Sets from Unit 8-9 Today

# Assignment 1

Arithmetic Ops
Recursive Function
If-Else

# Assignment 1

Released this Friday

Due next Friday

# Practical Exam 1

Your venue will be announced on Piazza soon.
Acclimatize yourself.

# Midterm

Venue: MPSH 1 (B)

2 October

4pm - 6pm

# Catch Up Session

This Saturday

UNIX / vim

(the basic)

# Catch Up Session

Must have read through the UNIX tutorial and have gone through vimtutor.

# Catch Up Session

Please register online by end of tomorrow. (see Piazza post)

# Reminder

Use XShell on the PCs in our labs to ssh into the PE hosts and code.

# Reminder

Read plagiarism and late submission policy on the CS1010 website.

# Readings

Overview about your programming assignments.

https://kahoot.it

Kahoot!

```
long square(long x)
{
  return x*x;
}

double hypotenuse_of(long base, long height)
{
  return sqrt(square(base) + square(height));
}

int main()
{
   :
  hypotenuse = hypotenuse_of(base, height);
   :
}
```

Where are we in CS1010?

**Problem Solving**

**C language / syntax**

**Behavioural / Mental model**

**Tools / Good Practice**

Where are we in CS1010?

**Problem Solving**

decomposition
recursion
flowchart

**C language / syntax**

**Behavioural / Mental model**

**Tools / Good Practice**

# Where are we in CS1010?

## Problem Solving

decomposition
recursion
flowchart

## C language / syntax

types in C
functions in C
arithmetic ops

## Behavioural / Mental model

## Tools / Good Practice

# Where are we in CS1010?

## Problem Solving

decomposition
recursion
flowchart

## C language / syntax

types in C
functions in C
arithmetic ops

## Behavioural / Mental model

machine code
data in memory
types

## Tools / Good Practice

# Where are we in CS1010?

| **Problem Solving** | **C language / syntax** |
| --- | --- |
| decomposition<br>recursion<br>flowchart | types in C<br>functions in C<br>arithmetic ops |
| **Behavioural / Mental model** | **Tools / Good Practice** |
| machine code<br>data in memory<br>types | clang<br>vim<br>bash |

# Today

## Problem Solving

decomposition
recursion
flowchart
conditionals
assertion

## C language / syntax

types in C
functions in C
arithmetic ops
if else
logical expressions

## Behavioural / Mental model

machine code
data in memory
types

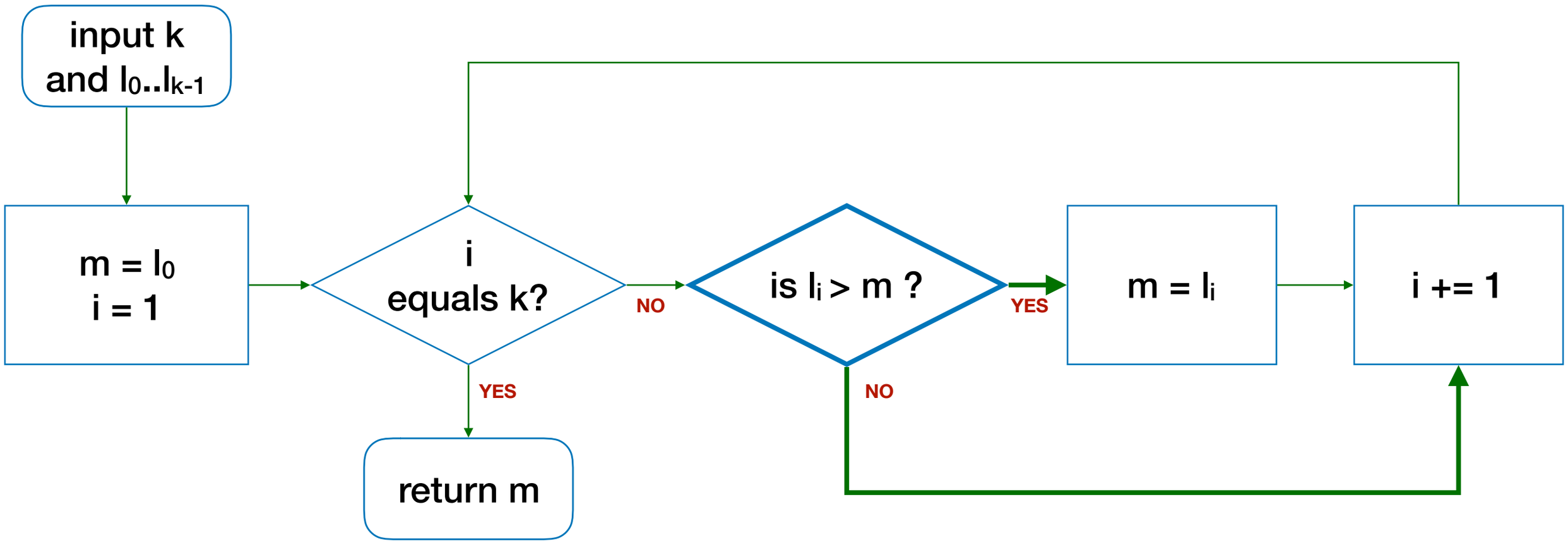
## Tools / Good Practice

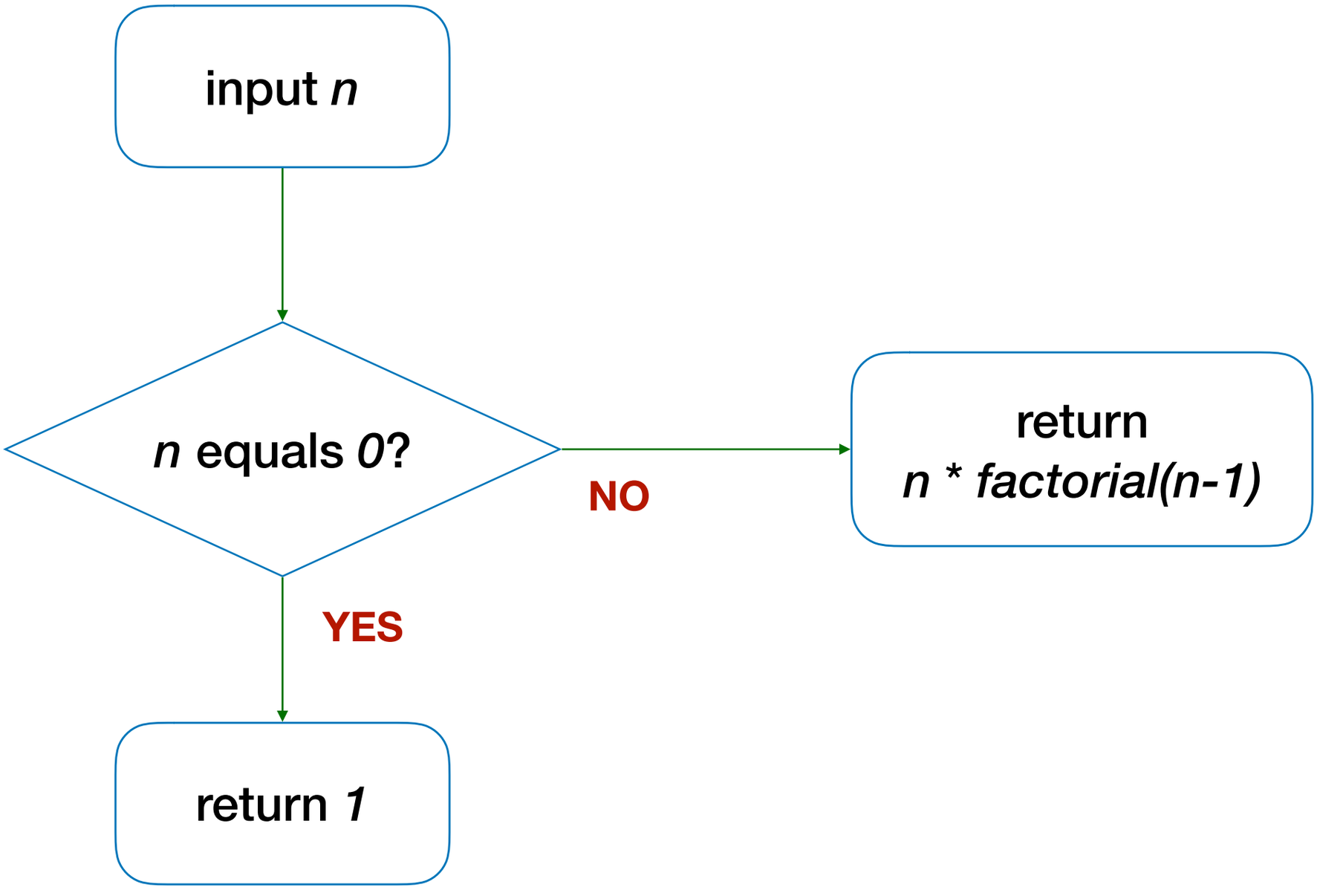
clang
vim
bash

# Lecture 4

4 September 2018

Admin Matters
Unit 8: **If Else**
Unit 9: **Logical Expression**
Unit 10: **Assertion**


input k
and $l_0..l_{k-1}$
$m = l_0$
$i = 1$
i
equals k?
YES
return m
NO
is $l_i > m$ ?
YES
$m = l_i$
NO
i += 1


input *n*
*n* equals *0*?
NO
return
*n * factorial(n-1)*
YES
return *1*

```
long factorial(long n)
{
   if (n == 0) {
      return 1;
   }
   return n * factorial(n - 1);
}
```

# Need to reason about all possibilities

| Score | Letter Grade |
| --- | --- |
| 8 or higher | A |
| Less than 8<br>but 5 or higher | B |
| Less than 5<br>but 3 or higher | C |
| Less than 3 | D |

| Score | Letter Grade |
| --- | --- |
| 8 or higher | A |
| Less than 8 | See Table 1 |

Table 1 (Less than 8)

| Score | Letter Grade |
| --- | --- |
| 5 or higher | B |
| Less than 5 | See Table 2 |

## Table 1 (Less than 8)

| Score | Letter Grade |
| --- | --- |
| 5 or higher | B |
| Less than 5 | See Table 2 |

## Table 2 (Less than 5)

| Score | Letter Grade |
| --- | --- |
| 3 or higher | C |
| Less than 3 | D |

```
if (score >= 8)
  if (late_penalty != 0)
    cs1010_println_string("late submission”);
  else
    cs1010_println_string("you can do better!”);
```

```
if (score >= 8) {
   if (late_penalty != 0) {
        cs1010_println_string("late submission”);
   } else {
    cs1010_println_string("you can do better!”);
   }
}
```

```
if (score >= 8) {
   if (late_penalty != 0) {
        cs1010_println_string("late submission”);
   }
} else {
    cs1010_println_string("you can do better!”);
}
```

# The `bool` data type can take two values `true` or `false`

```
#include <stdbool.h>
```

&&

||

!