Syllabus - 605.202 Data Structures

Instructor Contact	Scott Cost	Eleanor Chlan	Scott Almes	Farhana Shah
	rcost@jhu.edu	echlan1@jhu.edu	salmes1@jhu.edu	fshah7@jhu.edu

There is additional contact information in Blackboard, under Course Contact Info in the left menu. In general, we will respond to emails within 24 hours. Your emails should contain the course number and name in the subject. Include your Discussion Group if relevant. If the matter is URGENT, please include URGENT in the subject and we will strive for a fast turnaround. We will also keep you informed, if any of us expects to be without email access for more than a day. If you need to call an instructor, remember that we are in the Eastern Time Zone and keep your call to between the hours of 9 a.m. and 9 p.m. Eastern time.

Sections 81-87 for Spring 2021 have been combined into a single section in Blackboard. The four of us have divided the responsibility for the course.

Scott C – Homework and Programming Assignments
Farhana – Discussions
Scott A – Office Hours, Discussion Prompts,
Eleanor – Office Hours, course logistics, course developer and coordinator, quizzes, zyBook,
Graders – We will have graders that are assigned by section. Look for Course Contact Info in the left menu of the course.

Feel free to contact any of us. Grade issues should initially be addressed with the person who graded the submission over which you have a concern and then, if needed, it can be elevated.

Course announcements may be used to make corrections or changes to requirements/deadlines. **You are responsible for any announcements made.**

Office Hours via Zoom (Scott A. and Eleanor)

Office Hours will be held via Zoom. The link is given in the left menu of the course website. Recordings will be available under the Office Hours Recordings section of the course website, shown as a link in the left menu. Students are encouraged to email the relevant instructor with any significant questions in advance. Office Hours start with Module 1 and are posted on the calendar and under Getting Started. Instructors reserve the right to terminate office hours if no students are present after 20 minutes past the start time. Instructors also reserve the right to record office hours only if three or more students are present. **Office Hours are scheduled for Thursdays at 9 pm (Scott A), Sundays at 8 pm (Eleanor), and Mondays at 9 pm (Scott A).** *Zoom Information is in Help & Support on the left menu*

Course Description

This course investigates abstract data types (ADTs), recursion, algorithms for searching and sorting, and basic algorithm analysis. ADTs to be covered include lists, stacks, queues, priority queues, trees, sets, and dictionaries. The emphasis is on the trade-offs associated with implementing alternative data structures for these ADTs. There will be four substantial programming assignments. This course will be taught in a language agnostic fashion. Students may choose to develop their work in Java, C++, or Python. (Not for Graduate credit.)

Prerequisites

One year of college mathematics. 605.201 Introduction to Programming Using Java or 605.206 Introduction to Programming in Python or equivalent.

Additional Prerequisites

Students are assumed to be comfortable and proficient in C++, Java, or Python. **Basic programming and object-oriented programming techniques are NOT covered.** You should review of dynamic references and file I/O. **We recommend this course be taken by itself.** If you have significant travel requirements limiting your internet access, you may wish to reconsider taking this course.

Course Goals

At the conclusion of this course, with practical experience garnered from judicious homework assignments. weekly discussions, and programming assignments, students will be able describe the various structures and strategies in the programmer's toolbox to selectively choose the appropriate tools, analyzing the pros and cons of each, to match the tool to the application.

Course Objectives

By the end of this course, students will be able to:

- Accurately and appropriately use Big "O" notation to describe and compare algorithms, while differentiating algorithm functionality from implementation.
- Accurately detail at least two implementations for each standard data structure: stacks, queues, lists, and binary trees; and analyze the pros and cons of each implementation to match applications in programming assignments.
- Explain the recursive paradigm through practice with non-trivial programming assignments and assess practical applications of recursion.
- Analyze the pros and cons of the standard sorting and searching strategies to select an optimal strategy for a specific programming application.

Programming Language

As stated in the course description, this course is being taught in a language agnostic fashion. Students may choose from C++, Java, or Python. You will need to select one of the three languages in Module 0 and stick with it for the entire term.

Textbook

The required textbook is an e-book. In Module 1, click on the "zyBooks Activity 1" link and **subscribe using your JHU email** - *Do not go to the zyBooks website and create a new account.* A subscription is **$68.00** and will last until 5/27/2021. Students will be able to subscribe until May 1, 2021. You must access the text from **within** each module to get the credit posted to Blackboard. While it is preferred that you complete the reading activities within the time frame of the module, for time management reasons you may work ahead and still get the credit.

Optional Textbooks for those that like a real textbook:

- Augenstein, M. J., Langsam, Y., & Tenenbaum, A. M. (1995). Data Structures Using C and C++, Pearson; ISBN-10: 0130369977 OR ISBN-13: 978-0130369970
- Carrano, F., Henry, T., Data Abstraction & Problem Solving with C++: Walls and Mirrors 7th Edition (2016), Pearson. ISBN-10: 0134463978 OR ISBN-13: 978-0134463971
- Augenstein, M. J., Langsam, Y., & Tenenbaum, A. M. (2003). Data Structures Using Java. New York, NY: Prentice Hall.
- ISBN-10: 0130477214 OR ISBN-13: 978-0130477217
- Prichard, J. J., & Carrano, F. M. (2010). Data abstraction and problem solving with java: Walls and Mirrors (3rd Ed.). Boston, MA: Addison Wesley. ISBN-10: 0132122308 ISBN-13: 978-0132122306
- Miller, B.N., Ranum, D.L., (2005) Problem Solving with Algorithms and Data structures Using Python. Franklin Beedle and Assoc; ISBN-10: 1590280539 OR ISBN-13 : 978-1590280539

References

Under Syllabus and Course Information, you will find a section with additional references.

Required Software

You will need the following software:

- A compiler in C++, or Java or Python. Python Users must compile their work, even though it is an interpreted language.
- An (optional) IDE on your computer to facilitate developing programming assignments. The following are freeware IDEs you may wish to try:
    - Eclipse (http://www.eclipse.org/downloads/)
    - JCreator (http://www.jcreator.com/download.htm)

    - NetBeans (http://www.netbeans.org/)
      IntelliJ (https://www.jetbrains.com/idea/)
- Word processing and spreadsheet applications, e.g., Microsoft Office or an open-source alternative like OpenOffice.org. http://download.openoffice.org/other.html#en-US
- Software for "zipping" and "unzipping" files. Two popular options are:
    - For Windows - WinZip at http://www.winzip.com/
    - For both Windows and Macintosh - Stuffit at http://www.stuffit.com/
- Software that makes PDFs, since everything must be submitted as a PDF. Use the word processing software of your choice, then convert to a PDF. Use LaTeX, Adobe Acrobat, or CutePDF (http://www.cutepdf.com/). Open Office (Libre Office) also has PDF creation.

Mathematical Symbols

At various points during the semester, you may need to submit work incorporating mathematical symbols. There are several options for dealing with this.

- Use equation editing software in a PDF source document (e.g. LaTeX, or MS Equation)
- Write in Html using any standard Html editor (e.g., Frontpage, or Mozilla Kompozer (http://www.kompozer.net/). On an HTML page you can use Unicode Entity Codes for Math to get the symbols. https://sites.psu.edu/symbolcodes/accents/math/mathchart/ Unicode Entity Codes for Math can be used to enter symbols directly into text boxes on the Blackboard site. For example, they can be used in the test tool if answers require symbols.
- You can write by hand (or type and just write in the symbols by hand), scan the paper, and submit as an attachment. **Photographs of documents will not be accepted and will not be graded.**

Computer System

This course assumes you have access to a personal computer or laptop to use for programming during the course. It needs to be a machine on which you have sufficient administrator access that you can install software, as needed. The operating system can be Windows (recommended), Linux or MAC OS.

NOTE: Problems with the computer system that you have chosen to use for developing your programming assignments do NOT constitute a legitimate excuse for late assignments. You are strongly encouraged to have backup arrangements in the event of a drive failure, power problems, or other issues that limit your access.

Student Coursework Requirements

It is expected that each module will take approximately 10-15 hours per week to complete. Here is an approximate breakdown: reading the assigned sections of the zyBook and doing the interactive exercises (approximately 1-2 hours per week, listening to the audio annotated slide presentations (approximately 3-4 hours per week), and taking the quizzes (approximately 1 hour, including review), participating in the discussions (approximately 2 hours per week) and doing assigned Homework and working ahead on programming assignments (approximately 3-5 hours per week). During the modules in which we focus on the programming assignments, you may need to spend 10-15 hours or more on the course. Additional time beyond that may be required if you have limited or out-of-date programming experience. **You must be able to commit time during the week for this course. You will not be able to do everything you need to do solely on the weekends.** This course will consist of five basic student requirements:

- **Homework Assignments** (200 points – Scott C)

  There will be assigned weekly homework. The number of assigned problems varies each week since the usefulness of written homework varies with the content. Solutions will be released two days after the end of the module. Solutions are found under the 'Homework and Labs' link in the left menu.

  Problems are selected to emphasize aspects of the lecture material, to cover points omitted in lecture and to prepare students for the exams. It is in your own best interest to do the homework. Homework problems are not intended to be executed, even if the problem asks for code**. Write pseudo-code. The important aspect of the homework is the logic.** You are responsible for including the appropriate level of detail.

The Homework grading scheme awards full points if a reasonable, in-depth, and sincere effort is made on each problem. We reserve the right to reduce the points otherwise.

- 4 points - a reasonable, in-depth, and sincere effort (only trivial errors permitted)
- 3 points - a sincere effort but lacks depth and may have minor errors
- 2 points - a minimal effort with minor errors
- 1 point - a minimal effort, with significant errors

Include your name as part of the file name, e.g., JSmithHW1.pdf, and include your name inside the PDF file.

- **Programming Projects** (400 points – Scott C.)

There will be four assigned programming assignments (labs), each worth 100 of the total grade. All programming assignments will consist of C++, JAVA, or Python programs coded and executed by the individual student. Collaboration is not permitted. Late programming assignments will be penalized 5 points for each day late. **Assignments more than 1 week late cannot be accepted**, except by prior arrangement with Scott C. **Programs which do not produce some form of legitimate output cannot be accepted for grading.** Programs are expected to read and write from/to named files. Please see the Programming Assignments Guidelines located in the Programming Resources section of the course website for additional information on the programming assignments and how they will be graded. **Help yourself do better by reviewing the Guidelines prior to submitting each programming assignment.**

Programming assignments are graded as follows

- 40% - Correctness - In problem solution and results.
- 20% - Style/Proper Coding - Following a reasonable, consistent style with STRONG documentation, with appropriate use of structures, modularity, error checking, etc.
- 10% - Input/Output - Labeled, formatted, correct use of prompts, correctly handles specified inputs and outputs as well as additional cases provided by the student, is user friendly.
- 20% - Analysis - See more detail in the Programming Assignments Guidelines
- 10% - Enhancements - Recognition of superior work on one of the required aspects of the assignments or work above and beyond the requirements. If you add an extra feature to your code, make sure it is "in addition to", not "in place of" a required component of the problem. If you earn more than 10 points on Enhancements, they will be noted as extra credit. Extra Credit is capped at 2 points on each lab.

We expect that you can already write minimal, working code. **IMPORTANT:** You are expected to do your own work. Help from other sources must be acknowledged. It is okay to discuss the problem with others for perspective or to make sure you understand it correctly, but the code you write must be your own. Downloading code from other sources for the programming assignments while strongly discouraged should absolutely be properly accredited. It is prohibited except for the Lab 4 on Sorting.

The labs tend to be time-consuming and you are encouraged to start them as early as possible. Even if we have not yet covered the material you need to know to write the central piece of the code, you can often do other things, like write drivers, or set up your I/O. You are not permitted to use libraries other than I/O or math functions. In particular, you are not allowed to use vectors or ArrayLists.

If you are disappointed in your programming assignment score, with a score below 75%, you may fix and resubmit it before Module 12 in the summer or before Module 14 in the fall and spring terms. Submissions after that must be arranged on a case-by-case basis with Scott C. Fixes will only be accepted on correctness, modularity, and use of named files. Fixes will not be accepted on documentation, input sets, error checking, enhancements, or the Analysis. Fixes are subject to a 30% penalty in fairness to those with a correct original submission, e.g., if you fix something which had a ten-point deduction you will only get back 7 points.

- **Preparation and Participation** (250 points)

Class Discussions (100 points - Farhana)

There will be 10 graded discussions (the lowest of 11 is dropped) worth 10 points each. Student participation in class discussions is imperative for a successful online class experience and students are expected to participate in each discussion, using the Discussions forums on the course website. Discussions provide a venue to codify your thinking by

explaining something to others and allow you the benefit of other's perception and experience. The goal is to examine the topic thoroughly, apply critical thinking skills, and gauge your mastery of the material. An acceptable post is 50-100+ words, is properly referenced, if needed, and demonstrates an in-depth grasp of the material. Students must post an original post to each discussion question and reply to at least two of your different classmate's postings, with four posts total for full credit. However, you should feel free to post more often. The rubric emphasizes posting on different days to encourage interaction. There are no specific posting deadlines within the module. All posts will be read by an instructor. Although, it is not realistic to reply to every post, an instructor will post 3 or 4 times for each question. Instructors will post summaries of each discussion thread from the preceding week to close them out. All posts must be made within the scope of the module and within your assigned group. Posts made before the start or after the end of the module will be ignored and are subject to deletion.

Some Notes about Discussions - If you are unfamiliar with discussion forums you may be unsure about exactly what you are supposed to post, so here are some guidelines.

- It is important to remember that discussions are intended to be interactive. So, if you do all your posts at the end, no one can interact with you. Similarly, if you do all your posts up front, others can respond, but you need to come back in and respond to them as well. In volleyball, it is not enough for someone to serve and someone to hit it back, you must keep the ball in the air and not let it drop.
- A quality post should address the question posed, but it does not have to be the end-all and be-all of responses. Leave something for others to say. Your response, especially follow-on responses, can include one or more of the following:
    - Providing additional information to the discussion.
    - Elaborating on previous comments from others.
    - Presenting explanations of concepts or methods to help fellow students.
    - Presenting reasons for or against a topic in a persuasive fashion.
    - Sharing your own personal experiences that relate to the topic.
    - Providing a URL and explanation for an area you researched on the Internet. If you use something from the internet, or other resource, without citation then it will be construed as plagiarism, you will get a zero on the discussion, and you will be reported to the Office of the Associate Dean.

- It is fine to say that you agree (or disagree) with someone, but it is important to say why and to give supporting arguments. This shows you understand the subject and that you are thinking about it. One of the primary purposes of discussion is to help you take ownership of the material and to get you to think about it more deeply.
- Students are encouraged to be creative in thread titles. **You must include your name in the thread.** For example, "Sometimes it is good to be last - Jane" or "Call me LIFO - Harry"
- If you have difficulty composing a written post, consider experimenting with a voice to text application, which will allow you to compose out loud.
- Finally, of course, you want to phrase all your posts in a positive and constructive manner, the way you would want responses to your posts to be.

Discussion Rubric (Farhana)

Students will be evaluated both on the quality and quantity of their discussion postings. See the rubric below. See the full Discussion Rubric under Syllabus and Course Information.

Relevance/ Understanding (3.5 points) - Posts are consistently in depth and detailed, with new, insightful ideas clearly connected to the topic, demonstrating a command of the discussion topic. High quality posts may include references, quotes, links, or diagrams or images

Cross-Section (3 points) - Posts to at least two threads other than own. Starts own thread.

Timeliness (3.5 points) - Posts throughout discussion (on three different days) and posts often (four or more posts).

Instructors reserve the right to deduct points for not meeting all the discussion requirements. Discussions found to be plagiarized will be assigned a score of zero and the zero grade will NOT be applied as the dropped discussion.

Prompts (50 points – Scott A.)

Lecture material will be accompanied by discussion prompts to help you prepare for the discussion. You make notes as you go through the readings and lectures, then write up your answers to turn in. The scoring emphasizes completion effort, not correctness, but your efforts will prepare you to participate more richly and fully in the discussion. It is to your benefit to approach these sincerely. You will do the prompts during the week of the material and have the actual discussion the

following week in many cases. All prompts should be **submitted as a PDF** attachment. Please include **your name as part of the file name,** e.g. JSmithHW1.pdf, and include your name inside the file. Please submit your answers as a numbered list and include ONLY the answers to the questions. Do not include the question as it will trigger the anti-plagiarism software. If you are found to plagiarize from other sources, you will get a zero on your submission, with no make-up. **This requirement is not about right or wrong answers, but about getting you to think about the material.**

Prompt Rubric – Each prompt is 1 point. The number of prompts per module will vary.

- 1 point – a reasonable and sincere effort
- 0 points – no effort
- Instructors reserve the right to assign partial credit for a weak, minimal response

Online Text Interaction (100 points – Eleanor)

The final piece of the participation grade is completing the interactive exercises in the online textbook. While it is preferred that you complete the reading activities within the time frame of the module, for time management reasons you may work ahead and still get the credit. You must access the zyBook though the Link in the Blackboard Module in order to get the credit posted back to Blackboard. Parts of the zyBook marked optional will not be scored, but you are encouraged to complete them. **You can track your zyBook performance on the zyBook website.** It will be adjusted at the end of the semester as a percentage applied to 100 points.

- **Quizzes** (150 points – Eleanor)

Each module of the course content has one or more quizzes items in the module. Students are expected to complete these each week during the associated module to gauge their mastery of the material and identify areas needing additional study and review. **These questions are scored by the computer.** Quizzes are weighted proportionally. Quizzes will remain available until the end of the semester for study and review purposes. You will get up to two attempts for each quiz. **The recorded score will be the average of the two attempts**. You may not always want to use both attempts.

Due Dates

A note about due dates: **All times are specified in terms of EST/EDT, local to the instructors.** Final Discussion posts, prompts, zyBooks, homework, quizzes, and programming assignments for the module are always due on Tuesday 11:59 PM. See the calendar and course outline. **LATE work will not be accepted, except by prior arrangement.** Submissions after the deadline will be accepted by Blackboard but will be marked late. We generally will not quibble over a couple of minute's lateness. Multiple submissions will also be accepted, so if you think of something you forgot to do, a new answer for a question, or if we ask you to redo something, just go ahead and resubmit to Blackboard in the same place as the first submission. *There is a three-hour grace period on the deadline for Homeworks and Labs only.*

Weekly Learning Guide

This is a general learning guide for all the modules except the ones with programming assignments. They will have a separate learning guide. The learning guides are intended to help you stay on top of the course in an organized manner, and to serve as a checklist each week. Both guides are available in Blackboard under the link "Syllabus and Course information" in the left menu.

- Begin the module by viewing the Introduction video if there is one.
- Review the Discussion Prompt Questions for the Module.
- Continue by skimming the assigned chapters of the textbook.
- Jot down responses to the Discussion Prompt Questions as you go through the text.
- Review the Homework assignment, if any, to see if you need to ask clarification questions.
- Attend the first Office Hour for clarification on Homework, labs, or content, Thursday 9 pm. This Office Hour will include a walkthrough the assigned Homework.
- Use the posted solution, available after midnight of Day 2 of the module, to figure out what you did wrong on the homework of the last module and try to correct it.
- View the various content videos in the Lectures folder
- Start posting to the discussion forum.
- Jot down responses to the Discussion Prompt Questions as you go through the video.
- Go back to read the relevant chapters in depth.
- Work on the next programing assignment, as time is available

- Please post any questions to the appropriate thread in the Content Questions forum in the Discussions section of the course website.
- Review the Vocabulary list for the module and make sure you know the terms and meanings.
- Take notes throughout the week for questions you may need to ask during Office hours.
- Continue to post to the discussion question for this module.
- Attend Office Hours on Sunday 8 pm.
- Continue to post to the discussion question for this module. Goal: post on 3+ different days.
- Attend Office Hours on Monday 9 pm.
- Continue to post to the discussion question for this module.
- Complete the quiz or quizzes if you have not yet done so.
- Post four times or more total to the discussion question for this module by Tuesday midnight.
- Complete this module by submitting the homework, prompts, zyBook, quizzes, or programming assignment by Tuesday midnight.

Grading

Grades will be based on 1000 points over the semester. All students need to get a B- or higher (800 points or more) in Data Structures to satisfy the prerequisite requirement. Students earning a grade below B- will not be allowed to move forward with the program. As you see from the table, there are some extra credit points available in most of the grade components. We will use a +/- grading system with grades of A+, A, A-, B+, B, B-, C, and F possible, as follows:

	A+	≥ 980 points
980 points	> A	≥ 940 points
940 points	> A-	≥ 900 points
900 points	> B+	≥ 880 points
880 points	> B	≥ 840 points
840 points	> B-	≥ 800 points
800 points	> C	≥ 700 points
700 points	> D	≥ 600 points
600 points	> F	≥ 0

Item	**Note**	**Part of Grade**	**Actual Points Available***
Homework assignments		200	210
Programming Assignments (4)	Equally weighted, but not equally difficult	400	408
Discussion/Participation	Discussions – Lowest discussion dropped (100 points). Prompts – 50 points zyBook – 100 points.	250	100 54 100
Quizzes	Average of up to two attempts per quiz (100 pts).	150	157
Total		1000	1029

*As you can see, approximately a quarter letter grade of built-in extra credit points are available over the semester.

Feedback

When you click on Grade Center, in the left menu, it shows all the assessments for the semester, in a long list. Individual assessments like quizzes, or discussions or labs are listed on the left and the associated score is shown on the right (course averages and medians are shown underneath your score on the right). Items will get filled in as we go through the semester. On the left, under the name of the assessment, it gives the due date, gives the category, and then under that it may have some links to a description, grading criteria, etc. Assessments which use the rubric tool will also have a link to the rubric once it is graded. Click on "View Rubric" and it will show you the score in each category, along with any feedback within the rubric itself. Other kinds of feedback appear as a word balloon on the right next to your score. Check under Syllabus and Course Information for an example image.

Help & Support

Students should refer to Help & Support on the left menu for a listing of all the student services and support available to them. Questions and discussion are encouraged.

Policies and Guidelines

Plagiarism

Plagiarism is defined as taking the words, ideas, or thoughts of another and representing them as one's own. If you use the ideas of another, provide a complete citation in the source work; if you use the words of another, present the words in the correct quotation notation (indentation or enclosed in quotation marks, as appropriate) and include a complete citation to the source. See the course text for examples.

Students should read policies pertaining to academic misconduct and netiquette at http://ep.jhu.edu/genpolguid. Please read below to see how the Academic Misconduct Policy applies to this class. Students will be asked to take an Honor Pledge. **You are expected to do your own work and turn in your own work. Help from other sources must be acknowledged.** It is okay to discuss problems with others for perspective or to make sure you understand it correctly, but the code you write must be your own. Downloading code from other sources for the programming assignments 1, 2, and 3 is not allowed. It is permitted for the programming assignment on sorting (Lab 4) but must be properly attributed. Contact us if you have any questions, no matter how slight, about this policy, or if you have questions about a particular assignment. Students violating this code will receive a zero on the plagiarized work, e.g., plagiarizing a homework problem will result in a zero on the entire homework assignment. Plagiarism will be reported to the Associate Dean. Subsequent incidents will result the student failing the course.

Academic Integrity

Academic Misconduct Policy

All students are required to read, know, and comply with the Johns Hopkins University Krieger School of Arts and Sciences (KSAS) / Whiting School of Engineering (WSE) Procedures for Handling Allegations of Misconduct by Full-Time and Part-Time Graduate Students.

This policy prohibits academic misconduct, including but not limited to the following: cheating or facilitating cheating; plagiarism; reuse of assignments; unauthorized collaboration; alteration of graded assignments; and unfair competition. You may request a paper copy of this policy at this by contacting jhep@jhu.edu

Policy on Disability Services

Johns Hopkins University (JHU) is committed to creating a welcoming and inclusive environment for students, faculty, staff and visitors with disabilities. The University does not discriminate on the basis of race, color, sex, religion, sexual orientation, national or ethnic origin, age, disability or veteran status in any student program or activity, or with regard to admission or employment. JHU works to ensure that students, employees and visitors with disabilities have equal access to university programs, facilities, technology and websites.

Under Section 504 of the Rehabilitation Act of 1973, the Americans with Disabilities Act (ADA) of 1990 and the ADA Amendments Act of 2008, a person is considered to have a disability if c (1) he or she has a physical or mental impairment that substantially limits one or more major life activities (such as hearing, seeing, speaking, breathing, performing manual tasks, walking, caring for oneself, learning, or concentrating); (2) has a record of having such an impairment; or (3) is regarded as having such an impairment class. The University provides reasonable and appropriate accommodations to students and employees with disabilities. In most cases, JHU will require documentation of the disability and the need for the specific requested accommodation.

The Disability Services program within the Office of Institutional Equity oversees the coordination of reasonable accommodations for students and employees with disabilities and serves as the central point of contact for information on physical and programmatic access at the University. More information on this policy may be found at Disabilities Services website or by contacting (410) 516-8075.

Disability Services

Johns Hopkins Engineering for Professionals is committed to providing reasonable and appropriate accommodations to students with disabilities.

Students requiring accommodations are encouraged to contact Disability Services at least four weeks before the start of the academic term or as soon as possible. Although requests can be made at any time, students should understand that there may be a delay of up to two weeks for implementation depending on the nature of the accommodations requested.

Requesting Accommodation

New students must submit a Disability Services Graduate Registration Form along with supporting documentation from a qualified diagnostician that:

- Identifies the type of disability
- Describes the current level of functioning in an academic setting
- Lists recommended accommodations

Questions about disability resources and requests for accommodation at Johns Hopkins Engineering for Professionals should be directed to:

EP Disability Services
Phone: 410-516-2306
Fax: 410-579-8049
E-mail: ep-disability-svcs@jhu.edu