

# Debugging in C and C++

## GDB

- Compile with `-g` option
- Run `gdb <programme name>`
- Within `gdb`, `r` runs the programme
- `r arg1 arg2` runs the programme with arguments

## Detect Memory Leaks with ASan

ASan: Address Sanitizer. This is a memory error detector for C/C++.

It finds:

- Use after free (dangling pointer dereference)
- Heap buffer overflow
- Stack buffer overflow
- Global buffer overflow
- Use after return
- Use after scope
- Initialization order bugs
- Memory leaks

To compile, add the `-fsanitize=address` and `-g` options. Example Makefile for C++:

```
SOURCES:= $(wildcard *.cpp) $(wildcard *.hpp)
OBJECTS:= $(wildcard *.cpp)
OUT:= bin/main
main: $(SOURCES)
    g++ -W -Wall -std=c++17 -o $(OUT) $(OBJECTS)

debug: $(SOURCES)
    g++ -W -Wall -std=c++14 -g -fsanitize=address -o $(OUT) $(OBJECTS)
```