

LAPENNA Program

Q&A 1

Kwai Wong, Stan Tomov
Rocco Febbo, Julian Haloy, Stephen Qiu

University of Tennessee, Knoxville

Sept. 8, 2021

Acknowledgements:

- Support from NSF, UTK, JICS, ICL, NICS
- LAPENNA, www.jics.utk.edu/lapenna, NSF award #202409
- www.icl.utk.edu, cfdlab.utk.edu, www.xsede.org,
www.jics.utk.edu/recsem-reu,
- MagmaDNN is a project grown out from the RECSEM REU Summer program supported under NSF award #1659502
- Source code: www.bitbucket.org/icl/magmadnn
- www.bitbucket.org/cfdl/opendnnwheel



INNOVATIVE
COMPUTING LABORATORY

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

JICS
Joint Institute for
Computational Sciences
ORNL
Computational Sciences

OAK RIDGE
National Laboratory

Exercise

Computing Platforms

“ To learn data science is to do data science”

Google Colab, jupyter notebook
python, R tutorials
upload file to /content
mount your google drive

<https://colab.research.google.com/notebooks/intro.ipynb>

<https://www.youtube.com/watch?v=inN8seMm7UI>

<https://github.com/dataprofessor>

<https://www.youtube.com/watch?v=huAWa0bqxtA>

<https://www.youtube.com/watch?v=Ri1MfaSISW0>

Use this link to run R code on Google COLAB

<https://colab.research.google.com/notebook#create=true&language=r>

✓ Google drive share space, google colab, colab.research.google.com

Shared with me > LAPENNA material

Name ↑	Owner
References https://drive.google.com/drive/folders/0ANAJ4JCNrBlbUk9PVA	UT_LAPENNA-FALL-2021 > WORKSHOP-JULY-2021
Search in Drive	Now you can block people in Drive To prevent people outside
UT_LAPENNA-FALL-2021 1 person	Now you can block people in Drive To prevent people outside
Suggested	Now you can block people in Drive To prevent people outside
 Copy of Python Tutorial.ipynb You modified today	 Copy of Linear Algebra in P... You modified today
LECT-0-INTRO	2021-LAPENNA-U0-Introduction.pdf
WORKSHOP-JULY-2021	2021-LAPENNA-U1-Ecosystem-R1.pdf 2021-LAPENNA-U2-LA-R1.pdf 2021-LAPENNA-U3-StatisticalML-R1.pdf 2021-LAPENNA-U4-DNN-R1.pdf 2021-LAPENNA-U5-MLP-R1.pdf 2021-LAPENNA-U6-CNN1-R1.pdf 2021-LAPENNA-U7-CNN2-R1.pdf 2021-LAPENNA-U8-CNN3-Backcal-R1.pdf 2021-LAPENNA-U9-Optimization-R1.pdf 2021-LAPENNA-U10-Unet-R1.pdf 2021-LAPENNA-U11-GPU-R1.pdf

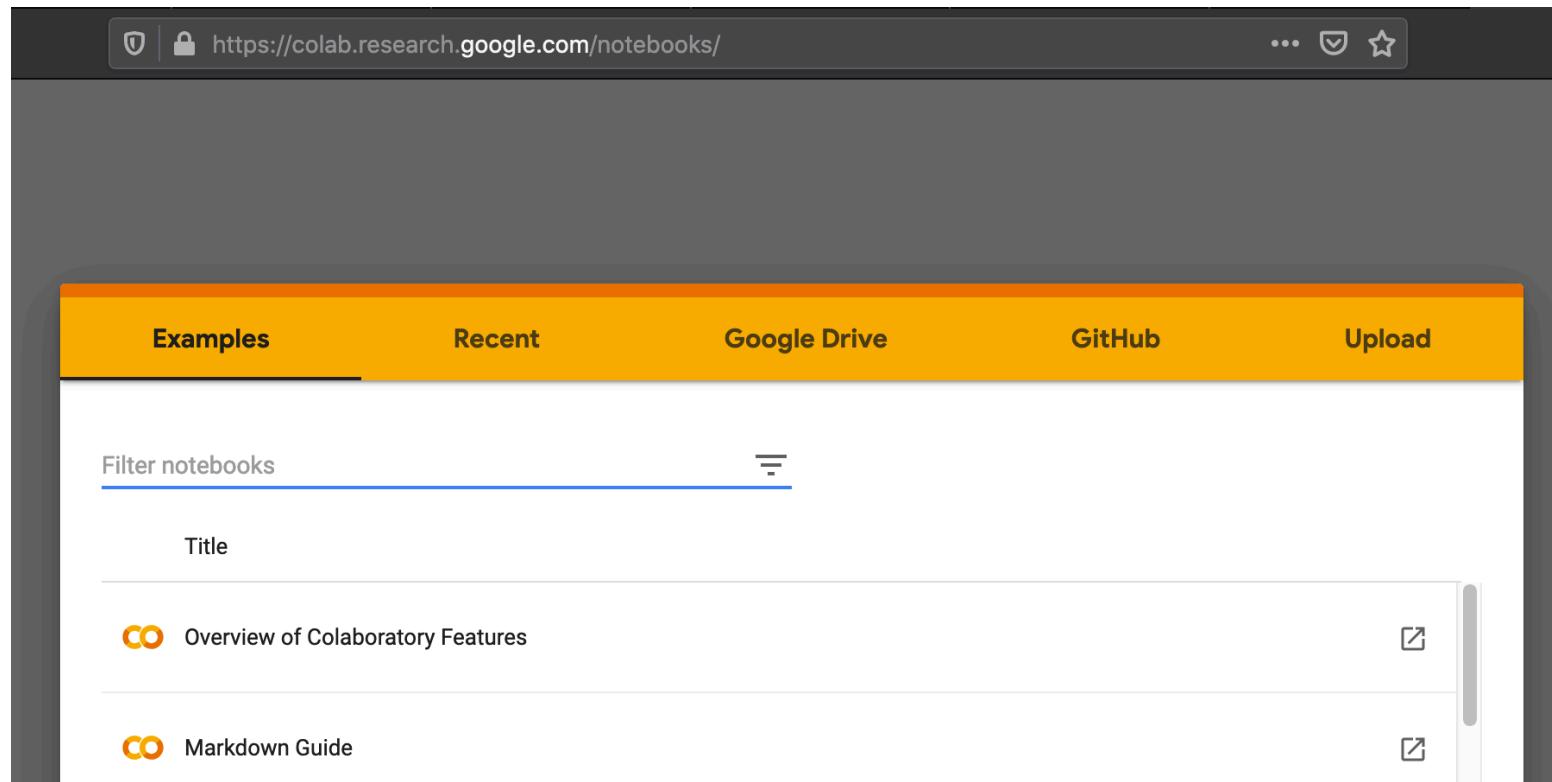
Use this link to run pure native R code on Google COLAB

<https://colab.research.google.com/notebook#create=true&language=r>

To run R code on Google COLAB in conjunction with python code, simply use the colab code notebook : (create a new notebook)

<https://colab.research.google.com/notebooks/>

<https://colab.research.google.com/notebook#create=true&language=python>



Upload file to temporary colab drive

Location (path) : /content

Untitled4.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

Play button

Untitled4.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Play button

Untitled4.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Play button

Untitled4.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Files

sample_data

Play button

```
!ls -l  
!pwd
```

total 11272
-rw-r--r-- 1 root root 49000 Feb 2 05:28 DataTrnLbts.txt
-rw-r--r-- 1 root root 193000 Feb 2 05:27 DataTrn.txt
-rw-r--r-- 1 root root 11289600 Feb 2 05:29 digits1234.txt
drwxr-xr-x 1 root root 4096 Jan 20 17:27 sample_data

Favorites

- 2020quizcode
- cart.R
- cart2.R
- code.demon...tion.inclass
- codes.Rproj
- datasets.R
- DataTrn.txt
- DataTrnLbts.txt
- digits1234.txt
- leastsquare.reg.cls.2009
- Linear.Classification
- Linear.Regession
- mds.R
- nonparametrics
- testerror.R

Locations

- Network

Media

- Music
- Photos
- Movies



```
%load_ext rpy2.ipython
```

```
/usr/local/lib/python3.6/dist-packages/rpy2/robjects/pandas2ri.py:14: FutureWarning: pandas.core
    from pandas.core.index import Index as PandasIndex
/usr/local/lib/python3.6/dist-packages/rpy2/robjects/pandas2ri.py:34: UserWarning: pandas >= 1.0
    warnings.warn('pandas >= 1.0 is not supported.')
```

- ✓ Upload datasets.R to /content of colab drive, double click to view.
- ✓ Use Edit of the webpage (not colab) to do cut and paste to the new cell
- ✓ To run a code cell using R, started with “%%R”

Files

- ..
- sample_data
- DataTrn.txt
- DataTrnLbels.txt
- datasets.R
- digits1234.txt

+ Code + Text

```
[5]   from pandas.core.index import Index as PandasIndex
/usr/local/lib/python3.6/dist-packages/rpy2/robjects/pandas2ri.py:14: FutureWarning: pandas.core
    from pandas.core.index import Index as PandasIndex
/usr/local/lib/python3.6/dist-packages/rpy2/robjects/pandas2ri.py:34: UserWarning: pandas >= 1.0
    warnings.warn('pandas >= 1.0 is not supported.')
```

%%R

```
###read digits data
n<-200;
imgdata<-scan(file="digits1234.txt")
imgdata<-matrix(imgdata,nrow=n*4,byrow=T) #now imagedata
###converting from integer representation to one of K rep
class<-c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
labels<-matrix(0,nrow=n*4,ncol=4)
labels[matrix(c(1:(n*4),class),nrow=n*4)]<-1
```

RAM Disk Editing

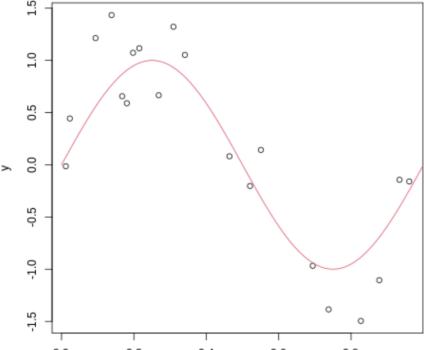
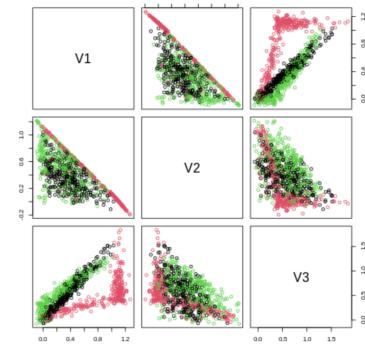
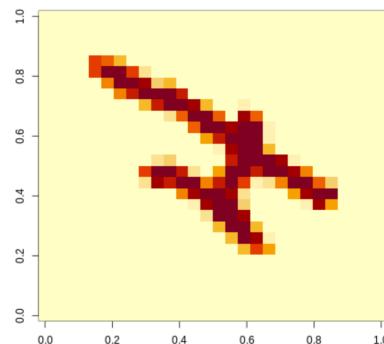
datasets.R X

```
1 ###read digits data
2 n<-200;
3 imgdata<-scan(file="digits1234.txt")
4 imgdata<-matrix(imgdata,nrow=n*4,byrow=T) #now imagedata
5 ###converting from integer representation to one of K rep
6 class<-c(rep(1,n),rep(2,n),rep(3,n),rep(4,n))
7 labels<-matrix(0,nrow=n*4,ncol=4)
8 labels[matrix(c(1:(n*4),class),nrow=n*4)]<-1
9 ###show image number 602
10 im<-matrix(imgdata[602,],nrow=28,ncol=28)
11 image(t(im))
12
```

Modify the file location:

“file=/content/digit1234.txt”,
“file=“/content/dataTrn.txt”,
“file=“/content/dataTrnLblas.txt”

Then click run



Step 1 To connect Google Drive (GDrive) with Colab, execute the following two lines of code in a cell of Colab:

```
from google.colab import drive  
drive.mount("/content/gdrive")
```

<https://towardsdatascience.com/google-colab-import-and-export-datasets-eccf801e2971>

Running the shell will return a URL link and ask for an authorization code:

```
1 from google.colab import drive  
2 drive.mount("/content/gdrive")
```

Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=420431011111-6jre80y&redirect_uri=https%3A%2F%2Faccounts.google.com%2Foauth2%2Fcallback&response_type=code&scope=https://www.googleapis.com/auth/drive.readonly

Enter your authorization code:

Step 2 Follow to the mentioned link, sign in Google account, and copy the authorization code by clicking at highlighted spot:

Step 3 Paste the authorization code in the shell and finally, Google Drive will be mounted at `/content/gdrive`. Note that, files in the drive are under the folder `/content/gdrive/My Drive/`. Now, we can import files in GDrive using a library like Pandas.

```
1 from google.colab import drive  
2 drive.mount("/content/gdrive")
```

Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=420431011111-6jre80y&redirect_uri=https%3A%2F%2Faccounts.google.com%2Foauth2%2Fcallback&response_type=code&scope=https://www.googleapis.com/auth/drive.readonly

Enter your authorization code:
.....

Mounted at /content/gdrive

Google

Sign in

Please copy this code, switch to your application and paste it there:

4/2QFkz_71FyTx6RmvhvFrIPHcXeKABEWsrUPABE98mk1KUveK
jREx80Y 

Step 4 For instance, we have a dataset (`sample.csv`) in `/My Drive/sample data` folder in GDrive.

import pandas as pd

```
pd.read_csv('/content/gdrive/My  
Drive/sample data/sample.csv')
```

- ✓ XSEDE computer accounts, www.xsede.org/ → user portal
- ✓ Documentation → MFA , Duo authentication, setting up duo on phone or pad



XSEDE|USER PORTAL

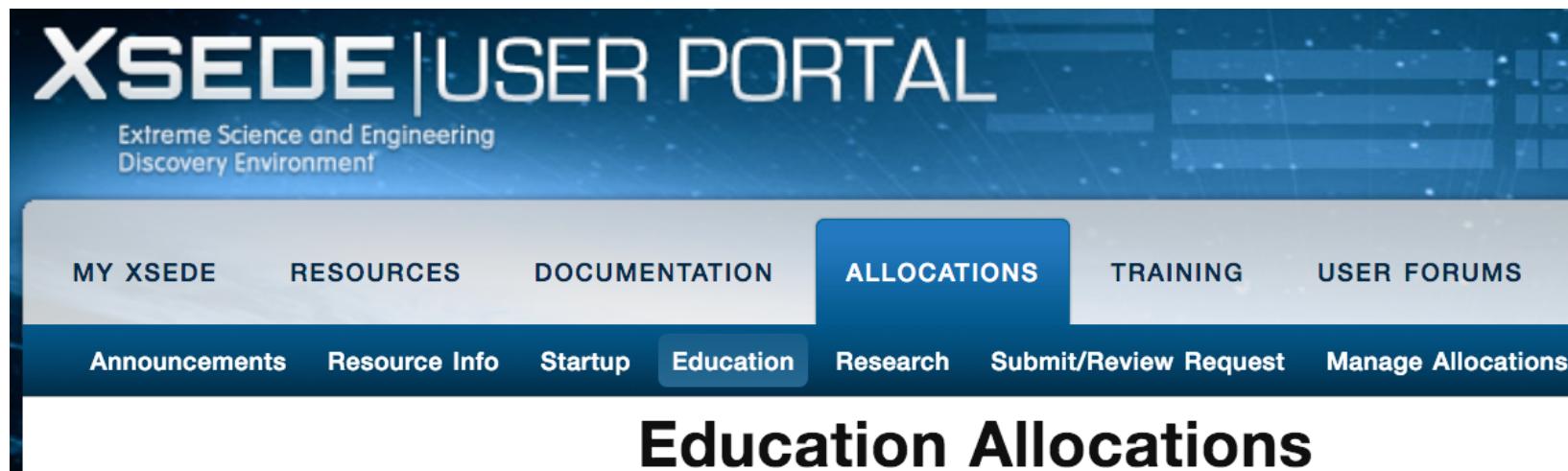
Extreme Science and Engineering Discovery Environment

MY XSEDE RESOURCES DOCUMENTATION ALLOCATIONS TRAINING USER FORUMS

Get Started Data Analysis Data Management User Guides Community Codes News Usage Policy MFA

XSEDE MFA with Duo

- ✓ Educational account for courses, startup allocation submission
- ✓ User code, login, general operations



XSEDE|USER PORTAL

Extreme Science and Engineering Discovery Environment

MY XSEDE RESOURCES DOCUMENTATION ALLOCATIONS TRAINING USER FORUMS

Announcements Resource Info Startup Education Research Submit/Review Request Manage Allocations

Education Allocations

✓ **XSEDE user code, login, general operations**

Shared with me > LAPENNA material > References > XSEDE ▾ 

Name ↑

Owner



Utilizing XSEDE Resources 

Rocco Febbo

Ensure you have an XSEDE account and have setup 2 factor authentication.

Obtain access to an SSH terminal. If you're on Windows you can use MobAXterm.
(<https://mobaxterm.mobatek.net/>)

Type:

ssh login.xsede.org

Logging In

Enter your password and select 2 factor authentication option. (either phone Duo app)
Once logged in, you will be greeted by the Single Sign on Hub message.

\$> gsish bridges2

BRIDGES-2 Important dates

Research allocation submission dates	March 15 - April 15
Early User Program	Jan. 12 - Feb. 9
Production operations begin	First quarter 2021

Core Concepts

- Converged HPC + AI + Data
- Custom topology optimized for data-centric HPC, AI and HPDA
- Heterogeneous node types for different aspects of workflows
- CPUs and AI-targeted GPUS
- 3 tiers or per-node RAM: 256GB, 512GB, 4TB
- Extremely flexible software environment
- Community data collections & Big Data as a Service

Bridges-2 will be available at no cost for research and education, and at cost-recovery rates for other purposes.

488 RM nodes will have 256GB of RAM, and 16 will have 512GB of RAM. All RM nodes will have:

- ✓ Two AMD EPYC 7742 CPUS, each with:
 - ✓ 64 cores, 128 threads
 - ✓ 2.25-3.40GHz
 - ✓ 256MB L3
 - ✓ 8 memory channels
 - ✓ NVMe SSD (3.84TB)

24 GPU nodes

- ✓ Eight NVIDIA Tesla V100-32GB SXM2 GPUs
- ✓ Two Intel Xeon Gold 6248 “Cascade Lake” CPUs: 20 cores, 40 threads, 2.50–3.90GHz, 27.5MB LLC, 6 memory channels
- ✓ 512GB of RAM: DDR4-2933
- ✓ 7.68TB NVMe SSD

✓ [colab.research.google.com,](https://colab.research.google.com)

The screenshot shows the Google Colab interface. At the top, there are two cards: "Google Colab" and "TensorFlow with GPU". Below them is a large button labeled "Open with Google Colaboratory". The main area displays the "TensorFlow with GPU" notebook. The notebook has a table of contents on the left with sections like "Tensorflow with GPU", "Enabling and testing the GPU", and "Observe TensorFlow speedup on GPU relative to CPU". The "Tensorflow with GPU" section contains code for checking GPU availability and performing a convolution operation. The "Enabling and testing the GPU" section provides instructions for enabling GPUs and includes a code cell showing the connection to a GPU. The "Observe TensorFlow speedup on GPU relative to CPU" section includes code for comparing CPU and GPU execution times.

```
#tensorflow_version 2.x
import tensorflow as tf
import timeit

device_name = tf.test.gpu_device_name()
if device_name != '/device:GPU:0':
    raise SystemError('GPU device not found')

print(
    '\n\nThis error most likely means that this notebook is not '
    'configured to use a GPU. Change this in Notebook Settings via the '
    'command palette (cmd/ctrl-shift-P) or the Edit menu.\n\n'
)
raise SystemError('GPU device not found')

def cpu():
    with tf.device('/cpu:0'):
        random_image_cpu = tf.random.normal((100, 100, 100, 3))
        net_cpu = tf.keras.layers.Conv2D(32, 7)(random_image_cpu)
    return tf.math.reduce_sum(net_cpu)

def gpu():
    with tf.device('/device:GPU:0'):
        random_image_gpu = tf.random.normal((100, 100, 100, 3))
        net_gpu = tf.keras.layers.Conv2D(32, 7)(random_image_gpu)
    return tf.math.reduce_sum(net_gpu)

# We run each op once to warm up; see: https://stackoverflow.com/a/45067900
cpu()
gpu()

# Run the op several times.
print('Time (s) to convolve 32x7x7x3 filter over random 100x100x100x3 images '
      '(batch x height x width x channel). Sum of ten runs.')
print('CPU (s):')
cpu_time = timeit.timeit('cpu()', number=10, setup="from __main__ import cpu")
print(cpu_time)
print('GPU (s):')
gpu_time = timeit.timeit('gpu()', number=10, setup="from __main__ import gpu")
print(gpu_time)
print('GPU speedup over CPU: {}'.format(int(cpu_time/gpu_time)))

[User]
Time (s) to convolve 32x7x7x3 filter over random 100x100x100x3 images (batch x height x width x channel). Sum of ten runs.
CPU (s):
3.862475891000031
GPU (s):
0.10837535100017703
GPU speedup over CPU: 35x
```

- ✓ COLAB Introduction <http://www.youtube.com/watch?v=vVe648dJOdl>
- ✓ Google COLAB : <https://www.youtube.com/watch?v=inN8seMm7UI>

```
kwong@KWONGs-MBP ~ % ssh login.xsede.org -l kwong
Please login to this system using your XSEDE username and password:
password:
Duo two-factor login for kwong
```

Enter a passcode or select one of the following options:

Last login: Wed Jul 7 15:12:33 2021 from 216.96.182.239

```
# Welcome to the XSEDE Single Sign-On (SSO) Hub!
#
# This system is for use by authorized users only, and is subject to the XSEDE
# Acceptable Use Policy, described at https://www.xsede.org/usage-policies.
# All activities on this system may be monitored and logged.
#
# Your storage on this system is limited to 100MB. Backup is not provided.
#
# From this system, you may login to other XSEDE system login hosts on which
# you currently have an active account. To see a list of your accounts, visit:
# https://portal.xsede.org/group/xup/accounts
#
# To login to an XSEDE system login host, enter: gsish <login-host>
# where <login-host> is the hostname, alias or IP address of the login host.
# The following default gsish host aliases have been defined:
#
#      bridges  bridges2  comet  comet-gpu  darwin
#      expanse  kyric    osg     rmacc-summit  stampede2
#
# For example, to login to the Comet system at SDSC, enter: gsish comet
#
# E-mail help@xsede.org if you require assistance in the use of this system.
```

```
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[kwong@ssohub ~]$ gsish bridges2
***** W A R N I N G *****
```

You have connected to br014.ib.bridges2.psc.edu, a login node of Bridges 2

```
kwong@KWONGs-MBP ~ % ssh login.xsede.org -l kwong
```

```
[kwong@ssohub ~]$ gsissh bridges2
```

```
[wongk@bridges2-login014 ~]$ interact --gpu
A command prompt will appear when your session begins
"Ctrl+d" or "exit" will end your session

--partition=GPU-small,GPU-shared --gpus=v100:1
salloc -J Interact --partition=GPU-small,GPU-shared --gpus=v100:1
salloc: Pending job allocation 2516669
salloc: job 2516669 queued and waiting for resources
salloc: job 2516669 has been allocated resources
salloc: Granted job allocation 2516669
salloc: Waiting for resource configuration
salloc: Nodes v002 are ready for job
[wongk@v002 ~]$ nvidia-smi
```

```
Mon Jul 26 22:31:18 2021
```

```
+-----+
| NVIDIA-SMI 460.73.01  Driver Version: 460.73.01  CUDA Version: 11.2  |
+-----+
| GPU Name      Persistence-M| Bus-Id     Disp.A | Volatile Uncorr. ECC | |
| Fan  Temp  Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|          |          |          | MIG M.   |
+-----+
| 0  Tesla V100-SXM2... On | 00000000:3A:00.0 Off |          0 | |
| N/A  29C   P0  39W / 300W |  0MiB / 32510MiB |  0%     Default |
|          |          |          | N/A |
+-----+
```

```
+-----+
| Processes:                               |
| GPU  GI  CI      PID  Type  Process name     GPU Memory |
| ID   ID              ID           Usage  |
+-----+
| No running processes found               |
+-----+
```

File Space

File Spaces

There are several distinct file spaces available on Bridges-2, each serving a different function.

- \$HOME, your home directory on Bridges-2, 25GB, /jet/home/username
- \$PROJECT, persistent file storage on Ocean. \$PROJECT is a larger space than \$HOME. The path of your Ocean home directory is /ocean/projects/groupname/username, where *groupname* is the Unix group id associated with your grant. Use the id command to find your group name.
- \$LOCAL, Node-local scratch storage on the node running a job
- \$RAMDISK, Scratch storage in the local memory associated with a running job. You can also use the memory allocated for your job for IO rather than using disk space. In a running job, the environment variable \$RAMDISK will refer to the memory associated with the nodes in use.

Transfer File

Bridges2: scp

```
$ scp -P 2222 /path/to/local/file username@data.bridges2.psc.edu:/path/on/bridges
```

The username is your PSC username

Home path: /jet/home/username/

1. Login to <https://ondemand.bridges2.psc.edu>
2. Use upload button to move a file from local machine to bridges2

Expanse: scp

```
$ scp /path/to/local/file username@login.expanse.sdsc.edu:/path/on/bridges
```

The username is your PSC username, passwd is same as xsede passwd

Home path: /home/username/

Use Open OnDemand to login

1. Login to <https://ondemand.bridges2.psc.edu>

The screenshot shows the Bridges2 OnDemand interface. At the top, there's a navigation bar with links for Bridges2 OnDemand, Files, Jobs, Clusters, Interactive Apps, and My Interactive Sessions. Below the navigation bar, there's a logo for BRIDGES-2 and a "Message of the Day". The main area features a file browser window showing a directory structure under /jet/home/wongk/. The browser includes buttons for View, Edit, Rename/Move, and Download. A message box displays a warning about connecting to vm003.br and terms of use. To the right, a file viewer window shows a list of files from yesterday, including various PDF and PPTX documents. At the bottom, there are buttons for Open in Terminal, New File, New Dir, Upload, Show Dotfiles, and Show Owner/Mode.

OPEN OnDemand

Bridges2 OnDemand Files Jobs Clusters Interactive Apps My Interactive Sessions

VPS
Log in to Bridges2

Username
Password

Log in

Directory

INSTALL
demand
tensorflow_datasets
tmp_ondemand_ocean_asc160019p_symlink

/jet/home/wongk/

View Edit A-z Rename/Move Download

name

.. TF-INSTALL ondemand tensorflow_datasets tmp_ondemand_ocean_asc160019p_symlink 2021-LAPENNA-UE.pdf EX-CNN-Forward-Backward.ipynb Untitled.ipynb Untitled1.ipynb Untitled2.ipynb cifar100.py ex-mnist.py

***** W A R N I N G *****
You have connected to vm003.br

This computing resource is the It is for authorized use only. notice of, and agree to comply Policy, available at http://www.psc.edu/bridges/notice.html. Improper use of this system may result in civil charges/criminal penalties. By continuing to use this system, you consent to these terms and conditions.

LOG OFF IMMEDIATELY if you do not consent to these terms and conditions.

For documentation on Bridges 2
Please contact help@psc.edu with any questions.

Browse... No files selected.

Yesterday

2021-LAPE...ll-2021.pptx
2021-LAPE...roduction.pdf
2021-LAPE...duction.pptx
2021-LAPE...stem-R1.pdf
2021-LAPE...tem-R1.pptx
2021-LAPE...-2-LA-R1.pdf
2021-LAPE...-LA-R1.pptx
2021-LAPE...alML-R1.pdf
2021-LAPE...IML-R1.pptx
2021-LAPE...DNN-R1.pdf
2021-LAPE...NN-R1.pptx
2021-LAPE...MLP-R1.pdf
2021-LAPE...LP-R1.pptx
2021-LAPE...NN1-R1.pdf
2021-LAPE...N1-R1.pptx
2021-LAPE...NN2-R1.pdf
2021-LAPE...N2-R1.pptx

Previous 30 Days

2021-LAPE...roduction.pdf
2021-LAPE...duction.pptx
2021-LAPE...stem-R1.pdf
2021-LAPE...tem-R1.pptx
2021-LAPE...-2-LA-R1.pdf
2021-LAPE...-LA-R1.pptx
2021-LAPE...alML-R1.pdf
2021-LAPE...IML-R1.pptx
2021-LAPE...DNN-R1.pdf
2021-LAPE...NN-R1.pptx
2021-LAPE...MLP-R1.pdf
2021-LAPE...LP-R1.pptx
2021-LAPE...NN1-R1.pdf
2021-LAPE...N1-R1.pptx
2021-LAPE...NN2-R1.pdf
2021-LAPE...N2-R1.pptx

Favorites

Recents Applications Desktop Documents Downloads

Media

Music Photos Movies

Tags

Red Orange Yellow

To... Open in Terminal New File New Dir Upload Show Dotfiles Show Owner/Mode

https://portal.xsede.org/data-management

MY XSEDE RESOURCES DOCUMENTATION ALLOCATIONS TRAINING USER FORUMS HELP ABOUT

Get Started Data Analysis Data Management User Guides Community Codes News Usage Policy MFA Containers

Data Transfer & Management

Last update: August 19, 2021

[Top of page](#)

Introduction
Globus Intro & Setup >
Using Globus >
scp & sftp
Data Integrity and Validation >
Data Transfer Performance >

Introduction

Transferring data includes moving files from local machines to XSEDE, as well as transfers between XSEDE resources. This section gives a high level overview on the recommended XSEDE data transfer methods.

There are a variety of methods for transferring files across XSEDE. XSEDE provides a Web Browser interface (easy to use, universally available), command-line interfaces (for casual use, scripting, and automation), and application programming interfaces (for developing applications). You may choose between Globus, globus-url-copy and uberftp, or scp and sftp. See [Table 1](#) below for details on each method.

Table 1. Data Transfer Methods

USAGE MODE	TRANSFER METHOD	THINGS TO KNOW
WEB BROWSER INTERFACE	Globus Web application	easy-to-use web interface; uses XSEDE web single sign-on; desktop download available
COMMAND LINE INTERFACE	Globus Command Line Interface (CLI) scp & sftp	managed, reliable, and auto-tuned transfer; supports scripting; uses XSEDE web single sign-on; requires command line/scripting skill; requires a Python installation easy, familiar interface; must use local (resource-specific) username and password; no automatic failure recovery

Transfer files via Globus

Set Up Globus

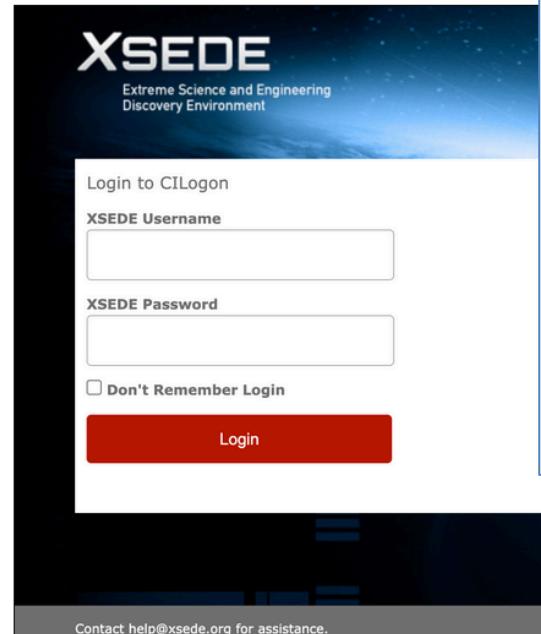
1. Link your XSEDE Identity

Make sure that Globus knows you are registered with XSEDE. This will allow you to use any of the features listed above. First, open a web browser and navigate to the [Globus website](#).

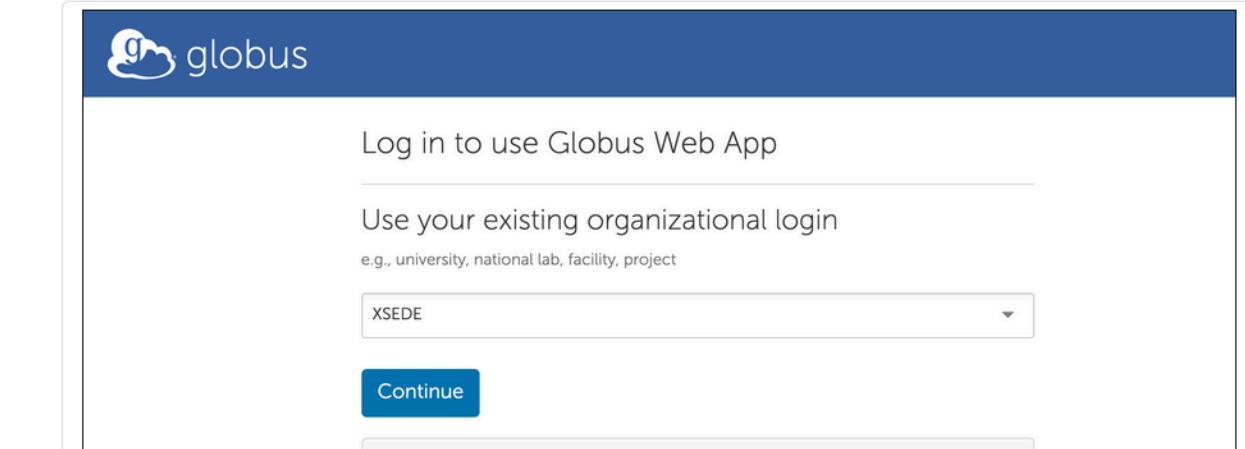
1. Click the "Login" button at the top-right of the page.

- If you've recently used Globus, you'll be automatically logged in. See [Add XSEDE to your Globus Profile](#).
- If you haven't used Globus recently (or ever), you will be prompted to select an organization where you already have an account. Type or select "XSEDE" to specify that you want to use your XSEDE identity as in Figure 1. below.

2. After clicking Continue, you will be asked to enter XSEDE Username and Password. You will have the familiar XSEDE interface, will display beginning with "idp.xsede.org" as in Figure 2 b



The screenshot shows the XSEDE login page. At the top, it says "XSEDE Extreme Science and Engineering Discovery Environment". Below that is a "Login to CILogon" section with fields for "XSEDE Username" and "XSEDE Password", both currently empty. There is a checkbox for "Don't Remember Login" and a red "Login" button. At the bottom, it says "Contact help@xsede.org for assistance."

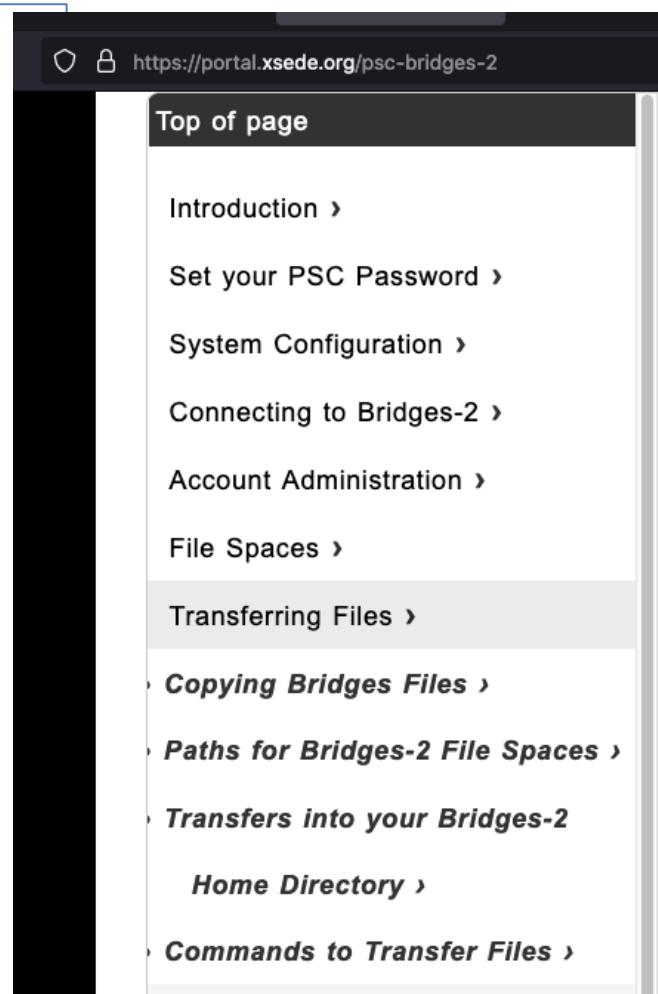


The screenshot shows the Globus login page. At the top, it has the Globus logo and the text "globus". Below that is a "Log in to use Globus Web App" section. Underneath it is a "Use your existing organizational login" section with a dropdown menu showing "XSEDE" and a "Continue" button.

Figure 2. Enter your XSEDE username and password at the XSEDE login page

Transfer files via Globus

- ✓ Run `globusconnectpersonal` on your local machine
- ✓ Go to globus.org and login with XSEDE credentials
- ✓ Select your local machine as your first collection
- ✓ Select your endpoint collection
 - Bridges2: psc#bridges2-xsede
 - Expanse: xsede#expanse
- ✓ Navigate to the file you want to send from your local machine and select it
- ✓ Specify path on remote machine
 - Bridges2 Home: /jet/home/<YOUR USERNAME>/
 - Expanse Home: /expanse/home/<YOUR USERNAME>/
 - Select start above the side your local machine is on



Transfer files via Globus

File Manager

Collection: REU1917 Desktop

Path: ~/Downloads/

Transfer & Sync Options

Start

NAME	LAST MODIFIED	SIZE
magma_bitbucket	7/9/2021, 11:31 AM	—
magma_test	7/9/2021, 9:51 AM	—
magma-2.5.4	7/1/2021, 10:24 AM	—
<input checked="" type="checkbox"/> magma-2.5.4.tar.gz	10/8/2020, 11:40 PM	7.94 MB
magma-2.6.1.tar.gz	7/27/2021, 10:14 AM	10.11 MB
magmadnn	6/9/2021, 11:58 AM	—
MagmaDnn Intro.pptx.pdf	7/27/2021, 10:19 AM	1.18 MB
magmadnn_dev	6/9/2021, 11:58 AM	—
magmadnn-drive	6/29/2021, 11:49 AM	—
magmadnn-testing	7/30/2021, 4:04 PM	—

NAME	LAST MODIFIED	SIZE
BIN	7/22/2021, 10:19 AM	—
magma	7/22/2021, 2:17 PM	—
magma-2.5.4.tar.gz	8/19/2021, 4:57 PM	7.94 MB
magma-build	8/2/2021, 12:02 PM	—
magma-build-2	8/2/2021, 12:01 PM	—
magma-test	8/2/2021, 10:39 AM	—
magmadnn-testing	7/30/2021, 3:46 PM	—
pytorch-classification	7/28/2021, 2:20 PM	—

FILE MANAGER

BOOKMARKS

ACTIVITY

ENDPOINTS

GROUPS

CONSOLE

FLOWS

ACCOUNT

LOGOUT

HELP

Panels

- ✓ Go to globus.org and login using your XSEDE credentials
- ✓ Download Globus Connect Personal from <https://www.globus.org/globus-connect-personal> so you can transfer files from your local machine
- ✓ For linux:
- ✓ You must use the standard system python for globusconnectpersonal, not an anaconda python
- ✓ To check: \$ which python3 (should return /usr/bin/python3)
- ✓ If it is not using the correct python, you must edit your PATH environment variable to use the standard system python (\$ export PATH=/usr/bin:\$PATH)
- ✓ Extract the tar.gz (\$ tar -xvf globusconnectpersonal-latest.tar.gz)
- ✓ \$ cd globusconnectpersonal-3.1.5
- ✓ \$./globusconnectpersonal
- ✓ Click Login and the machine will be added to your Globus portal
- ✓ In the globus portal when you search for a collection, there is a tab called “Your Collections” which your machine will be listed under, or you can search your machine’s name
- ✓ You must run the globusconnectpersonal program every time you want to transfer files from your local machine

Exercise

Python Calculations

1. Write a python code to compute $C = A \times B$ and plot a curve of the FLOPS against the matrix size N when the computation is done on a CPU. Do the same on the GPU.
2. Repeat question #5 using R

```
[18] import numpy as np  
  
A = np.random.rand(5000, 5000).astype('float64')  
B = np.random.rand(5000, 5000).astype('float64')
```

```
%timeit np.dot(A, B)
```

1 loop, best of 3: 7.39 s per loop



```
%%R  
library(dplyr)  
A<-matrix(runif(25000000),nrow=5000)  
B<-matrix(runif(25000000),nrow=5000)  
system.time(C<-A%*%B)
```

user	system	elapsed
15.449	0.025	7.840

Python Exercises on Linear Algebra

Google Colab

Python Introduction and
Linear Algebra Review

Boris Ivanovic

CS 231A

April 7, 2017

cs231n-colab-tutorial.ipynb

Linear Algebra in Python

Tensorflow Code examples

Exercise

Performance Calculations (BLAS)

<https://www.psc.edu/resources/bridges-2/user-guide-2/>

<https://software.intel.com/content/www/us/en/develop/documentation/mkl-tutorial-c/top/multiplying-matrices-using-dgemm.html>

<https://shaalltime.medium.com/benchmark-numpy-with-openblas-and-mkl-library-on-amd-ryzen-3950x-cpu-96184f91057f>

https://www.youtube.com/watch?v=fiNG_Btbx0g

Numbers : Lots of Them: bit, byte, FLOP (S)

- Core : computing unit : processor
- Dual core machine (Intel or AMD CPU) : a CPU with 2 cores, each core is a 2.4 GHz computing unit with 2GB of RAM (memory in the processor not disk space)
- Binary bits (b) : “0” or “1”, 1 Byte (B) = 8 bits
- Binary number : $11111111 = (2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0) = (2^8 - 1) = 255 !!$
- **32 bits** machine or operating system => largest integer (all positive) = $(2^{32} - 1) = (4,294,967,296 - 1)$ or range of integer = $-(2^{31})$ to $(2^{31} - 1)$
- **64 bits** machine or operating system => range of integer = $-(2^{63})$ to $(2^{63} - 1)$
- Kilo (K) = 10^3 (or 2^{10}) ; Mega (M) = 10^6 (or 2^{20}); **Giga (G)** = 10^9 (or GiB = 2^{30}); Tera (T billion) = 10^{12} (or 2^{40}) ; Peta (P) = 10^{15} (or 2^{50})
- **GB in base 10 unit 10^9 , GiB in 2 power unit (2^{30})**
- **FLoating Point Operation (+, -, /, *)** : $(10.1 + 0.1) * 1.0 / 2.0 = 5.1 \Rightarrow 3 \text{ FLOP}$
- FLOPS = FLOP per second :: 1 PetaFLOPS (kraken) = **10^{15} FLOP in one second**
- **FLOPS in a core = (clock rate) x (floating point operation in one clock cycle)**
- **Peak Rate = (FLOPS in one compute unit, core) x (no. of core)**

Exercises

1. A desktop computer has the following specification: quad-core, 24GB RAM, 2.5 GHz, and each core has 8 floating units (perform 8 DP operations in one clock cycle). What is the theoretical double precision peak performance of the desktop in FLOPS?
2. Given A is a $M \times K$ matrix, B is a $K \times N$ matrix, what is the number of floating point operation (FLOP) of $A \times B$?
3. Given $M=K=N$ for A and B , what is the maximum dimension N that the computer can hold?
4. Based on the answer above, what is the time needed to compute $C=A \times B$ if 90% of the theoretical rate of the desktop can be attained for the BLAS3 operation?

1. A desktop computer has the following specification: quad-core, 24GB RAM, 2.5 GHz, and each core has 8 floating units (perform 8 DP operations in one clock cycle). What is the theoretical double precision peak performance of the desktop in FLOPS?

FLOPS in one core= (clock rate) x (floating point operation in one clock cycle)

Peak Rate = (FLOPS in one core) x (no. of core)

Peak FLOPS = 2.5 GHz * 8 * 4 cores = 80 GFLOPS

2. Given A is a M x K matrix, B is a K x N matrix, what is the number of floating point operation (FLOP) of A x B?

C=A x B ; A : A(M , K) ; B : B(K, N) , C = C(M, N) ; one element of C need (2K-1) DP

FLOP = (2K-1) x M x N

3. Given M=K=N for A and B, what is the maximum dimension N that MA6633DC can hold?

For double precision float, we need 8 bytes to store each element.

If M=N=K=N, in total we need $3N^2$ elements

$3N^2 * 8 = 24 \text{ GB}$, thus **N = floor(sqrt(1e9)) = 31622**

4. Based on the answer above, what is the time needed to compute C=A x B if 90% of the theoretical rate of MA6633DC can be attained for the BLAS3 operation?

$90\% * 80 \text{ GFLOPS} = 72 \text{ GFLOPS}$

The FLOP we need to compute A x B is $\sim 2N^3$; **Time = 2 (31622)³ FLOP / 72 GFLOPS = 878 seconds**

Example: matrix-matrix multiplication (Level-3 BLAS)

www.netlib.org/blas/

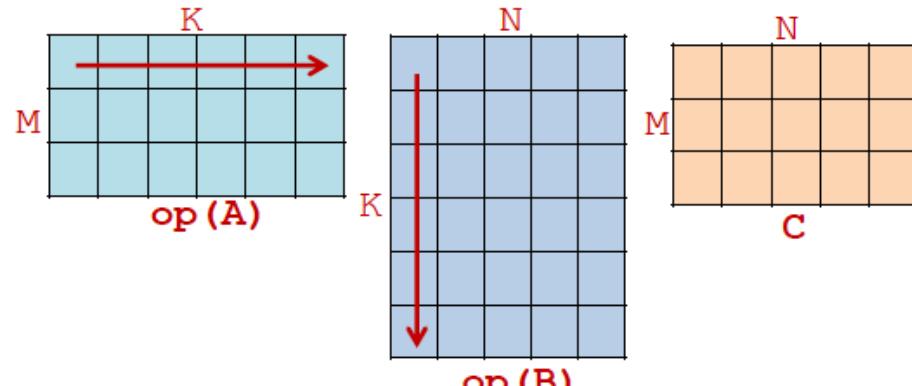
$$C \leftarrow \alpha \times op(A) \times op(B) + \beta \times C$$

SUBROUTINE **SGEMM**(TRANSA, TRANSB, M, N, K, ALPHA, A, LDA, B, LDB, BETA, C, LDC)

S: single precision

GE: general matrix

MM: matrix-matrix



TRANSA: the form of $op(A)$ to be used

'N' or 'n', $op(A) = A$

'T' or 't', $op(A) = A^T$

'C' or 'c', $op(A) = A^T$ (transpose, when A is real)

A^H (conjugate transpose, when A is complex in CGEMM)

M: the number of rows of $op(A)$ and C

N: the number of columns of $op(B)$ and C

K: the number of columns of $op(A)$ and the number of rows of $op(B)$

ALPHA, BETA: the scalar coefficients α and β

LDA, LDB, LDC: the first dimension of A , B and C as declared in the calling program

A: real array of dimension (LDA, ka), where ka is K when TRANSA='N', and is M otherwise

B: real array of dimension (LDB, kb), where kb is N when TRANSB='N', and is K otherwise

C: real array of dimension (LDC, N)

- To use BLAS and LAPACK:

1. Download the zipped file [blas_lapack_lib.zip](#) containing the pre-built libraries.
2. Unzip and put the four files **libblas.dll**, **libblas.lib**, **liblapack.dll** and **liblapack.lib** in the **GNU_emacs_Fortran** folder

- To link the BLAS library in compiling:

```
> gfortran -L. -lblas my_program.f90
```

Example: A test driver for using BLAS **sgemm**; compare with intrinsic function **matmul**

```
> gfortran -L. -lblas t blas.f90
```

```
program t blas
implicit none
integer,parameter :: m=2000, k=1500, n=1000
integer :: lda=m, ldb=k, ldc=m
real :: random, t1, t2, t3, alpha, beta
real :: a(m,k), b(k,n), c(m,n), c_intr(m,n)
integer :: i, j, iseed
!
!-- Assign random values to a, b and c
iseed = 65432
do j = 1, k
do i = 1, m
  a(i,j) = random(iseed)
end do
end do
do j = 1, n
do i = 1, k
  b(i,j) = random(iseed)
end do
end do
do j = 1, n
do i = 1, m
  c(i,j) = random(iseed)
end do
end do
alpha = random(iseed)
beta = random(iseed)
```

>Continued on next page...

```
-- Compute alfa*a*b+beta*c using intrinsic function matmul and BLAS sgemm
call cpu_time(t1)
!
c_intr = alpha*matmul(a,b) +beta*c
!
call cpu_time(t2)
!
call sgemm('N','N',m,n,k,alpha,a,lda,b,ldb,beta,c,ldc)
!
call cpu_time(t3)
!
!-- Find the maximum error of abs(c-c_intr) using intrinsic function maxval
write(*,*) 'Max error: ', maxval(abs(c-c_intr))
write(*,*) 'Elapsed time of matmul: ', t2-t1
write(*,*) 'Elapsed time of sgemm: ', t3-t2
!
end program
```

```
> gfortran -L. -lblas t blas.f90 random.f90
> a.exe
> Max error:  1.23977661E-05
> Elapsed time of matmul:  3.5724230
> Elapsed time of sgemm:  25.474962
```

```
(base) [wongk@bridges2-login013 LAPENNA]$ interact --ntasks-per-node=4
```

```
(base) [wongk@bridges2-login013 LAPENNA]$ interact --gpu
```

```
#!/bin/bash
# use the bash shell
set -x
# echo each command to standard out before running it
date
# run the Unix 'date' command
echo "Hello world, from Bridges-2!"
# run the Unix 'echo' command
```

Batch Script

```
(base) [wongk@bridges2-login013 LAPENNA]$ sbatch bscript
Submitted batch job 3507672
(base) [wongk@bridges2-login013 LAPENNA]$ more slurm-3507672.out
+ date
Wed Sep  8 12:08:25 EDT 2021
+ echo 'Hello world, from Bridges-2!'
Hello world, from Bridges-2!
```

```
(base) [wongk@bridges2-login012 BLAS]$ squeue -u wongk
JOBID PARTITION    NAME    USER ST      TIME NODES NODELIST(REASON)
```

<https://software.intel.com/content/www/us/en/develop/documentation/mkl-tutorial-fortran/top/multiplying-matrices-using-dgemm.html>

(base) [wongk@bridges2-login012 BLAS]\$ more mm-intel.f90

```
PROGRAM MAIN
IMPLICIT NONE
```

```
DOUBLE PRECISION ALPHA, BETA, t1, t2
INTEGER M, K, N, I, J
PARAMETER (M=4000, K=4000, N=4000)
DOUBLE PRECISION A(M,K), B(K,N), C(M,N)
```

```
PRINT *, "This example computes real
matrix C=alpha*A*B+beta*C"
PRINT *, "using Intel(R) MKL function
dgemm, where A, B, and C"
PRINT *, "are matrices and alpha and beta
are double precision "
PRINT *, "scalars"
PRINT *, ""
```

```
PRINT *, "Initializing data for matrix
multiplication C=A*B for "
PRINT 10, " matrix A(",M," x",K, ") and
matrix B(", K," x", N, ")"
10 FORMAT(a,I5,a,I5,a,I5,a,I5,a)
PRINT *, ""
ALPHA = 1.0
BETA = 0.0
```

```
PRINT *, "Intializing matrix data"
PRINT *, ""
```

```
DO I = 1, M
  DO J = 1, K
    A(I,J) = (I-1) * K + J
  END DO
END DO

DO I = 1, K
  DO J = 1, N
    B(I,J) = -((I-1) * N + J)
  END DO
END DO

DO I = 1, M
  DO J = 1, N
    C(I,J) = 0.0
  END DO
END DO

call cpu_time(t1)

PRINT *, "Computing matrix product using Intel(R)
MKL DGEMM "
PRINT *, "subroutine"
CALL DGEMM( 'N' , 'N' , M , N , K , ALPHA , A , M , B , K , BETA , C , M )
PRINT *, "Computations completed."
PRINT *, ""

call cpu_time(t2)
PRINT *, "Elapse time of MM = ", t2-t1
PRINT *, ""
PRINT *, "GFLOPS of MM = ",
(N*N*N)*2.0/1000.0/1000.0/1000.0/t2-t1
PRINT *, ""

PRINT *, "Example completed."
STOP

END
```

```
all:simpleblas openblas
simpleblas:
    gfortran -O3 mm-intel.f90 -o mm-simple-exe ./BLAS-
3.10.0/blas_LINUX.a
openblas:
    gfortran -O3 mm-intel.f90 -o mm-openblas-exe -lopenblas
```

Makefile

```
(base) [wongk@bridges2-login012 BLAS]$ module list
```

Currently Loaded Modules:

```
1) allocations/1.0  2) gcc/10.2.0   3) openmpi/4.0.5-gcc10.2.0  4)
openblas/0.3.12-gcc10.2.0
```

```
(base) [wongk@bridges2-login012 BLAS]$ make
gfortran -O3 mm-intel.f90 -o mm-simple-exe ./BLAS-3.10.0/blas_LINUX.a
gfortran -O3 mm-intel.f90 -o mm-openblas-exe -lopenblas
(base) [wongk@bridges2-login012 BLAS]$
```

<https://www.openblas.net/>

```
(base) [wongk@bridges2-login012 BLAS]$
./mm-simple-exe
```

matrix A,B,C (4000 x 4000)

Intializing matrix data

Elapse time of MM
= 42.145660999999997

GFLOPS of MM = 2.9104307852566587

```
(base) [wongk@bridges2-login012 BLAS]$
./mm-openblas-exe
```

matrix A, B, C (4000 x 4000)

Intializing matrix data

Elapse time of MM
= 2.7656499999999999

GFLOPS of MM = 44.150656172802520

Homework Exercise

Run BLAS3 (DGEMM) in C

Exercise
Performance Calculations
(LAPACK, MAGMA)

Download Magma

Go to <https://icl.utk.edu/magma/software/index.html> and download latest magma

```
$ cd ~
```

```
$ wget http://icl.utk.edu/projectsfiles/magma/downloads/magma-2.6.1.tar.gz
```

```
$ tar -xvf magma-2.6.1.tar.gz
```

```
$ cd magma-2.6.1
```

```
$ cp make.inc-examples/make.inc.openblas make.inc
```

```
$ vim Makefile
```

Edit line 104 to change the install prefix to /jet/home/\$(USER)/magma-build

```
$ vim make.inc
```

Comment out lines 98 106 118 in make.inc to avoid the make.checks which yield errors

Building Magma

```
$ module load cuda/10.2.0 cudnn/8.0.4  
$ make -j256  
$ make install  
$ interact --gpu  
$ cd testing  
$ ./testing_dgemm  
$ exit
```

Note: if using Magma from bitbucket you will need to have python

However, “module load python” will change g++ to version 10.2, and Magma needs g++ version 8 or earlier, so we need to create a python link and add it to \$PATH

```
$ mkdir BIN & ln -s /usr/bin/python3  
BIN/python  
$ export  
PATH=/jet/home/$USER/BIN:$PATH
```

```
[jhalloy@ssohub ~]$ gsissb bridges2
```

```
***** W A R N I N G *****
```

You have connected to br014.ib.bridges2.psc.edu, a login node of Bridges 2.
This computing resource is the property of the Pittsburgh Supercomputing Center.
It is for authorized use only. By using this system, all users acknowledge

```
(base) [jhalloy@bridges2-login014 ~]$ wget
```

```
http://icl.utk.edu/projectsfiles/magma/downloads/magma-2.6.1.tar.gz
```

```
--2021-09-08 14:12:41-- http://icl.utk.edu/projectsfiles/magma/downloads/magma-  
2.6.1.tar.gz
```

```
Resolving icl.utk.edu (icl.utk.edu)... 160.36.131.221
```

```
Connecting to icl.utk.edu (icl.utk.edu)|160.36.131.221|:80... connected.
```

```
HTTP request sent, awaiting response... 200 OK
```

```
Length: 10111350 (9.6M) [application/x-gzip]
```

```
Saving to: 'magma-2.6.1.tar.gz'
```

```
magma-2.6.1.tar.gz
```

```
100%[=====>] 9.64M 3.41MB/s in 2.8s
```

```
2021-09-08 14:12:44 (3.41 MB/s) - 'magma-2.6.1.tar.gz' saved [10111350/10111350]
```

Building Magma

```
(base) [jhalloy@bridges2-login014 ~]$ tar -xvf magma-2.6.1.tar.gz
```

magma-2.6.1/

magma-2.6.1/ReleaseNotes

magma-2.6.1/make.gen.hipMAGMA

magma-2.6.1/include/

magma-2.6.1/include/magmablas_d_v1_map.h

magma-2.6.1/include/magma_c.h

magma-2.6.1/include/magma_zlapack.h

magma-2.6.1/include/magma_zbulgeinc.h

magma-2.6.1/include/magma_svbatched.h

magma-2.6.1/include/magmablas_ds_v1_map.h

magma-2.6.1/include/magma_dgehrd_m.h

magma-2.6.1/include/magmablas_zc.h

magma-2.6.1/include/magma_config.h.in

....

```
(base) [jhalloy@bridges2-login014 ~]$ cd magma-2.6.1
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ pwd  
/jet/home/jhalloy/magma-2.6.1
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ cp  
make.inc-examples/make.inc.openblas make.inc
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ vim  
Makefile
```

Makefile, make.inc

```
LDFLAGS ?= $(FPIC)           -fopenmp
ifndef $(and DEVCCFLAGS, NVCCFLAGS)
DEVCCFLAGS ?= -O3 -DNDEBUG -DADD_
endif
# DEVCCFLAGS are populated later
#in `backend-specific`
# Extension for object files: o for unix, obj
#for Windows?
o_ext      ?= o
# where to install to?
prefix     ?= /jet/home/$USER/magma-build
# LINE 104
# -----
# MAGMA-specific programs & flags
ifeq ($(blas_fix),1)
# prepend -lblas_fix to LIB
#(it must come before LAPACK library/framework)
LIB := -lblas_fix $(LIB)
endif
LIBS      = $(LIBDIR) $(LIB)
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ vim make.inc
# checks
# check for openblas
#-include make.check-openblas # line 98
# -----
# backend-specific
# add appropriate cuda flags
ifeq ($(BACKEND),cuda)
# -include make.check-cuda # line 106
DEVCCFLAGS += -Xcompiler "$(FPIC)" -std=c++11
# link with cuda specific libraries
LIB += -lcublas -lcusparse -lcudart -lcudadevrt
INC += -I$(CUDADIR)/include
endif
# add appropriate HIP flags
ifeq ($(BACKEND),hip)
# -include make.check-hip # line 118
DEVCCFLAGS += $(FPIC) -std=c++11
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ module load cuda/10.2
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ module load cudnn
```

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ module list
```

Currently Loaded Modules:

1) allocations/1.0 2) cuda/10.2.0 3) cudnn/8.0.4

Buidling Magma

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ nproc
```

256

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ make -j256
```

```
cp include/magma_config.h.in include/magma_config.h
```

```
sed -i -e 's/#cmakedefine MAGMA_CUDA_ARCH_MIN @MAGMA_CUDA_ARCH_MIN@/#define  
MAGMA_CUDA_ARCH_MIN 300/g' include/magma_config.h
```

```
cc -O3 -fPIC -DNDEBUG -DADD_ -Wall -fopenmp -std=c99 -c -o control/sizeptr.o control/sizeptr.c
```

```
sed -i -e 's/#cmakedefine MAGMA_HAVE_CUDA/#define MAGMA_HAVE_CUDA/g'  
include/magma_config.h
```

```
touch control/sizeptr.o
```

```
gfortran -O3 -fPIC -DNDEBUG -DADD_ -Wall -Wno-unused-dummy-argument -c -o testing/lin/zbdt01.o  
testing/lin/zbdt01.f ....
```

Building Magma

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ make install
cp include/magma_config.h.in include/magma_config.h
sed -i -e 's/#cmakedefine MAGMA_CUDA_ARCH_MIN @MAGMA_CUDA_ARCH_MIN@
#define MAGMA_CUDA_ARCH_MIN 300/g' include/magma_config.h
sed -i -e 's/#cmakedefine
MAGMA_HAVE_CUDA#define MAGMA_HAVE_CUDA/g' include/magma_config.h
sed -i -e 's/#cmakedefine
MAGMA_HAVE_HIP#define MAGMA_HAVE_HIP/g' include/magma_config.h
mkdir -p /jet/home/jhalloy/magma-build
mkdir -p /jet/home/jhalloy/magma-build/include
mkdir -p /jet/home/jhalloy/magma-build/lib
mkdir -p /jet/home/jhalloy/magma-build/lib/pkgconfig
cp include/*.h      /jet/home/jhalloy/magma-build/include
cp include/*.mod    /jet/home/jhalloy/magma-build/include
cp sparse/include/*.h /jet/home/jhalloy/magma-build/include
cp lib/libmagma.a lib/libmagma.so lib/libmagma_sparse.a lib/libmagma_sparse.so
/jet/home/jhalloy/magma-build/lib
make pkgconfig
make[1]: Entering directory '/jet/home/jhalloy/magma-test/magma-2.6.1'
mkdir -p /jet/home/jhalloy/magma-build/lib/pkgconfig
cat lib/pkgconfig/magma.pc.in          ||
sed -e s:@INSTALL_PREFIX@:"/jet/home/jhalloy/magma-build":    ||
sed -e s:@CFLAGS@:"-DNDEBUG -DADD_ -fopenmp -l/usr/local/cuda/include": ||
sed -e s:@LIBS@:"-fopenmp -L/usr/local/openblas/lib -lopenblas -lcublas -lcusparse
-lcudart -lcudadevrt -lcublas -lcudart": ||
sed -e s:@MAGMA_REQUIRED@:          \
> /jet/home/jhalloy/magma-build/lib/pkgconfig/magma.pc
make[1]: Leaving directory '/jet/home/jhalloy/magma-test/magma-2.6.1'
```

Running Magma - DGEMM

```
(base) [jhalloy@bridges2-login014 magma-2.6.1]$ interact --gpu
```

A command prompt will appear when your session begins
"Ctrl+d" or "exit" will end your session

```
--partition=GPU-small,GPU-shared --gpus=v100:1
salloc -J Interact --partition=GPU-small,GPU-shared --gpus=v100:1
salloc: Pending job allocation 3509234
salloc: job 3509234 queued and waiting for resources
salloc: job 3509234 has been allocated resources
salloc: Granted job allocation 3509234
salloc: Waiting for resource configuration
salloc: Nodes v002 are ready for job
```

```
(base) [jhalloy@v002 magma-2.6.1]$ ls
```

```
blas_fix  CMake.src.hip  example  interface_cuda  magmablas_hip  make.check-cuda
Makefile      make.inc          README-v2.txt  sparse   tools

BUGS.txt      control    fortran    interface_hip      make.check-hip      Makefile.gen
make.inc-examples          README-Windows  sparse_hip

CMakeLists.txt  COPYRIGHT    HIP-notes.txt  lib      make.check-acml  make.check-mkl
Makefile subdir README          ReleaseNotes  src

CMake.src.cuda  docs      include    magmablas      make.check-atlas  make.check-openblas
make.gen.hipMAGMA README_FP16_Iterative_Refinement.txt results  testing
```

Running Magma - DGEMM

```
(base) [jhalloy@v002 magma-2.6.1]$ cd testing
(base) [jhalloy@v002 testing]$ ./testing_dgemm
% MAGMA 2.6.1 32-bit magma_int_t, 64-bit pointer.
Compiled with CUDA support for 3.0
% CUDA runtime 10020, driver 11020. OpenMP threads 1.
% device 0: Tesla V100-SXM2-32GB, 1530.0 MHz clock, 32510.5 MiB memory, capability 7.0
% Wed Sep  8 14:38:59 2021
% Usage: ./testing_dgemm [options] [-h|--help]
% If running lapack (option --lapack), MAGMA and cuBLAS error are both computed
% relative to CPU BLAS result. Else, MAGMA error is computed relative to cuBLAS result.
% transA = No transpose, transB = No transpose
%   M      N      K    MAGMA Gflop/s (ms)  cuBLAS Gflop/s (ms)  CPU Gflop/s (ms)
MAGMA error  cuBLAS error
%=====
1088 1088 1088 1501.15 ( 1.72) 1548.93 ( 1.66) --- (---) 2.09e-17 --- ok
2112 2112 2112 5448.59 ( 3.46) 5475.39 ( 3.44) --- (---) 1.52e-18 --- ok
3136 3136 3136 5501.02 ( 11.21) 6119.18 ( 10.08) --- (---) 8.36e-19 --- ok
4160 4160 4160 5655.88 ( 25.46) 6207.80 ( 23.19) --- (---) 1.09e-18 --- ok
5184 5184 5184 5676.09 ( 49.09) 7064.41 ( 39.44) --- (---) 7.87e-19 --- ok
6208 6208 6208 6234.40 ( 76.75) 7279.16 ( 65.74) --- (---) 6.01e-19 --- ok
7232 7232 7232 5713.66 ( 132.40) 6783.72 ( 111.52) --- (---) 9.52e-19 --- ok
8256 8256 8256 5991.68 ( 187.84) 6714.29 ( 167.62) --- (---) 7.83e-19 --- ok
9280 9280 9280 5928.94 ( 269.59) 7371.75 ( 216.82) --- (---) 6.58e-19 --- ok
10304 10304 10304 5925.37 ( 369.26) 7409.45 ( 295.30) --- (---) 5.63e-19 --- ok
```

Exercise on PSC bridges

HPL Benchmark

```
wget https://www.netlib.org/benchmark/hpl/hpl-2.3.tar.gz
```

HPL (High Performance Linpack): Solving $Ax = b$

<http://www.netlib.org/benchmark/hpl/>

$$\begin{aligned} 2x_1 + 2x_2 + 2x_3 &= 1 \\ 3x_1 + 4x_2 + 5x_3 &= 2 \\ 4x_1 + 6x_2 + 7x_3 &= 3. \end{aligned}$$

$$A = \begin{bmatrix} 2 & 2 & 2 \\ 0 & 1 & 2 \\ 4 & 6 & 7 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1/2 \\ 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 2 & 2 \\ 3 & 4 & 5 \\ 4 & 6 & 7 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 2 & 2 \\ 0 & 1 & 2 \\ 0 & 2 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1/2 \\ 1 \end{bmatrix}.$$

$$A = \begin{bmatrix} 2 & 2 & 2 \\ 0 & \star & \star \\ 0 & \star & \star \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ \star \\ \star \end{bmatrix}$$

$$A = \begin{bmatrix} 2 & 2 & 2 \\ 0 & 1 & 2 \\ 0 & 0 & -1 \end{bmatrix}, \quad b = \begin{bmatrix} 1 \\ 1/2 \\ 0 \end{bmatrix}.$$

$$x_3 = 0, \quad x_2 = 1/2 - x_3 = 1/2, \quad 2x_1 + 2x_2 + 2x_3 = 1 \implies x_1 = 0.$$

Total operation count for Gaussian elimination with backward substitution

$$\frac{2}{3}n^3 + \frac{3}{2}n^2 - \frac{7}{6}n.$$

Jaguar (ORNL) : World Fastest Computer, 1.759 PF (2009)

- FLOPS – Floating Point Operation Per Second
- GFLOPS = 10^9 FLOPS ; TFLOPS = 10^{12} ; PFLOPS = 10^{15}
- FLOPS = (clock rate) x (floating point operation in one clock cycle)
- Peak Rate = (FLOPS in one CPU) x (no. of CPU)
- Cray XT5 one core AMD Opteron :
 - Rpeak : (2.6 GHz) x (4) x (224162 cores) = **2331284 GFLOPS**
 - Rmax : 1759000 GFLOPS → **75.4% of peak**

http://www.crc.nd.edu/~rich/CRC_EPYC_Cluster_Build_Feb_2018/Installing%20and%20running%20HPL%20on%20AMD%20EPYC%20v2.pdf

<https://developer.amd.com/spack/hpl-benchmark/>

<https://ngc.nvidia.com/catalog/containers/nvidia:hpc-benchmarks>

https://downloads.dell.com/manuals/all-products/esuprt_software/esuprt_it_ops_datcentr_mgmt/high-computing-solution-resources_white-papers79_en-us.pdf

Jaguar (ORNL) : World Fastest Computer, Peak=2.33 PF, 1.759 PF (2009)

- Solve $Ax = b$ using a standard benchmark C program (HPL)
- Nmax : Size of A for HPL (Solve $Ax=b$) = **5474272**
- Total Memory needed = (Nmax) x (Nmax) x (8 Bytes) = **239741 GB**
- Memory needed per core = **1.07 GB**, Max Memory = **2GB/core**
- Elapse Time : $2(Nmax)(Nmax)(Nmax)/3/Rmax \sim = 13$ hrs, Core=**224162**

Manufacturer:	Summit	IBM
Cores:	2,414,592	
Memory:	2,801,664 GB	
Processor:	IBM POWER9 22C 3.07GHz	
Interconnect:	Dual-rail Mellanox EDR Infiniband	
Performance		
Linpack Performance (Rmax)	148,600 TFlop/s	
Theoretical Peak (Rpeak)	200,795 TFlop/s	
Nmax	16,473,600	
HPCG [TFlop/s]	2,925.75	
Power Consumption		
Power:	10,096.00 kW (Submitted)	

Manufacturer:	Fukagu	Fujitsu
Cores:	7,299,072	
Memory:	4,866,048 GB	
Processor:	A64FX 48C 2.2GHz	
Interconnect:	Tofu interconnect D	
Performance		
Linpack Performance (Rmax)	415,530 TFlop/s	
Theoretical Peak (Rpeak)	513,855 TFlop/s	
Nmax	20,459,520	
HPCG [TFlop/s]	13,366.4	
Power Consumption		
Power:	28,334.50 kW (Submitted)	

```
wget https://www.netlib.org/benchmark/hpl/hpl-2.3.tar.gz
```

```
(base) [wongk@bridges2-login012 ~]$ wget https://www.netlib.org/benchmark/hpl/hpl-2.3.tar.gz
--2021-09-08 12:58:45-- https://www.netlib.org/benchmark/hpl/hpl-2.3.tar.gz
Resolving www.netlib.org (www.netlib.org)... 160.36.131.221
Connecting to www.netlib.org (www.netlib.org)|160.36.131.221|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 660871 (645K) [application/x-gzip]
Saving to: 'hpl-2.3.tar.gz'

hpl-2.3.tar.gz      100%[=====] 645.38K  1.95MB/s  in 0.3s
```

```
2021-09-08 12:58:46 (1.95 MB/s) - 'hpl-2.3.tar.gz' saved [660871/660871]
```

```
(base) [wongk@bridges2-login012 ~]$ tar xvf hpl-2.3.tar.gz
(base) [wongk@bridges2-login012 ~]$ cd hpl-2.3
(base) [wongk@bridges2-login012 hpl-2.3]$ ls
AUTHORS ChangeLog Makefile README acinclude.m4 config.sub include missing www
BUGS HISTORY Makefile.am THANKS aclocal.m4 configure install-sh setup
COPYING INSTALL Makefile.in TODO compile configure.ac makes src
COPYRIGHT Make.top NEWS TUNING config.guess depcomp man testing
(base) [wongk@bridges2-login012 hpl-2.3]$
```

```
(base) [wongk@bridges2-login012 ~]$ module load gcc/10.2.0 openmpi/4.0.5-gcc10.2.0 openblas/0.3.12-gcc10.2.0
(base) [wongk@bridges2-login012 ~]$ module list
```

Currently Loaded Modules:

1) allocations/1.0 2) gcc/10.2.0 3) openmpi/4.0.5-gcc10.2.0 4) openblas/0.3.12-gcc10.2.0

```
(base) [wongk@bridges2-login012 hpl-2.3]$ pwd  
/jet/home/wongk/hpl-2.3  
(base) [wongk@bridges2-login012 hpl-2.3]$ ./configure --prefix=/jet/home/wongk/hpl-2.3  
checking for gcc... /jet/packages/spack/opt/spack/linux-centos8-zen/gcc-8.3.1/gcc-10.2.0-  
tfzxq7udz2a53dmujvasy4uz33t27iwv/bin/gcc  
checking whether the C compiler works... Yes
```

```
config.status: creating Makefile  
config.status: creating src/Makefile  
config.status: creating testing/Makefile  
config.status: creating include/hplconfig.h  
config.status: executing depfiles commands
```

HPL installation on bridges2

```
(base) [wongk@bridges2-login012 hpl-2.3]$ make  
.....  
make[1]: Entering directory '/jet/home/wongk/hpl-2.3'  
make[1]: Nothing to be done for 'all-am'.  
make[1]: Leaving directory '/jet/home/wongk/hpl-2.3'  
(base) [wongk@bridges2-login012 hpl-2.3]$ make install
```

```
make[1]: Leaving directory '/jet/home/wongk/hpl-2.3'  
(base) [wongk@bridges2-login012 hpl-2.3]$ cd testing  
(base) [wongk@bridges2-login012 testing]$ ls  
Makefile Makefile.am Makefile.in matgen pmatgen ptest ptimer timer xhpl  
(base) [wongk@bridges2-login012 testing]$ cd ptest  
(base) [wongk@bridges2-login012 ptest]$ ls  
HPL.dat HPL_pddriver.c HPL_pddriver.o HPL_pdinfo.c HPL_pdinfo.o HPL_pdtest.c HPL_pdtest.o  
(base) [wongk@bridges2-login012 ptest]$ cp HPL.dat ..../bin/.  
(base) [wongk@bridges2-login012 ptest]$ cd ..../bin/  
(base) [wongk@bridges2-login012 bin]$ ls  
HPL.dat xhpl
```

```
(base) [wongk@r001 bin]$ mpirun -np 4 ./xhpl
```

```
=====
HPLinpack 2.3 -- High-Performance Linpack benchmark -- December 2, 2018
Written by A. Petitet and R. Clint Whaley, Innovative Computing Laboratory, UTK
Modified by Piotr Luszczek, Innovative Computing Laboratory, UTK
Modified by Julien Langou, University of Colorado Denver
=====
```

An explanation of the input/output parameters follows:

T/V : Wall time / encoded variant.
N : The order of the coefficient matrix A.
NB : The partitioning blocking factor.
P : The number of process rows.
Q : The number of process columns.
Time : Time in seconds to solve the linear system.
Gflops : Rate of execution for solving the linear system.

The following parameter values will be used:

N : 8000
NB : 128
PMAP : Row-major process mapping
P : 2 1
Q : 2 1
PFACT : Left
NBMIN : 2
NDIV : 2
RFACT : Left
BCAST : 1ring
DEPTH : 0
SWAP : Mix (threshold = 64)
L1 : transposed form
U : transposed form
EQUIL : yes
ALIGN : 8 double precision words

```
(base) [wongk@r001 bin]$ mpirun -np 4 ./xhpl
```

- The matrix A is randomly generated for each test.
- The following scaled residual check will be computed:
$$\|Ax-b\|_{\infty} / (\text{eps} * (\|x\|_{\infty} * \|A\|_{\infty} + \|b\|_{\infty}) * N)$$
- The relative machine precision (eps) is taken to be **1.110223e-16**
- Computational tests pass if scaled residuals are less than **16.0**

T/V	N	NB	P	Q	Time	Gflops
WR00L2L2	8000	128	2	2	3.22	1.0619e+02

HPL_pdgesv() start time Wed Sep 8 11:42:57 2021

HPL_pdgesv() end time Wed Sep 8 11:43:00 2021

$\|Ax-b\|_{\infty}/(\text{eps} * (\|A\|_{\infty} * \|x\|_{\infty} + \|b\|_{\infty}) * N) = 2.73042026e-03 \dots \text{PASSED}$

T/V	N	NB	P	Q	Time	Gflops
WR00L2L2	8000	128	1	1	10.67	3.1986e+01

HPL_pdgesv() start time Wed Sep 8 11:43:02 2021

HPL_pdgesv() end time Wed Sep 8 11:43:13 2021

$\|Ax-b\|_{\infty}/(\text{eps} * (\|A\|_{\infty} * \|x\|_{\infty} + \|b\|_{\infty}) * N) = 3.11002919e-03 \dots \text{PASSED}$

```
(base) [wongk@r001 bin]$ more HPL.dat
HPLinpack benchmark input file
Innovative Computing Laboratory, University of Tennessee
HPL.out          output file name (if any)
6                device out (6=stdout,7=stderr,file)
1                # of problems sizes (N)
8000 29 30 34 35  Ns
1                # of NBs
128 1 2 3 4      NBs
0                PMAP process mapping (0=Row-,1=Column-major)
2                # of process grids (P x Q)
2 1 4            Ps
2 1 4 1          Qs
16.0             threshold
1                # of panel fact
0 1 2            PFACTs (0=left, 1=Crout, 2=Right)
1                # of recursive stopping criterium
2 4              NBMINS (>= 1)
1                # of panels in recursion
2                NDIVs
1                # of recursive panel fact.
0 1 2            RFACTs (0=left, 1=Crout, 2=Right)
1                # of broadcast
0                BCASTs (0=1rg,1=1rM,2=2rg,3=2rM,4=Lng,5=LnM)
1                # of lookahead depth
0                DEPTHs (>=0)
2                SWAP (0=bin-exch,1=long,2=mix)
64               swapping threshold
0                L1 in (0=transposed,1=no-transposed) form
0                U  in (0=transposed,1=no-transposed) form
1                Equilibration (0=no,1=yes)
8                memory alignment in double (> 0)
(base) [wongk@r001 bin]$
```

```
(base) [wongk@bridges2-login013 bin]$ more bscript-hpl
#!/bin/bash
#SBATCH -N 1
#SBATCH -p RM
#SBATCH -t 10:00
#SBATCH --ntasks-per-node=4

# this job will ask for 1 full RM node (4 cores) for 10 minutes
# this job would potentially charge 1 RM SUs

#echo commands to stdout
set -x

# run the Unix 'date' command
date

# move to working directory
# - all input data is stored in this directory
# - all output should be stored in this directory
# - please note that groupname should be replaced by your groupname
# - username should be replaced by your username
# - path-to-directory should be replaced by the path to your directory where the
# executable is

cd /jet/home/wongk/LAPENNA/HPL/hpl-2.3/bin

# run a pre-compiled program which is already in your project space

mpirun -np 4 ./xhpl
```

```
(base) [wongk@bridges2-login013 bin]$ more
slurm-3507871.out
+ date
Wed Sep  8 12:20:06 EDT 2021
+ cd /jet/home/wongk/LAPENNA/HPL/hpl-2.3/bin
+ mpirun -np 4 ./xhpl
```

```
(base) [wongk@bridges2-login013 bin]$ sbatch
bscript-hpl
Submitted batch job 3507871
```

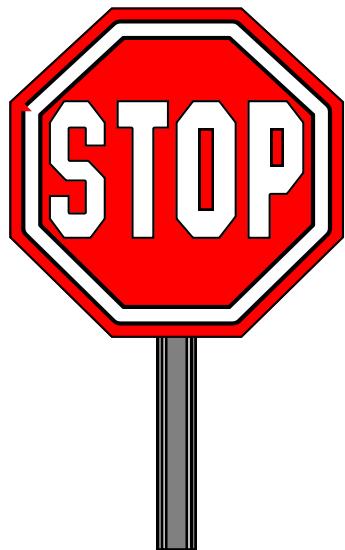
Homework Exercises

1. A desktop computer has the following specification: quad-core, 24GB RAM, 2.5 GHz, and each core has 8 floating units (perform 8 DP operations in one clock cycle). What is the theoretical double precision peak performance of the desktop in FLOPS?
2. Given a square $N \times N$ matrix A , what is the maximum dimension N that the computer can hold?
3. Given the $N \times N$ matrix, A , from (2), what is the number of floating point operation (FLOP) to solve $Ax=b$?
4. Based on the answer above, what is the time needed to solve $Ax=b$ if 90% of the theoretical rate of the desktop can be attained for such operation?
5. Run a few HPL tests on bridges2 and give a rough estimation of the percentage performance of the HPL benchmark against the peak performance. {2.3 (2.3 GHz) -3.4 (3.4 GHz) TFLOPS peak}.

Install and run the same HPL exercises on bridges2 on SDSC expanse and compare their results.

(Advance users) Install and run the same HPL exercises on bridges2 using spack.

The End



- The End!

