# Assignment -1 Vijaya Krishna Sameeraj Jonnavithula Date : September 8, 2024

1. Briefly research and summarize what gem5 is, including its uses in computer architecture research and the types of simulations it supports. Provide a 1-2 paragraph overview in your own words.

Gem5, a computer architecture simulator with great flexibility and extensibility, is extensively utilised in research projects in academia and industry. It makes it possible for researchers to build intricate models of computer systems, which facilitates the investigation of a variety of architectural elements, from straightforward processors to intricate multi-core systems. Gem5 is unique in that it can simulate at several abstraction levels, such as functional simulations for verifying correctness and cycle-accurate simulations for in-depth performance analysis.

The simulator enables users to model and experiment with different processor designs and configurations because it supports a variety of instruction set architectures (ISAs), including ARM, x86, RISC-V, and MIPS. Gem5 is a vital tool for academics studying CPU architecture, memory hierarchy, cache coherence protocols, and network-on-chip (NoC) architectures because of its adaptability. Additionally, gem5 enables full-system simulation, which mimics the operation of a whole operating system on a processor and sheds light on the interactions between software and hardware. This expertise is essential for assessing how architectural modifications affect performance in practical situations.

Gem5 is an effective instrument for furthering computer architecture research overall, allowing for the thorough investigation and improvement of both new and old computing systems.

1. List the software dependencies needed to build gem5 (e.g., Python, GCC, SCons).

To build gem5, several software dependencies are required. Here's a list of the key dependencies:

Python: gem5 requires Python for scripting and as part of its build system. Python 3.6 or later is recommended.

SCons: SCons is the build system used by gem5. It is a software construction tool written in Python, and it's necessary for managing the build process.

GCC (GNU Compiler Collection): A C++ compiler is required to build gem5. GCC is commonly used, and GCC 7.5 or later is recommended for compatibility.

zlib: This is required for compression support within gem5, especially for handling compressed trace files.

libprotobuf-dev and protobuf-compiler: Protobuf is used for protocol buffer support in gem5. These libraries are needed for certain configurations and features.

libgoogle-perftools-dev: Optional but recommended for performance profiling using Google Performance Tools (gperftools).

SWIG (Simplified Wrapper and Interface Generator): Required if you are using the Python scripting interface, especially for integrating custom modules written in C++.

pkg-config: A helper tool used to determine the compile and linker flags that should be used when building software that depends on libraries.

libsqlite3-dev: Necessary for database support, particularly when using statistics output to SQLite databases.

libboost-all-dev: Boost libraries are used in gem5 for various utilities and features, particularly in the Python and C++ interface.

m4: A macro processor required for building some of the tools associated with gem5.

libffi-dev: Required for handling Foreign Function Interface in some configurations.

These dependencies are typically available through standard package managers on Linux distributions (e.g., apt, yum), and the specific versions required may vary based on the gem5 version you are building. Additionally, some optional dependencies may be required based on specific features or configurations you wish to use within gem5.

* 1. Provide instructions on how to install these dependencies on your operating system (Linux, macOS, or Windows with WSL).

Here are instructions for installing the dependencies required to build gem5 on Linux, macOS, and Windows with WSL.

Linux (Ubuntu/Debian-based)

Update your package list:

sudo apt-get update

Install the dependencies:

sudo apt-get install -y build-essential python3 python3-dev python3-pip scons \

gcc g++ zlib1g-dev libprotobuf-dev protobuf-compiler \

libgoogle-perftools-dev swig pkg-config libsqlite3-dev \

libboost-all-dev m4 libffi-dev

macOS

Install Homebrew (if not already installed):

/bin/ -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

Install the dependencies:

brew install scons gcc python3 protobuf google-perftools swig \

pkg-config sqlite boost m4 libffi

Ensure the correct version of GCC is used (Homebrew installs GCC as gcc-<version>):

export CC=/usr/local/bin/gcc-<version>

export CXX=/usr/local/bin/g++-<version>

Windows with WSL (Windows Subsystem for Linux)

Install WSL (if not already installed):

Open PowerShell as Administrator and run:

powershell

wsl --install

Restart your computer if required.

Install a Linux distribution (e.g., Ubuntu):

Install Ubuntu from the Microsoft Store or via PowerShell:

powershell

wsl --install -d Ubuntu

Update your package list in WSL:

sudo apt-get update

Install the dependencies in WSL:

sudo apt-get install -y build-essential python3 python3-dev python3-pip scons \

gcc g++ zlib1g-dev libprotobuf-dev protobuf-compiler \

libgoogle-perftools-dev swig pkg-config libsqlite3-dev \

libboost-all-dev m4 libffi-dev

Python Package Installation: For some configurations, you may need additional Python packages. You can install them using pip3:

pip3 install <package\_name>

Environment Setup: Ensure your environment variables are correctly set, especially for compilers, by adding them to your shell configuration file (e.g., ~/.rc, ~/.zshrc):

export CC=gcc  
export CXX=g++

Once these dependencies are installed, you can proceed to clone the gem5 repository and build it according to your specific needs.

* 1. Clone the gem5 repository from [gem5's official GitHub page](https://github.com/gem5/gem5). Include the command used to clone the repository.

The command used is : git clone <https://github.com/gem5/gem5.git>

A computer screen shot of a computer program

Description automatically generated

* 1. Navigate to the gem5 directory and list the available build options (e.g., different CPU models, cache configurations).

Types of CPU

A screen shot of a computer screen

Description automatically generated

Cache Configurations:

A screenshot of a computer screen

Description automatically generated

A screen shot of a computer

Description automatically generated

1. Build :

Navigated to the gem5 directory where all the requirements are installed and ran the following commands :   
  
scons build/X86/gem5.opt

A screenshot of a computer program

Description automatically generated

The build is complete after I increased the core using the code : scons build/X86/gem5.opt -j4

A screen shot of a computer program

Description automatically generated

To view the build, I changed the directory to the build directory, and you can see the .build and .opt files which confirm the build.

Also did the same for ARM CPU :

scons build/ARM/gem5.opt -j1  
  
A screen shot of a computer program

Description automatically generated

The ARM build was also successful as you can see from the build files in the list.

A computer screen with text on it

Description automatically generated

3.1 : Issues   
  
1. I was unable to get the build command to work

A screenshot of a computer program

Description automatically generated

Troubleshooting :   
  
1. Had to check on how many dependencies were installed, realized that I was missing them.   
  
Started installing all the dependencies:   
  
A screenshot of a computer program

Description automatically generated

Code used :   
  
wsl –install

wsl --install -d Ubuntu  
  
sudo apt-get update

sudo apt-get install -y build-essential python3 python3-dev python3-pip scons \

gcc g++ zlib1g-dev libprotobuf-dev protobuf-compiler \

libgoogle-perftools-dev swig pkg-config libsqlite3-dev \

libboost-all-dev m4 libffi-dev

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Then after installing, rebooting the system to see the changes in effect and navigating to the directory , I faced another issue of missing Python requirements  
  
A screenshot of a computer

Description automatically generated

Installed them too:   
  
A screenshot of a computer program

Description automatically generated

Understood how important the pre-requisites are and how many different ways we can build an environment on gem5.

After waiting for over 2 hours the compilation was terminated :   
  
A screenshot of a computer

Description automatically generated

To troubleshoot this, I went ahead and reinstalled and updated every directory and library. After looking into the process further, just realized that I did not restart the system and so I restarted the system and tried again.   
  
This worked but I had an other problem of the build running out of the memory :   
  
A screen shot of a computer program

Description automatically generated  
  
Ran the following commands to extend ram and memory for the build .   
  
sudo fallocate -l 4G /swapfile  
sudo chmod 600 /swapfile  
sudo mkswap /swapfile  
echo '/swapfile none swap sw 0 0' | sudo tee -a /etc/fstab  
  
A screen shot of a computer

Description automatically generated

A black screen with white text

Description automatically generated  
  
After this the memory allocation was successful and ran without any issues.

1. Any additional observations or comments on using gem5.

* The gem5 takes a lot of time to build the CPU on laptops.
* It takes anywhere from a few minutes to 45 minutes which is a lot of time to build.
* I understand we are not using high end servers, but if each time I have to build something I have to spend an hour on it, it just takes away the whole purpose of being quick and usable.
* On the other hand , it was fun working with gem5, the website is great and provides most of the information you need to get started and it lets you get your hands dirty as we are working from scratch.
* The experience of building something new is always fascinating and it also makes you understand why you need certain files and directories for it to work properly.
* While waiting for the build to complete , I did a little research on gem5 and found out that it is actually a project at at the University of Michigan as the m5 project and at the University of Wisconsin as the GEMS project.

REFERENCES:

<https://www.gem5.org/getting_started/>

<https://www.gem5.org/documentation/learning_gem5/part1/building/>

<https://www.gem5.org/documentation/general_docs/building>