Henry Routson
Henry_Rou@ProtonMail.com 0419 108 859 https://www.linkedin.com/in/henryroutson/

Experience

Leetcode programming
https://leetcode.com/HenryRoutson/

113 practice programming problems solved in numerous languages including
Python, C and Rust.
Over 1.4 thousand solution views.

ARES Software developer on Flight simulations 2023 - Present
https://eng.unimelb.edu.au/ares

Collaborating with industry software developers
and Masters Computing students
in Typescript and React 3JS Fibre.
So far I've implemented fins with a dynamic count into a 3JS render

Education

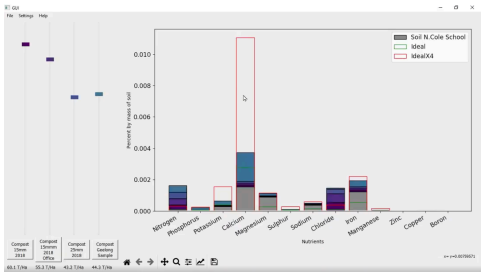
Ballarat Grammar - Academic scholarship, Specialist Math

The University of Melbourne - Major in Computing and Software Systems *Mid 2021 - Present*

Foundations of Computing (Python)	98%
Linear Algebra	81%
Algorithms and Data Structures (C)	76%
- implemented a QuadTree, Dijkstra and A*	(92% project average)
Foundations of Algorithms (C)	75%
- implemented DNA processing, matrix operations	(91% project average, 100% MST)
Object Oriented Software Development (Java)	76%
Engineering Modelling And Design	77%
Computer systems	Current

Studying and implementing networking and OS protocols
including schedulers, memory allocators and Remote procedure calls

Projects



Soil Nutrient calculator - Python (VCE Software development)
https://github.com/HenryRoutson/Soil-value-calculator

Architected GUI Software for a multi million dollar business
after 5 months of learning programming.
Performs Vector calculus to find the ideal compost
for a particular soil and crop combination.

AutoHeader - Rust programming (Extracurricular)
https://github.com/HenryRoutson/autoheader

```rust
fn setup(file_string: &str) {

    println!("\n\n\n\n Running  Setup \n  {} \n", file_string);

    if file_string.ends_with("~defs.h") { println!("~defs file already exists"); return; }

    // check if c or h file
    let ext = &file_string[file_string.len()-2..file_string.len()]; // extension
    if ext != ".c" && ext != ".h" { println!("file didn't contain a c or h extension"); return; }

    // check if defs file alreadys exists and setup is already done
    let file_string_no_ext = &file_string[0..file_string.len()-2];
    let defs_path = file_string_no_ext.to_string().add("~defs.h"); // remove extension and add
    let defs_path = Path::new(&defs_path);
    if defs_path.exists() {  println!("~defs file already exists"); return; }
```

Implemented "Public" keyword into the c programming language
to automate the creation of header files.
Uses automated testing and Regular Expressions.

CHelp - C (Extracurricular)
https://github.com/HenryRoutson/CHelp

Utilised C meta-programming to make dynamic memory in C easier.
Program displays data and location of un-freed mallocs.
From the README…

```
./tests/9_main


ERROR: wrong number of unfreed mallocs
        expected : 0
        found    : 1


        mallocs are listed below,
        in reverse allocation order
UNFREED       ---
file_name   : tests/9_main.c
line_number : 34
print_func  :
1 s2 3.000000
              ---

  FREED
```

+ Auto assert malloc and check if the size is negative or zero
+ Auto null after free
+ Assert all memory is freed at the end of a program running
+ Store messages for individual mallocs
+ Store print functions for each malloc, allowing generic debugging print functions
+ Runs in O(1) overhead
+ Uses Automated testing

Skills

Git SQL / Databases Data Structures and Algorithms Optimisation Linux