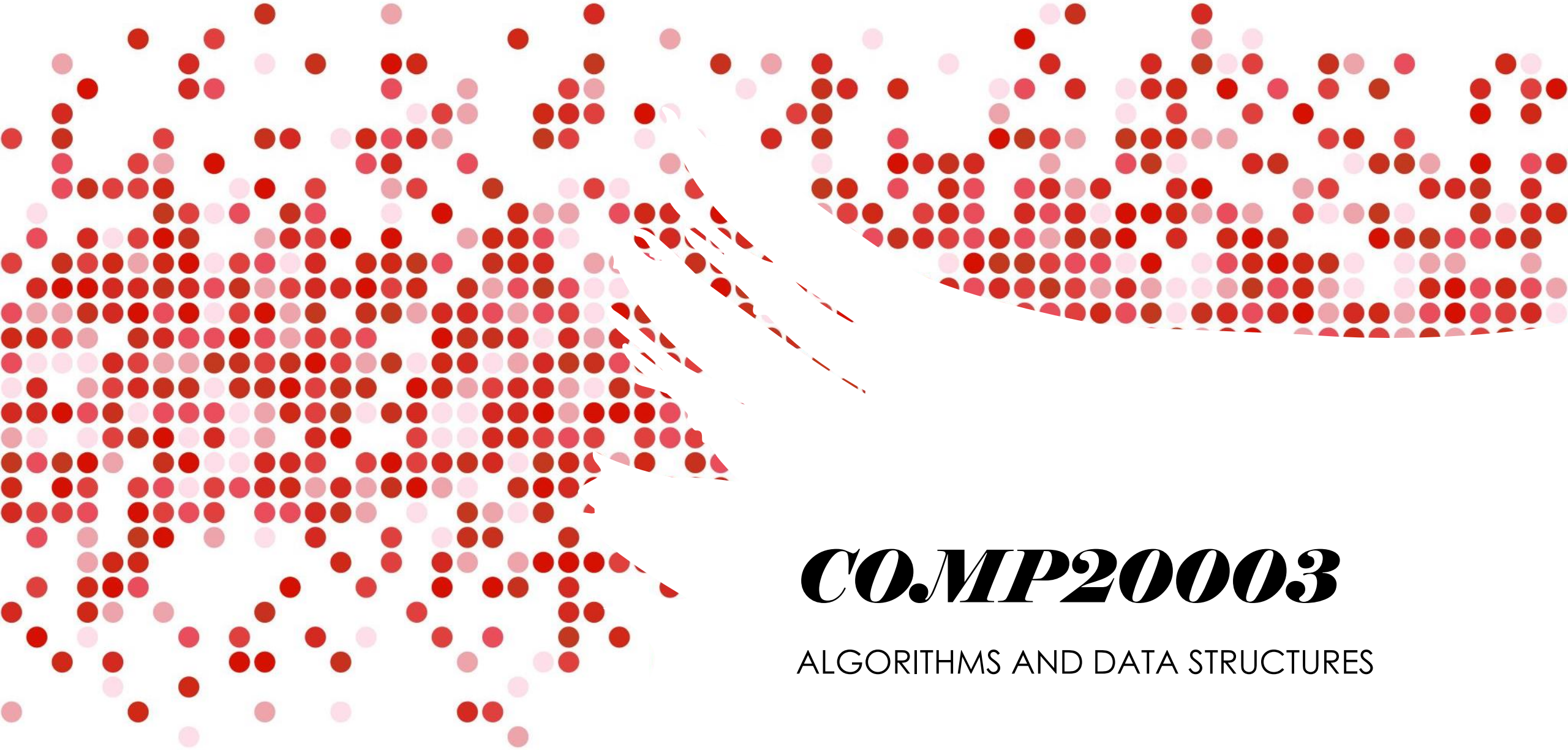


COMP20003

ALGORITHMS AND DATA STRUCTURES

Introduction - Me

- Name
- What I do
- My Background

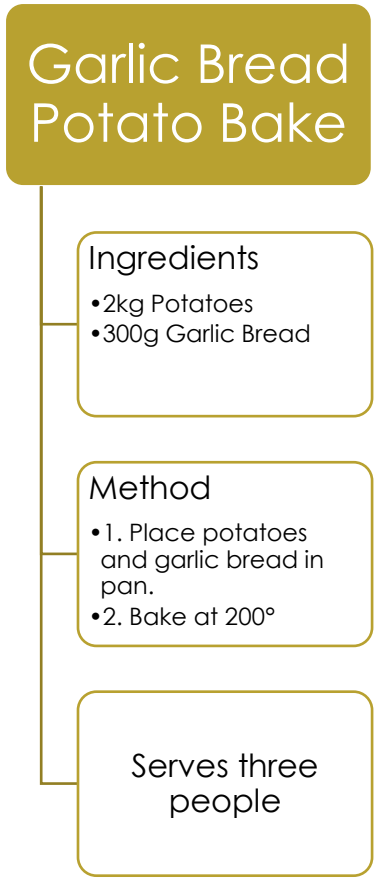
Introduction - Workshops

- Canvas
- Piazza
- Pathways into the subject
- Zoom poll – Should workshops be recorded?

Introduction - You

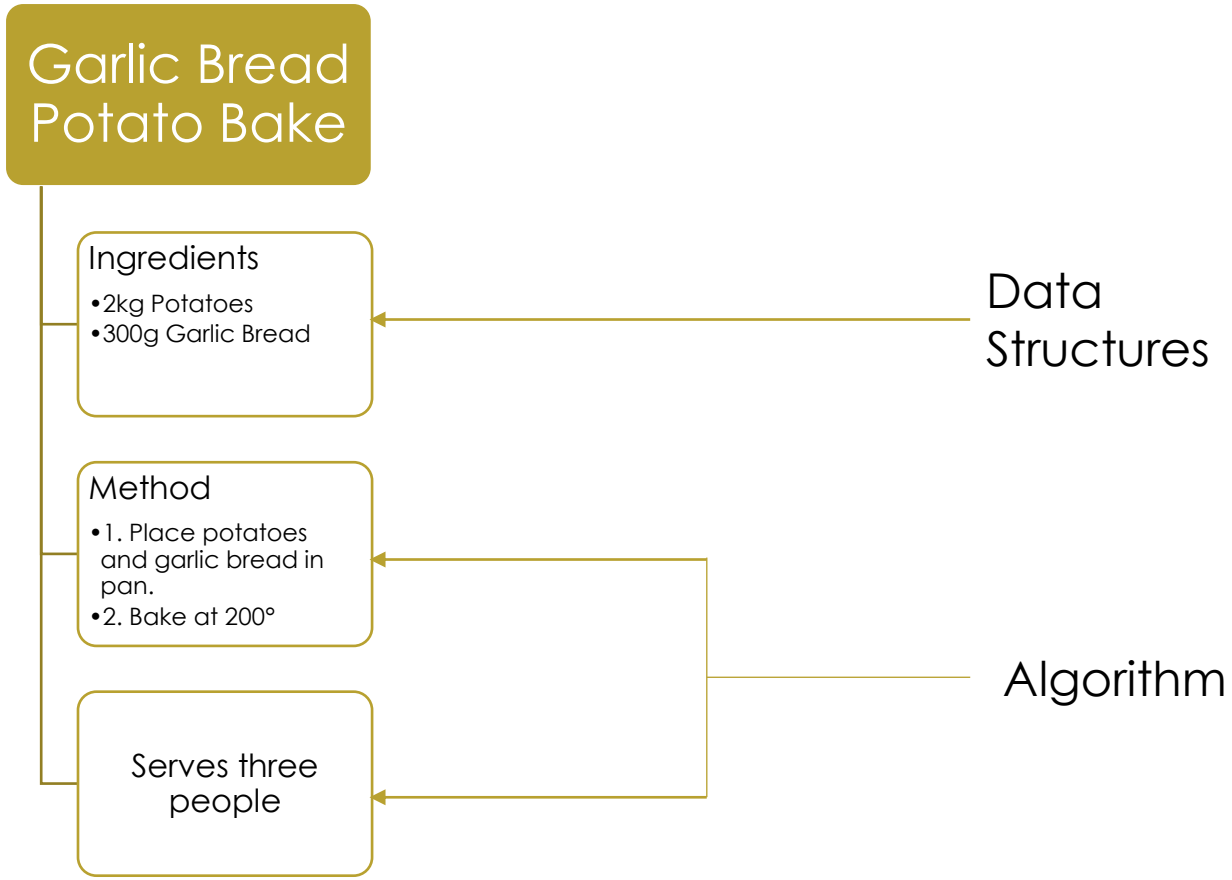
- Zoom poll – Origin Subject
- Zoom poll – Goals for the Subject
- A moment to share contact details with each other

Workshop

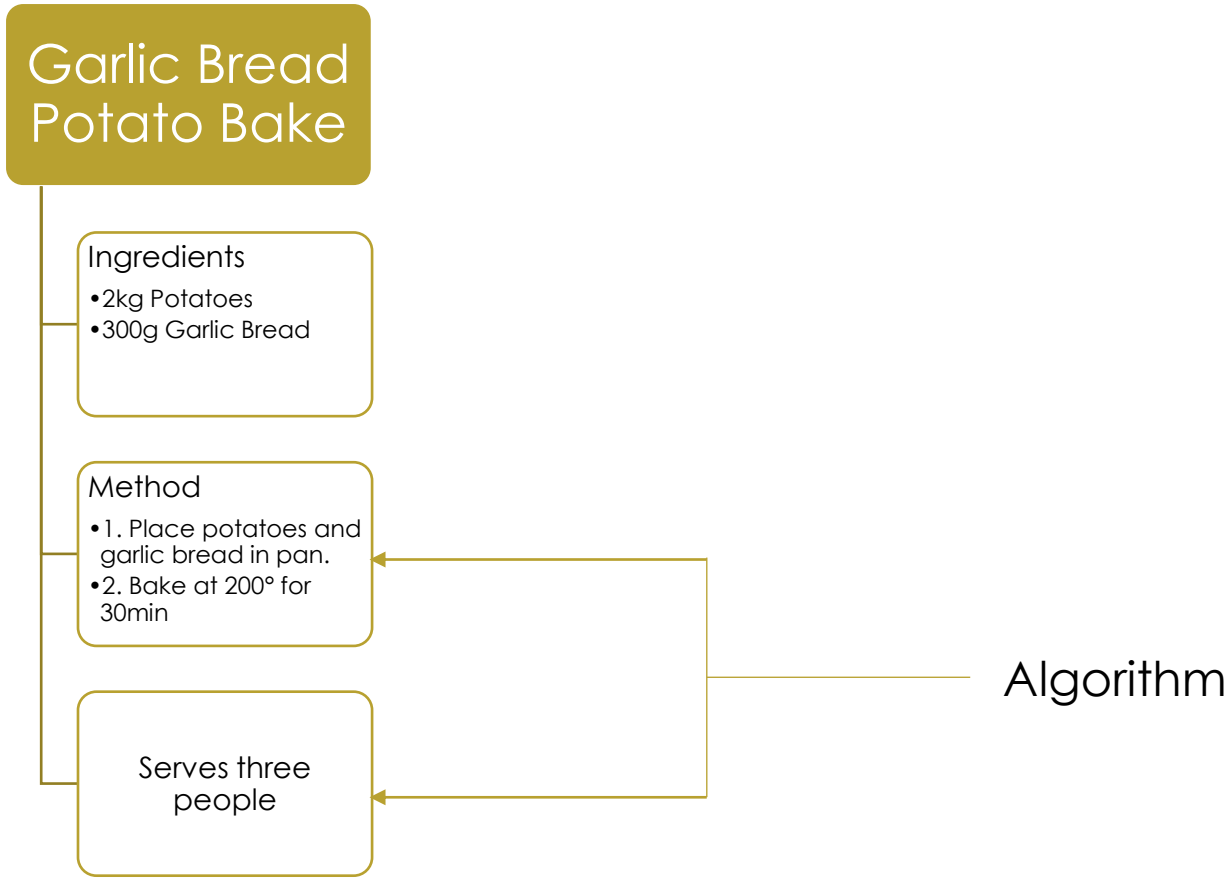


- Which part is the Data Structure?
- Which part is the Algorithm?

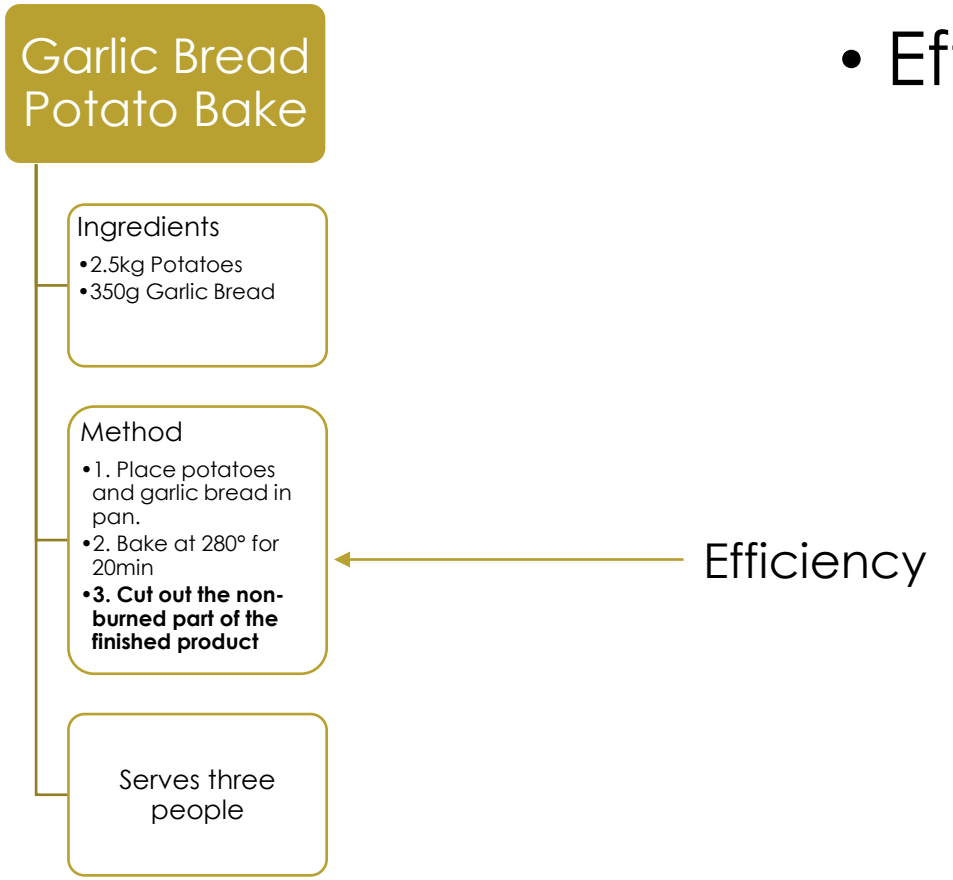
Workshop


Garlic Bread Potato Bake
Ingredients
•2kg Potatoes
•300g Garlic Bread
Method
•1. Place potatoes and garlic bread in pan.
•2. Bake at 200°
Serves three people
Data Structures
Algorithm

Workshop


Garlic Bread Potato Bake
Ingredients
•2kg Potatoes
•300g Garlic Bread
Method
•1. Place potatoes and garlic bread in pan.
•2. Bake at 200° for 30min
Serves three people
Algorithm

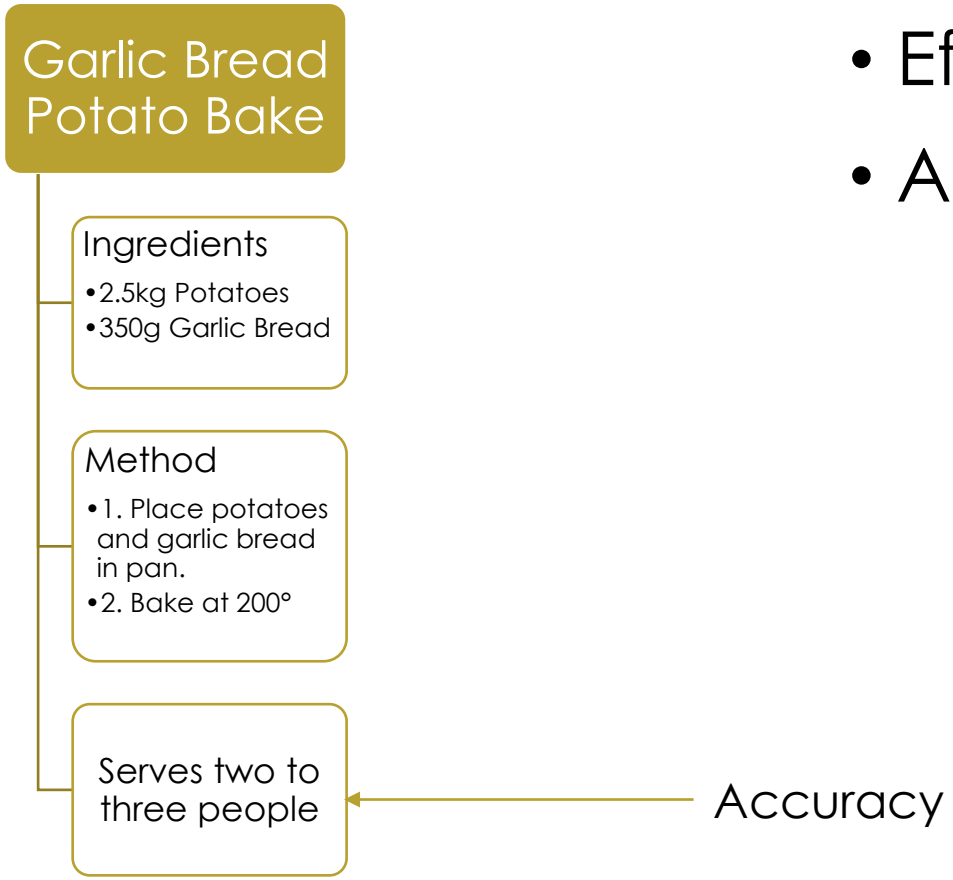
Workshop



- Efficiency

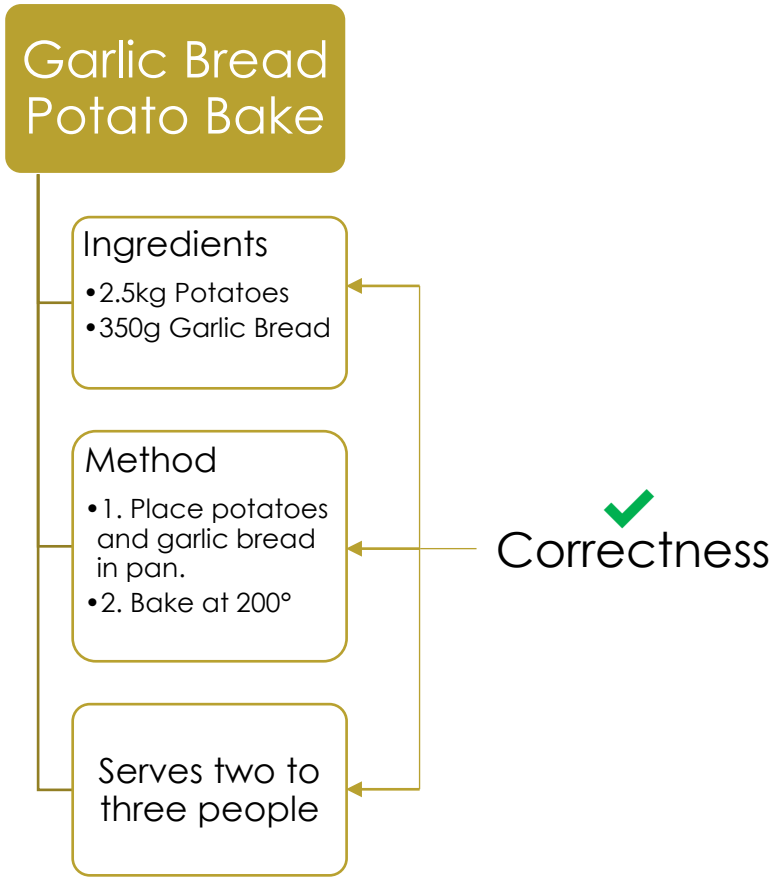
Efficiency

Workshop



- Efficiency
- Accuracy

Workshop



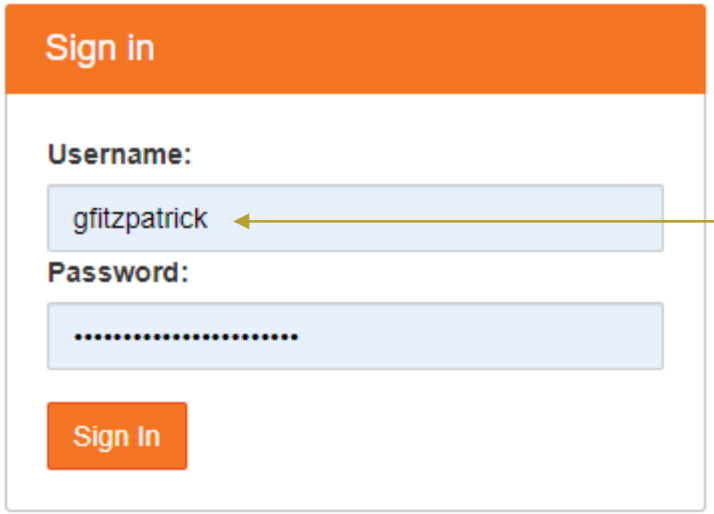
- Efficiency
- Accuracy
- Correctness

Workshop - Demonstration

- Jupyterhub – Visit

https://comp20003-jh.eng.unimelb.edu.au/

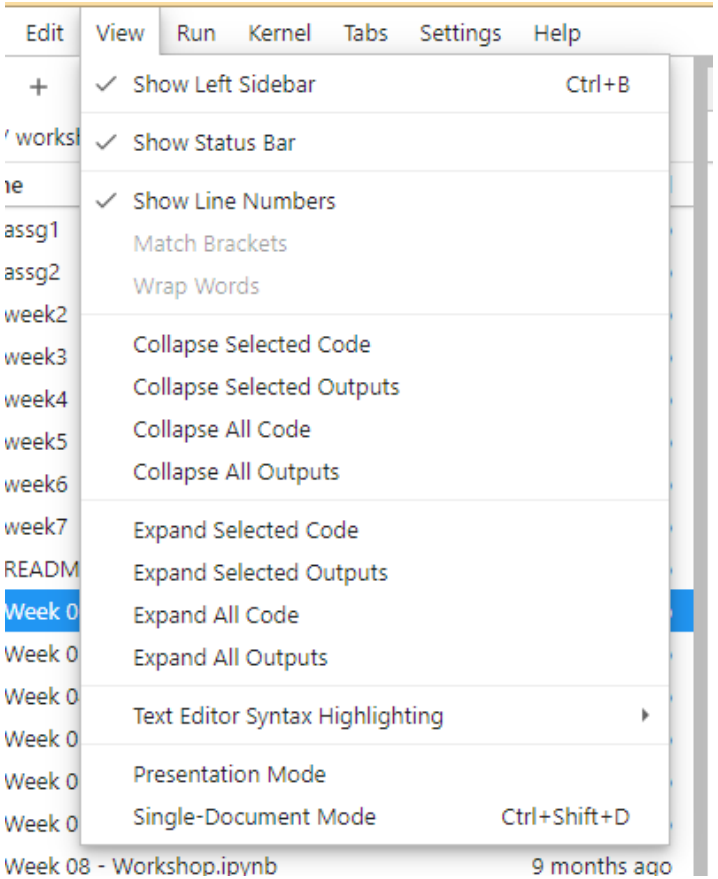
^^^^



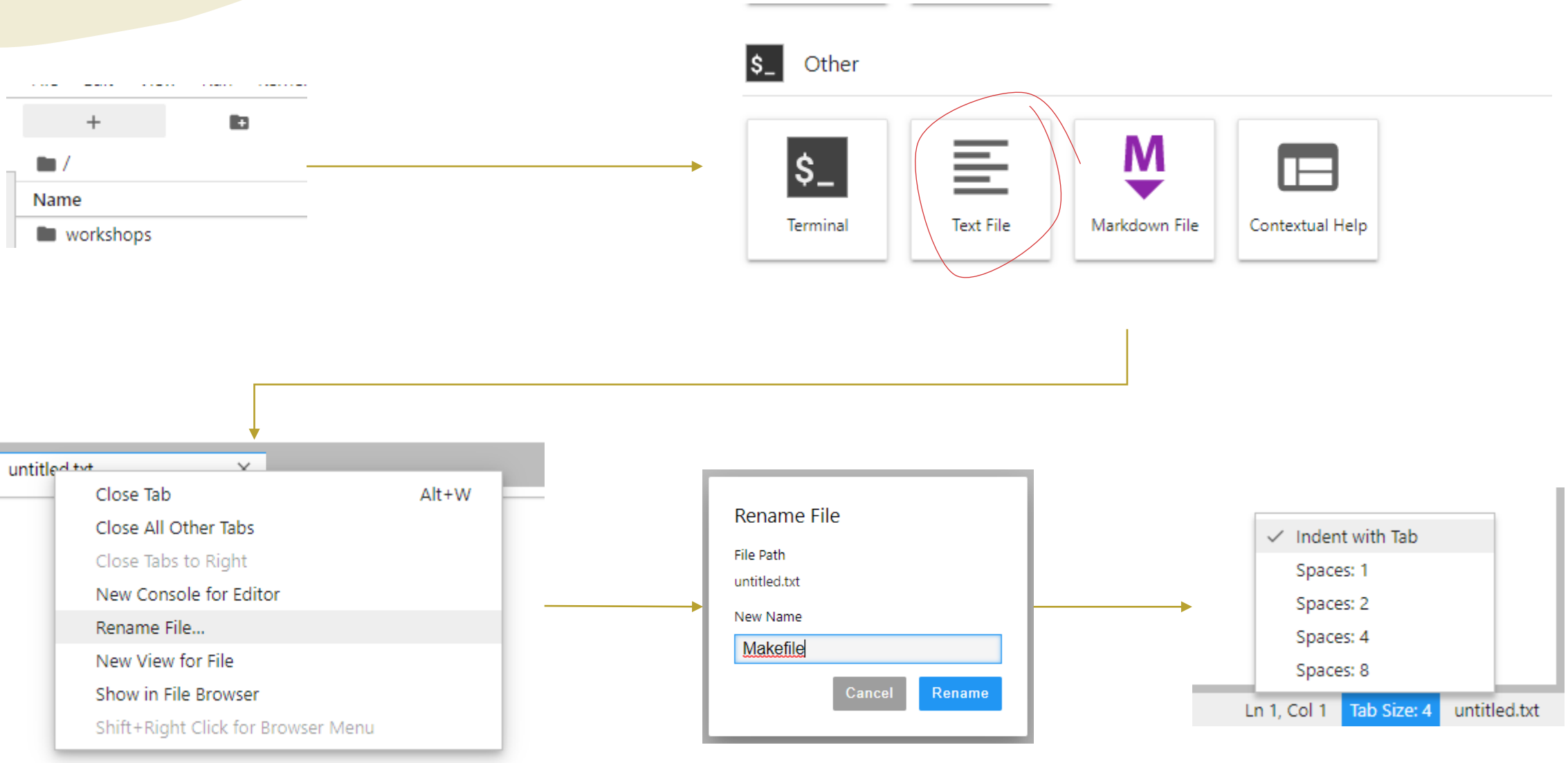
Make sure you type your username in lowercase

Workshop - Demonstration

- Jupyter Notebooks
- Showing Line Numbers



Makefiles


+
/
Name
workshops
$_ Other
$_
Terminal
Text File
M
Markdown File
Contextual Help
untitled.txt
Close Tab Alt+W
Close All Other Tabs
Close Tabs to Right
New Console for Editor
Rename File...
New View for File
Show in File Browser
Shift+Right Click for Browser Menu
Rename File
File Path
untitled.txt
New Name
Makefile
Cancel
Rename
Indent with Tab
Spaces: 1
Spaces: 2
Spaces: 4
Spaces: 8
Ln 1, Col 1
Tab Size: 4
untitled.txt

Makefiles

Target Name

Dependencies

Indented with Tabs (not spaces)

```
1  target_name: dependency_name_1 dependency_name_2
2      echo "target_name - First action"
3      echo "target_name - Second action"
4
5  dependency_name_1: dependency_name_2
6      echo "dependency_name_1 - action 1"
7      echo "dependency_name_1 - action 2"
8
9  dependency_name_2:
10     echo "dependency_name_2 - action 1"
11     echo "dependency_name_2 - action 2"
12
```

Recipe (steps)

```
jovyan@jupyter-gfitzpatrick:~$ make
echo "dependency_name_2 - action 1"
dependency_name_2 - action 1
echo "dependency_name_2 - action 2"
dependency_name_2 - action 2
echo "dependency_name_1 - action 1"
dependency_name_1 - action 1
echo "dependency_name_1 - action 2"
dependency_name_1 - action 2
echo "target_name - First action"
target_name - First action
echo "target_name - Second action"
target_name - Second action
```

Uses first target in file

Uses target name specified

```
jovyan@jupyter-gfitzpatrick:~$ make dependency_name_1
echo "dependency_name_2 - action 1"
dependency_name_2 - action 1
echo "dependency_name_2 - action 2"
dependency_name_2 - action 2
echo "dependency_name_1 - action 1"
dependency_name_1 - action 1
echo "dependency_name_1 - action 2"
dependency_name_1 - action 2
```

Malloc and Free

- malloc(5 * sizeof(int)) → returns pointer to space for 5 ints
- free(pointer) → frees a pointer **allocated with malloc**
- Assert/check all mallocs
- Free all mallocs after you've finished with them

Debugging with GDB

- Test cases → Simple which you know should work – One line or two lines of input, small numbers, etc.
- GDB → Slides on Canvas.

Pair Programming Exercises

- Log on to JupyterHub and work with your partner to solve the problems.
- One person drive the keyboard and the other work on solving the problem, after 15 minutes, try swapping roles.
- Check out man rand on the terminal, exit by pressing ‘q’