C bugs

#C #paper-note

Bugs

Dangling Pointer
Pointing to de-allocated memory. Can cause segmentation faults if the memory is outside of programs remit, or unexpected behaviour if it points to a random value in control of the program.

```c
int* initaliseInteger(int val) {
    int x = val;
    int* ptr = &x; // in stack memory
    return ptr; // dangling pointer
}
```

Double Free
A double free is when an address has `free()` called on it multiple times. This will lead to a segmentation fault.

Memory Leak
A memory leak is when memory is allocated but never de-allocated, leading to more and more memory held doing nothing.

Integer Overflow
An integer overflow is when an integer at the integer limit is added to, causing it to roll around to negative numbers. As integers are represented in a set number of binary digits, they have an upper limit, and once that limit is reached, if you add more, it will roll back around to the negative numbers.

Errors

Types of Error
syntax errors are when code does not follow the syntax of the language, semantic errors are when code leads to erroneous or unintended outputs.

Segmentation Fault
A segmentation fault is when the OS shuts down a program that tries to access memory outside of where it is permitted to

Stack Overflow
A Stack overflow is when the stack memory fills up past full. This is caused by a program having a very long call stack. Commonly stack overflows occur when there is excessive recursion.

Tools

There are many tools available to find bugs and errors in programs.

Static Analysis
Static analysis is any analysis of code done without executing it. This includes linters, compilers, type checking.

Dynamic Analysis
Dynamic analysis is when code is executed and its behaviours are examined. This can catch semantic errors.

Examples include:

- address sanitizer
  - memory sanitizer
    - detects uninitialized reads from memory such as those caused by double frees, or dereferencing uninitialized pointers.
  - leak sanitizer
    - checks for memory leaks.
  - undefined behaviour sanitizer
    - flags potentially unintended behaviours like integer overflow.

Debuggers
Debuggers are tools that run code step by step so it is easier to follow and determine where errors occur.