FACULTY HACK @GATEWAYS 23

Sunday Check-in: Syllabus, Sample Exercises, Supporting Gateways

HTTPS://HACKHPC.GITHUB.IO/FACULTYHACK-GATEWAYS23

SGX3 Extend. Expand. Exemplify. · Oak Ridge National Laboratory · omnibond Engineering ▪ Trust ▪ Identity · VOLTRON DATA · TACC TEXAS ADVANCED COMPUTING CENTER

Instructions

1. Create three slides after your team introduction slide(s) for:

   a. **Syllabus**
   b. **Sample Exercises**
   c. **Supporting Gateways**

   One slide for each response if necessary.

Practical Introduction to HPC and Research Computing

CAMSA Team

CAMSA FACULTY-HACK

4: Syllabus

- This course provides exposure to advanced topics in computer networks including recent research findings in this field. The topics include: internetworking, Internet concept, Client-server model for applications, Network and internet management. Also, this course covers recently emerging protocols and technologies such as: Virtualization and Software Defined Networks (SDNs), IPv6, wireless networks, Secure Socket Layer, and Transport Layer Security.
- The course integrates also hands on labs about the usage of High-Performance Computing (HPC) in computer networks and other computing Disciplines. The goal is to allow students to use such resources in their other courses or future research or experiments.

5: Sample Exercises

- In the past I used existing public materials in testbeds such as Geni.net, XSEDE, Deterlab, etc. Additionally, I provide samples of my own experience implementing those experiments and my own feedback to previous students.
- There is one particular example I used and like this year is ChameleonCloud shared experiments portal: (https://www.chameleoncloud.org/experiment/share/)
- The portal focuses on an idea that I like (reproducibility), and allows users to submit their feedback and also their own experiments.

- I am planning also to utilize the new TAMU cluster ACES (https://portal-aces.hprc.tamu.edu/pun/sys/dashboard)
- I am a member of SWEETER grant with TAMU with the aim of enabling the spread of usage of HPC resources across A&M system and the region.

6: Supporting Gateways

- https://access-ci.org/
- https://hprc.tamu.edu/

Bulldogs Team

Team Member: Dr. Rui Zhu
(Kettering University)

Mentor: Dr. John Holmen
(Oak Ridge National Laboratory)

Mentor: Yvonne Phillips
(Morehouse College)

- Target Course(s): CS425 Parallel Programming and Algorithms, CS457 Wireless and Mobile Security
- Goal:
  - Integrating HPC with Cybersecurity, Cryptography, and Machine Learning to develop curriculums
  - Identify applicable HPC resources from ORNL/wider HPC community and develop course descriptions
  - Create and refine course schedules, hands-on labs, etc.
- GitHub Repo: https://github.com/ruikobe/KetteringTeamFacHack23
- Theme Song: George Thorogood & The Destroyers - Bad To The Bone

Syllabus

GitHub link: https://github.com/ruikobe/KetteringTeamFacHack23/blob/main/CS425%20Syllabus.pdf

CS-425 Parallel Programming and Algorithms

The CS-425 course introduces you to the foundations of parallel computing including the principles of parallel algorithm design, analytical modeling of parallel programs, programming models for shared- and distributed-memory systems, parallel computer architectures, along with numerical and non-numerical algorithms for parallel systems. The course will include material on emerging multicore hardware, shared-memory programming models, message passing programming models used for cluster computing, data-parallel programming models for GPUs, and problem-solving on large-scale clusters using MapReduce. A key aim of the course is for you to gain a hands-on knowledge of the fundamentals of parallel programming by writing efficient parallel programs using some of the programming models that you learn in class.

Prerequisites: CS-401 Programming Methods for Data Science
CS-231 Programming Paradigms

Class Schedule: 240 minutes of lecture

Textbooks

- Multicore and GPU Programming: An Integrated Approach 2nd Edition, Gerassimos Barlas (2022)

Sample Exercises

1. Create accounts and get access to Kettering University High Performance Computer Cluster (KU-HPC). We will be using these machine for all our assignments and projects.

2. Compile hello_mpi.c and submit a job on both machines.

3. Write a simple MPI program that does the following: generate a random integer array, with 100 elements per process, on all processes. hint: initialize the RNG using some function of the rank of the process.

Potential Science Gateways

https://catalog.sciencegateways.org/#/home/8895729113541766680-242ac119-0001-012?pageName=Jupyter%20Notebook

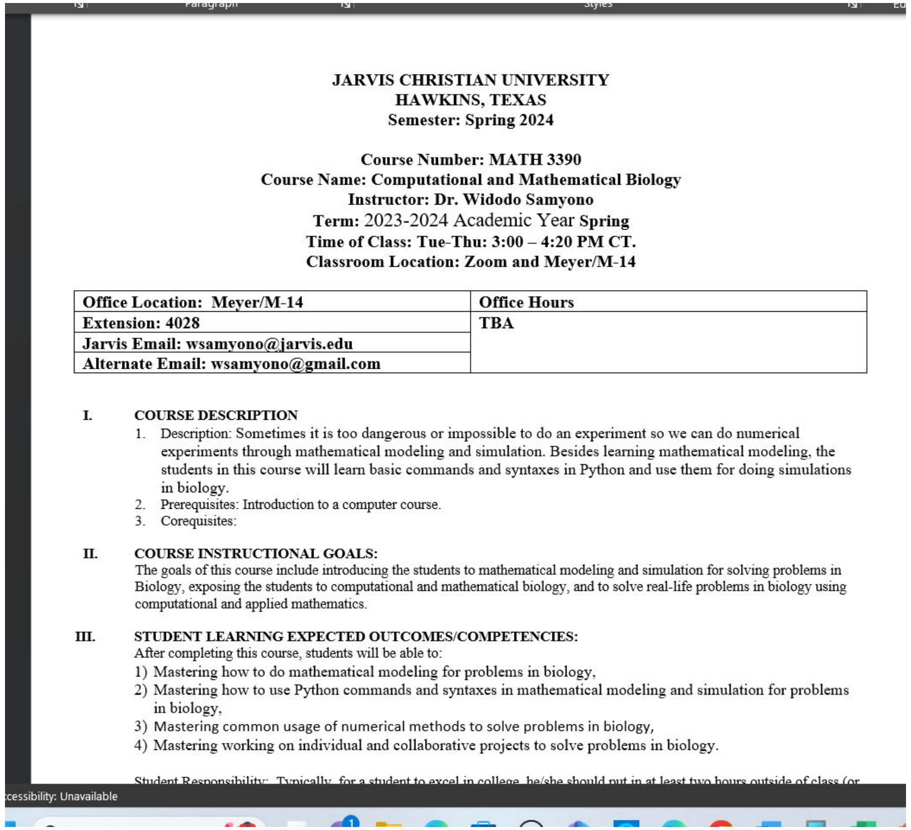
https://catalog.sciencegateways.org/#/home/2727092443435831786-242ac116-0001-012?pageName=SMALTR%20Gateway

https://catalog.sciencegateways.org/#/home/8870203472310365720-242ac11c-0001-012?pageName=CILogon

https://catalog.sciencegateways.org/#/home/7684577338378415640-242ac11d-0001-012?pageName=Center%20for%20Applied%20Internet%20Data%20Analysis%20(CAIDA)

Jarvis Bulldog Team

Syllabus https://github.com/wsamyono/BulldogTeamFacHackGA23/blob/d57bd2fa4c3138ea5ca917ae4cedd4927b4c8ebc/Spring%202024%20MATH%203390%20Course%20Syllabus%20WSamyono.docx



JARVIS CHRISTIAN UNIVERSITY
HAWKINS, TEXAS
Semester: Spring 2024

Course Number: MATH 3390
Course Name: Computational and Mathematical Biology
Instructor: Dr. Widodo Samyono
Term: 2023-2024 Academic Year **Spring**
Time of Class: Tue-Thu: 3:00 – 4:20 PM CT.
Classroom Location: Zoom and Meyer/M-14

Office Location: Meyer/M-14	**Office Hours**
Extension: 4028	**TBA**
Jarvis Email: wsamyono@jarvis.edu	
Alternate Email: wsamyono@gmail.com	

I. COURSE DESCRIPTION
1. Description: Sometimes it is too dangerous or impossible to do an experiment so we can do numerical experiments through mathematical modeling and simulation. Besides learning mathematical modeling, the students in this course will learn basic commands and syntaxes in Python and use them for doing simulations in biology.
2. Prerequisites: Introduction to a computer course.
3. Corequisites:

II. COURSE INSTRUCTIONAL GOALS:
The goals of this course include introducing the students to mathematical modeling and simulation for solving problems in Biology, exposing the students to computational and mathematical biology, and to solve real-life problems in biology using computational and applied mathematics.

III. STUDENT LEARNING EXPECTED OUTCOMES/COMPETENCIES:
After completing this course, students will be able to:
1) Mastering how to do mathematical modeling for problems in biology,
2) Mastering how to use Python commands and syntaxes in mathematical modeling and simulation for problems in biology,
3) Mastering common usage of numerical methods to solve problems in biology,
4) Mastering working on individual and collaborative projects to solve problems in biology.

Student Responsibility: Typically, for a student to excel in college, he/she should put in at least two hours outside of class (or

Team Member:
Widodo Samyono,
Jarvis Christian University

Team Mentor:
Je'aime Powell,
TACC

Team Theme Song: Hey Bulldog, The Beatles, https://www.youtube.com/watch?v=M4vbJQ-MrKo

Jarvis Bulldog Team

Sample Exercises

colab.research.google.com/drive/1lFrPTGGA05M8FOMBni-o3wltDCAJg1Be?usp=sharing#scrollTo=nLji5FdhUsnQ

JSEP2022Week4.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Exercise 10: The standard FASTA file-format, used to represent protein and nucleic acid sequences, consists of two parts: (i) The first line is a description of the biomolecule, starting with a greater-than sign (>) in the first column; this sign is immediately followed by a non-whitespace character and any arbitrary text that describes the sequence name and other information (e.g., database accession identifiers). (ii) The subsequent lines specify the biomolecular sequence as single-letter codes, with no blank lines allowed. A protein example follows:

```
>tr|Q8ZYG5|Q8ZYG5_PYRAE (Sm-like) OS = P aerophilum GN = PAE0790MASDISKCFATLGATLQDSIGKQVLVKLRDSHEIRGILRSFDQHVNLLLEDAEEIIDGNVYKRGTMVVRGENVLFISPVP
```

Begin this exercise by choosing a FASTA protein sequence with more than 3000 AA residues. Then, write Python code to read in the sequence from the FASTA file and: (i) determine the relative frequencies of AAs that follow proline in the sequence; (ii) compare the distribution of AAs that follow proline to the distribution of AAs in the entire protein; and (iii) write these results to a human-readable file.

Note: You need to download FASTA file.

```
# You may write the code for Exercise 10 below.
# You need to download and save the FATA file in the same directory as the code.
# The link to download the file is this https://www.rcsb.org/structure/7O1T
#
```

Jarvis Bulldog Team

Supporting Gateways

1) SGX3 Science Gateways https://sciencegateways.org/education-training/resources#/
2) Design Safe Gateways https://www.designsafe-ci.org/
3) Texas Advanced Computing Center (TACC) https://www.tacc.utexas.edu/
4) Jetstream2 https://jetstream-cloud.org/index.html
5) ACCESS https://access-ci.org/

Team Tech Tigers

Team Theme Song

i. Song name : Weird Science
ii. Artist : Oingo Boingo
iii. URL Link to the song: https://soundcloud.com/oingo-boingo-official/weird-science-album-version?si=e08c2d1f6ce54be18aa649d1ea08556c&utm_source=clipboard&utm_medium=text&utm_campaign=social_sharing

TEAM TECH TIGERS

SYLLABUS CHANGES

The only change to the Syllabus would be the Course Description and the Course Objectives

Course Description: ECE 101 – Introduction to Electrical & Computer Engineering

This course gives first year students a survey of the field of the electrical and computer engineering. Topics include the different subareas within the electrical and computer engineering field, professional careers for ECE students and the tools that will be utilized throughout the curriculum. The course discusses the curriculum, the available technical electives, basic concepts that will be used in the course of study, and a brief history of computing from the abacus to supercomputers, and professional careers for ECE students.

Course Objectives

Add an objective: Explain the history of computing machines.

TEAM TECH TIGERS

EXERCISES FOR STUDENTS

First Exercise: Get familiar with GitHub using online tutorials to be provided

Second Exercise: Set up a GitHub account and repository

Third Exercise: Create a README file in the repository with background information about the student

Fourth Exercise(?): Create an AWS Educate account and begin learning about cloud services

Fifth Exercise: Get familiar with SSH and the Linux command line using the Bandit online game

TEAM TECH TIGERS

CLOUD SERVICE RESOURCES

GitHub: github.com

AWS Educate: aws.amazon.com/education

Google for Education: edu.google.com

Team Altair

Bernie Boscoe, Southern Oregon University

Team Mentors : Veronica Vergara & Mohamed Elbakary

Team Theme Song: New Order, Thieves like us remix (1987)
https://soundcloud.com/markaymufc/new-order-thieves-like-us-mk-instrumental-cover-kleptomaniac-mix

Goals:

To add a module to an undergraduate Intro to Data Science course that demonstrates how to use Jupyter Notebooks in the cloud, with a large dataset, and if I can, GPUs to train an ML model that would not be possible to do without a GPU-enabled device. Outcomes would be an understanding of accessing cloud interfaces, basic terminal commands, an overview of the Jupyter notebook as both a local and cloud tool, and if possible, how to test if GPUs are being seen. Update: possibly using JetStream2

What I need help with: what resources have Jupyter notebooks with GPU option? How can we all share a space, for example for 25 students? How do I handle accounts? How can we load/make available a dataset for them to access?

https://github.com/bboscoe/gateways23

Team Altair: Syllabus

In addition to pre-existing Course Goals and Outcomes:

Course Goals

– Students will explore an HPC cluster environment via a browser-based platform and terminal, and familiarize themselves with large compute infrastructures for scientific exploration

–Students will use Jupyter notebooks to load a pre-trained machine learning model and evaluate its performance with their own datasets and code.

–Students will develop an intuition for what resources are needed depending on the size and complexity of the data, and how much training is desired

Team Altair: Sample Exercises

1. Taking a Jupyter notebook they ran on Colab that shows GPU detection code.
2. Loading the Cassava data from storage and training their own model using the notebook provided by instructor
3. Downloading model to their Colab or local computer and write code to test their own images

If I can get all of these things to work without issue I will eat my hat

Team Altair: Gateways

ACCESS
TACC
SciServer (JHU)

UNC PEMBROKE

Team Name: ThreatTracker

Computing Tools/Environment

- GitHub (to store code and data) (optional)
- Python 3.8+ with packages (faker)
- Oasis stix2-generator, stix2-validator, stix-visualizer
- Synthetic Data Vault
- MITRE ATT&CK STIX Data

Skills/Knowledge/Abilities

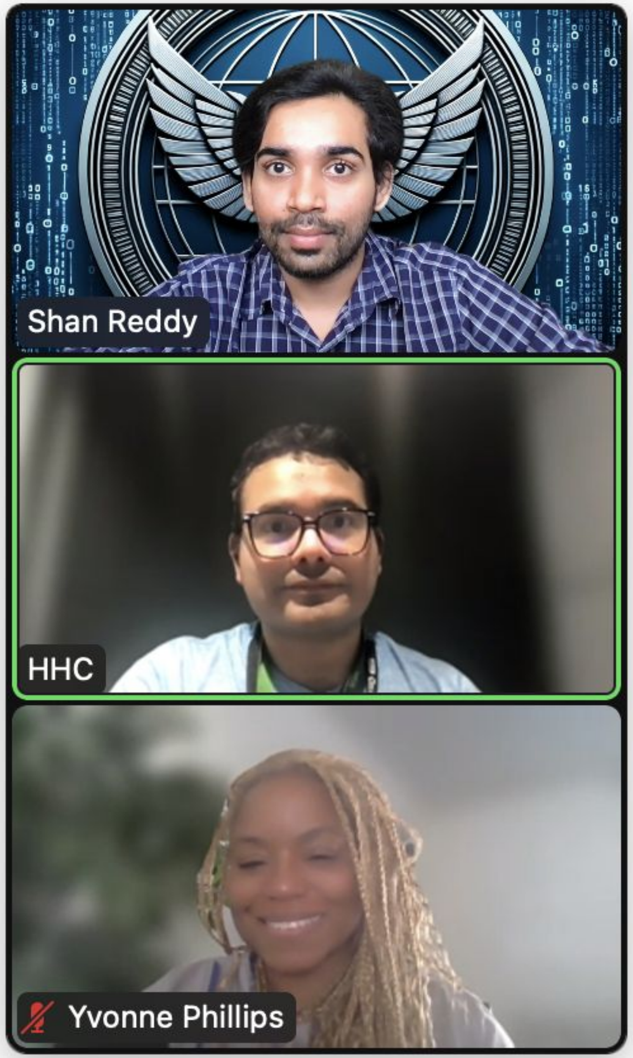
- Python
- Statistics
- Databases
- Basic cyber intrusion knowledge

Course Assessment

- 25% of the overall grade: Create frontend for Identity, Malware, and Threat Actor objects
- 25% of the overall grade: Generate STIX objects from user input, Finish STIX objects and store them in the database
- 25% of the overall grade: Generate/visualize a graph using three STIX objects Identity, Malware, and Threat Actor
- 25% of the overall grade: Anomaly detection using Deep Learning Algorithms.

Theme Song:

https://soundcloud.com/alslyn/synesthesia?in=sc-playlists/sets/brainwaves



FACULTY HACK@GATEWAYS 23

HTTPS://HACKHPC.GITHUB.IO/FACULTYHACK-GATEWAYS23

SGX3 FACULTY HACK GATEWAYS 2023

Supported by:

SGX3 Extend. Expand. Exemplify.

Oak Ridge National Laboratory

omnibond Engineering ▪ Trust ▪ Identity

VOLTRON DATA

TACC TEXAS ADVANCED COMPUTING CENTER

HACKHPC.ORG