Carleton UNIVERSITY
Canada's Capital University

CADMIUM –SIMULATOR

1. System requirements:

Ubuntu 16.04 (it may also work in previous versions). It will work in later versions. You can download Ubuntu from the following link: https://www.ubuntu.com/download/desktop

RAM 16GB (you will be able to run small models with 4GB ram)

Ubuntu Installation: (If you have Windows, option B is recommended)

Option A: Install Ubuntu along with Windows on a Windows machine

Follow these instructions:

https://www.tecmint.com/install-ubuntu-16-04-alongside-with-windows-10-or-8-in-dual-boot/

Option B: Install Ubuntu on a Virtual Box on a Windows computer:

1. Download and install Virtual Box. You can follow the instructions in this link:

https://www.wikihow.com/Install-VirtualBox

2. Follow the instructions in this link to install Ubuntu in Virtual Box:

https://medium.com/@tushar0618/install-ubuntu-16-04-lts-on-virtual-box-desktop-version-30dc6f1958d0

2. Dependencies:

Boost library for c++

- To install Boost:
    - Open Ubuntu terminal. To open Ubuntu terminal press: "CONTROL+ALT+t"
    - Type on the Ubuntu terminal the following command and press ENTER:
      sudo apt-get install libboost-all-dev
    - Type the administrative password and press ENTER
    - When you are asked "Do you want to continue?" type: *y* and then press ENTER
    - Wait until the installation is finished

Cadmium

- It is an external library. It is a header only simulator. It does not need any installation.
- You can download Cadmium from the following link:
  https://github.com/SimulationEverywhere/cadmium
- We explain how to include this library thorough a simple model example in section 4.

Documentation written by Cristina Ruiz Martin. If you have any question, you can email:
cristinaruizmartin@sce.carleton.ca

DESTimes

- It is an external library. It is a time class for Cadmium simulator. It is a header only library. It does not need any installation.
- You can download DESTimes from the following link: https://github.com/Laouen/DESTimes.git
- We explain how to include this library thorough a simple model example in section 4.

3. Compiler

The simulator is tested using g++7.2 compiler. Previous versions of g++ do not work because they cannot compile in c++17.

Instructions to install gcc 7.2 and g++ 7.2 and make them as default compiler:

(1) Open Ubuntu terminal. To open Ubuntu terminal press: "CONTROL+ALT+t"

(2) Type on the Ubuntu terminal the following command and press ENTER:
sudo add-apt-repository ppa:jonathonf/gcc-7.1

(3) Type the administrative password (if you are asked. The system may remember it)

(4) When you are asked "Press [ENTER] to continue or ctrl-c to cancel adding it" press ENTER

(5) Type on the Ubuntu terminal the following command and press ENTER:
sudo apt-get update

(6) Type the administrative password (if you are asked. The system may remember it)

(7) Type on the Ubuntu terminal the following command and press ENTER:
sudo apt-get install gcc-7 g++-7

(8) Type the administrative password (if you are asked. The system may remember it)

(9) When you are asked "Do you want to continue?" type: *y* and press ENTER

(10) Type on the Ubuntu terminal the following command and press ENTER:
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-7 60 --slave /usr/bin/gcc-ar gcc-ar /usr/bin/gcc-ar-7 --slave /usr/bin/gcc-nm gcc-nm /usr/bin/gcc-nm-7 --slave /usr/bin/gcc-ranlib gcc-ranlib /usr/bin/gcc-ranlib-7

(11) Type the administrative password (if you are asked. The system may remember it)

(12) Type on the Ubuntu terminal the following command and press ENTER:
sudo update-alternatives --install /usr/bin/g++ g++ /usr/bin/g++-7 60 --slave /usr/bin/g++-ar g++-ar /usr/bin/g++-ar-7 --slave /usr/bin/g++-nm g++-nm /usr/bin/g++-nm-7 --slave /usr/bin/g++-ranlib g++-ranlib /usr/bin/g++-ranlib-7

(13) Type the administrative password (if you are asked. The system may remember it)

(14) To check that you have g++-7 type on the terminal the following command and press ENTER:
g++ -v

Documentation written by Cristina Ruiz Martin. If you have any question, you can email:
cristinaruizmartin@sce.carleton.ca

4. Example

The Alternate bit protocol Simulator is provided as an example. To run the test and the simulator, follow the instructions on the README.txt that accompanies the model.

The time class in the vendor folder was obtained from the https://github.com/Laouen/DESTimes.git.

Documentation written by Cristina Ruiz Martin. If you have any question, you can email: cristinaruizmartin@sce.carleton.ca