# Valgrind Orbstack Install macOS Recommended

macOS x86 don't need to do this as valgrind is supported on there architecture but it could be nice to do anyway.

- **Prerequisite**
  - Command Line Tools part of Xcode
  - Xcode
  - Brew optional

- **Install**

  - ## Note % terminal commands within macOS

  - ## Note $ terminal commands within the machine

    - https://orbstack.dev/download or % brew install orbstack
    - Go through setup process, we want to use it for linux systems, and to allow network drive access, it helps manage file on the machine
    - For new machine name it whatever you want and select what you want, I choose
      - Name: CS3100
      - Distribution: Debian Bookworm
      - Version: Bookworm
      - CPU Type: Apple
      - (Don't choose apple if you are on x86)
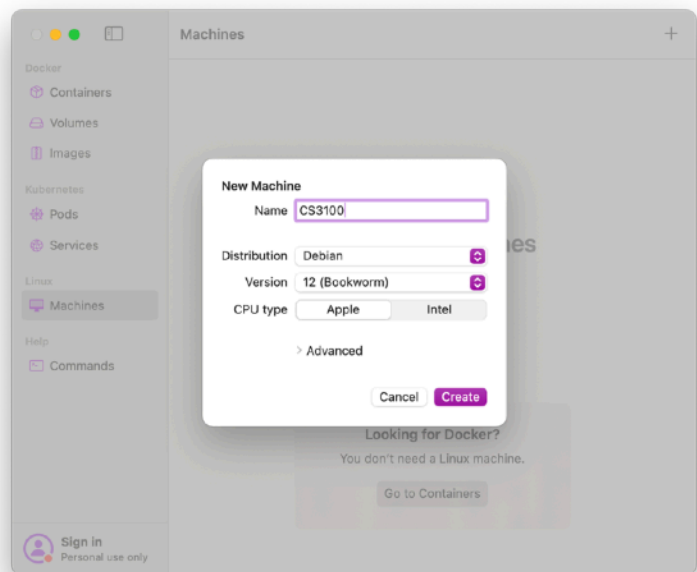
- Confirm that it is your default machine by right clicking and make default (not needed but convenient)
- Right click on the machine and open it in terminal
- We will need the following packages: gcc and valgrind
- Within the machine insert
  - $ sudo apt-get install gcc
  - $ sudo apt-get install valgrind

```
Last login: Fri Jan 19 09:46:58 on ttys009
/var/folders/1q/yvy8f7x573v6gq0wpf2grhx40000gn/T/orbstack-open-terminal_3A47E1BA
.sh ; exit;
coletonwatt@coletons-mbp ~ % /var/folders/1q/yvy8f7x573v6gq0wpf2grhx40000gn/T/or
bstack-open-terminal_3A47E1BA.sh ; exit;
coletonwatt@CS3100:/Users/coletonwatt$ sudo apt-get install gcc
```



```
coletonwatt@CS3100:/Users/coletonwatt$ sudo apt-get install valgrind
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  gdb libbabeltrace1 libboost-regex1.74.0 libc6-dbg libcurl3-gnutls
  libdebuginfod-common libdebuginfod1 libdw1 libglib2.0-0 libglib2.0-data
  libicu72 libpython3.11 libpython3.11-minimal libpython3.11-stdlib
  libreadline8 libsource-highlight-common libsource-highlight4v5 libsqlite3-0
  libxml2 media-types readline-common sensible-utils shared-mime-info ucf
  valgrind-dbg xdg-user-dirs
Suggested packages:
  gdb-doc gdbserver low-memory-monitor readline-doc valgrind-mpi kcachegrind
  alleyoop valkyrie
The following NEW packages will be installed:
  gdb libbabeltrace1 libboost-regex1.74.0 libc6-dbg libcurl3-gnutls
  libdebuginfod-common libdebuginfod1 libdw1 libglib2.0-0 libglib2.0-data
  libicu72 libpython3.11 libpython3.11-minimal libpython3.11-stdlib
  libreadline8 libsource-highlight-common libsource-highlight4v5 libsqlite3-0
  libxml2 media-types readline-common sensible-utils shared-mime-info ucf
  valgrind valgrind-dbg xdg-user-dirs
0 upgraded, 27 newly installed, 0 to remove and 0 not upgraded.
Need to get 68.1 MB of archives.
After this operation, 189 MB of additional disk space will be used.
```

- Download the runValgrind.sh file from canvas
  - https://weber.instructure.com/courses/581054/assignments/6158206?module_item_id=10298542
  - Or find it under Canvas files
- Move the .sh file to your class files within macOS and open that location in terminal
- Activate the machine at the location using and confirm files:
  - % orb
  - $ ls
- And give the sh file write access
  - $ chmod +x runValgrind.sh
- Now how compiling works we need to compile the c programs within the Linux I will be using the files we did in class
  - $ gcc -o week3p1 ./wk03_code/p1.c
  - $ ./week3p1
- Now test Valgrind
  - $ ./runValgrind.sh ./week3p1


```
CTestFile
Day 1 code.c
Notes.pages
Operating_Systems_-_Three_Easy_Pieces.pdf
book-riscv-rev1.pdf
macOSARMxv6.pages
macOSARMxv6.pdf
ostep-code
runValgrind.sh
wk03_code
xv6
xv6-public-master-2
xv6-riscv-book
coletonwatt@coletons-mbp CS 3100 % orb
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ ls
 Assignment1CommandLine.c    Operating_Systems_-_Three_Easy_Pieces.pdf
 Assignments                 ostep-code
 book-riscv-rev1.pdf         runValgrind.sh
 CTestFile                   wk03_code
'Day 1 code.c'               xv6
 macOSARMxv6.pages           xv6-public-master-2
 macOSARMxv6.pdf             xv6-riscv-book
 Notes.pages
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$
```



```
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ ./runValgrind.sh ./week3p1
==4673== Memcheck, a memory error detector
==4673== Copyright (C) 2002-2022, and GNU GPL'd, by Julian Seward et al.
==4673== Using Valgrind-3.19.0 and LibVEX; rerun with -h for copyright info
==4673== Command: ./week3p1
==4673==
hello world (pid:4673)
hello, I am parent of 4674 (pid:4673)
hello, I am child (pid:4674)
==4673==
==4673== FILE DESCRIPTORS: 3 open (3 std) at exit.
==4673==
==4673== HEAP SUMMARY:
==4673==     in use at exit: 0 bytes in 0 blocks
==4673==   total heap usage: 1 allocs, 1 frees, 1,024 bytes allocated
==4673==
==4673== All heap blocks were freed -- no leaks are possible
==4673==
==4673== For lists of detected and suppressed errors, rerun with: -s
==4673== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
==4674==
==4674== FILE DESCRIPTORS: 3 open (3 std) at exit.
==4674==
==4674== HEAP SUMMARY:
==4674==     in use at exit: 0 bytes in 0 blocks
==4674==   total heap usage: 1 allocs, 1 frees, 1,024 bytes allocated
==4674==
==4674== All heap blocks were freed -- no leaks are possible
==4674==
==4674== For lists of detected and suppressed errors, rerun with: -s
==4674== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$
```



```
runValgrind.sh
wk03_code
xv6
xv6-public-master-2
xv6-riscv-book
coletonwatt@coletons-mbp CS 3100 % orb
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ ls
 Assignment1CommandLine.c    Operating_Systems_-_Three_Easy_Pieces.pdf
 Assignments                 ostep-code
 book-riscv-rev1.pdf         runValgrind.sh
 CTestFile                   wk03_code
'Day 1 code.c'               xv6
 macOSARMxv6.pages           xv6-public-master-2
 macOSARMxv6.pdf             xv6-riscv-book
 Notes.pages
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ chmod +x runValgrind.sh
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ gcc -o week3p1 ./wk03_code/p1.c
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ week3p1
-bash: week3p1: command not found
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$ ./week3p1
hello world (pid:4670)
hello, I am parent of 4671 (pid:4670)
hello, I am child (pid:4671)
coletonwatt@TheRealCS3100:/Users/coletonwatt/Developer/CS 3100$
```

You have now finished and can run valgrind congrats!

- In the future all you need to do is open location terminal
  - % orb
  - $ gcc [flags] [compiled-file] [program-file]
  - $ ./runValgrind.sh [compiled-file]

If you need help you can reach out to me in class or through Discord (eleninja102)

Option 2 Install a Virtual Machine

I won't go through this as there are things online. I personally use VMware Fusion and UTM. Both are free VMware as student and UTM online. As a distribution I use Debian ARM. If you need help with this you can reach out.