|  |  |
| --- | --- |
|  | **Course Syllabus (v 1.3)** |

**CSCI 471 – Complexity and Computability**

**Fall 2018**

Class Times: 4:00 – 4:50 pm, Monday, Wednesday, Friday

Class Location: 7e222

Instructor: Ben Tyler

Office: 7e423

Office hours: by appointment

Email: [btyler@nu.edu.kz](mailto:btyler@nu.edu.kz)

Instructor: Sain Saginbekov

Office: 7e419

Office hours: TBD

Email: [sain.saginbekov@nu.edu.kz](mailto:sain.saginbekov@nu.edu.kz)

**Pre-Requisites**

This course is intended for fourth-year Computer Science students, and will be specifically required of all CS majors in the future. Students must have passed Discrete Mathematics with a C- to take this course. Mathematically-minded students who are planning on getting a minor in CS should also find this to be an interesting course, as we will discuss some of the theoretical foundations of computing.

**Course Overview**

The course covers the design and analysis of algorithms, and investigates the practical and theoretical limits of computing. We begin by exploring ways to develop algorithms for solving common problems, and use asymptotic analysis to quantify their efficiency. Topics such as searching and sorting, Greedy algorithms, divide-and-conquer techniques, dynamic programming, recurrence relations, and graph-based algorithms may be covered. The second part of the course will introduce the concepts of Turing machines and computability, and we will investigate ways to demonstrate (un-)decidability and (non-) recognizability of languages.

**Learning Outcomes**

By the end of this course,

* Students will become familiar with fundamental approaches used to solve computational problems, and how they can be implemented in an efficient way.
* Students will learn how to perform algorithmic analysis for estimating worst-case and average-case time complexity for basic algorithms.
* Students will be able to explain the Church-Turing Thesis and its significance to the field.
* Students will be able to explain why the Halting Problem has no algorithmic solution.
* Students will be able to provide examples of functions that are not computable, and prove that a problem is not computable through reduction.

**Course Materials**

There is no formal textbook for the course, though we will refer to several books throughout the semester, including:

* Introduction to Algorithms (Cormen et al)
* Introduction to the Theory of Computation (Sipser)
* Data Structures and Algorithm Analysis (Shaffer) – Provided in Moodle
* Algorithms (Sedgewick)

Outside of lecture materials, the course content will consist of online materials, handouts, tutorials, and scholarly articles – these will be provided via Moodle.

**Class Structure**

For this course, we will be using an approach that balances conceptual learning, group problem solving, and active participation by the students. Many of the class periods will involve a brief lecture, and then a period of problem solving and in-class exercises related to the day’s lesson. We will often require students to present solutions to such exercises on the board, which will count towards their “active participation” component of their grade.

This semester, we will also have several homework assignments which should be physically submitted to the TA at the beginning of class on the day they are due. Homework must be readable – if you have poor handwriting, type up your solutions. If we can’t read it, we will assume your answers are wrong. Also, if you don’t know the answer for a particular problem, you can receive 25% for that problem by simply stating “I don’t know” in place of your answer.

Students may also be required to come in for “live grading”, and demonstrate that they actually understand their work to the instructor or TA. When live grading is required, students will have to schedule a time to meet with the instructor or TA.

During the semester, each student will be required to do a 15-minute presentation, where the student picks a particular problem solution or algorithm (subject to prior approval by the instructor), and presents it to the class. Students should understand what they are talking about well enough to answer questions posed by their classmates and instructor.

Three quizzes will be given during the semester. The first two will be in-class quizzes that should be 55 minutes in length. They will be paper-based and you will not be allowed to use books, online resources, or notes – just your pen or pencil, and an eraser. The third quiz will be a take-home quiz that will be due during the exam period; you are not allowed to collaborate with others on the quiz, and you will get a 0 if you use answers obtained from online, textbooks, or other resources. Other than in exceptional cases, you will not be allowed to make up a quiz—you will be responsible for informing your instructor immediately in such circumstances where you believe a make-up is warranted.

This semester, there will not be a final exam for the course.

**Course Assessment**

The final grade is calculated as follows:

* Participation 5%
* Homework Assignments 20%
* Quizzes 25%
* Midterms 50%

Final letter grades will be assigned using the following:

A 95 or above

A- 90 up to 95

B+ 85 up to 90

B 80 up to 85

B- 75 up to 80

C+ 70 up to 75

C 65 up to 70

C- 60 up to 65

D+ 55 up to 60

D 50 up to 55

F 0 up to 50

**Course Policies**

**Attendance**

Missing classes and habitual tardiness will have a negative effect on your grade, both directly and indirectly. Attendance will be recorded on Moodle, and will count towards your grade. You may check your current number of absences at any time. We will be covering a lot of material over the semester, and will be having numerous class exercises which will require your direct participation to receive credit.

**Electronic resources**

Unlike many of your other CS classes, the focus of the work you do will not be on programming, but rather, resemble problem solving in an intermediate-level math course. We really do not need to conduct the course in the computer lab, though we may provide some computer-driven demonstrations to explore some of the concepts discussed in class.

You are expected to check your Nazarbayev University e-mail on a daily basis for updates and announcements about the course. Since we will be using lots of graphs and diagrams in our solutions, homework’s should be written by hand, and submitted physically to the instructor (not Moodle). This is also why we will not use Moodle for quizzes.

**Late policy**

Homework assignments *must* be physically submitted to the TA at the *beginning* of class (first 10 minutes) on the day they are due. In general, there is no late policy, other than you get zero points if you don’t submit your homework when it is due.

In cases of illness or family emergency, you must inform your instructor immediately if you believe you will not be able to submit your assignment on time. In such cases, an exception may be made at the discretion of your instructor.

**Classroom behavior**

You are expected to act respectfully towards your fellow classmates, TAs, and instructors inside and outside of the classroom. We have a limited amount of time to cover a lot of material this semester, so you need to pay attention during lectures and presentations, and do your in-class work when it is assigned. Talking on your phone, texting, chatting online, browsing Facebook or other social media sites, and talking excessively with your neighbors about non-class related stuff are just a few examples of behavior that is not acceptable, and will negatively impact your participation score.

**Academic integrity**

Nazarbayev University and The School of Science and Technology have established high standards for academic integrity, using an approach in which students are trained to produce original work according to professional standards, and to properly cite and reference the work of others when it is appropriate to do so.

The specific guidelines are published in the NU Student Handbook. In particular,

* The assignments in this class are designed to introduce important concepts and techniques, and enable you to explore the material independently so as to gain insight and comprehension of the subject. Doing the work is much more important than getting the right answer.
* The course is designed such that the new material presented each day builds on the skills developed in the preceding days; thus, any action that interferes with this process (missing class, skipping the assignment, copying) will seriously impede your progress.
* You are welcome—and encouraged—to talk through concepts and ideas with your fellow students and to study with them, but do not give or receive direct help from your classmates on a graded assignment.
* Assignments should be completed individually. If you distribute your work to others, even if you are not intending them to copy it, this is still considered academic misconduct.
* Even the appearance of cheating or inappropriate copying should be avoided.
* You may only get help on graded assignments from designated people—the instructors or TAs for the course. If you are struggling with an assignment, by all means, please seek help from them.

In the event that academic misconduct such as plagiarism or cheating is discovered, the student will receive no credit for the work, and the event reported to the Dean of your school. Egregious cases, or a second offense, can result in failure of the course and potential suspension or expulsion from the university.

When a student suspects that another student has violated the academic honesty policy, a report should be made to the appropriate faculty member.

**Preliminary Course Outline (Dates and topics are subject to change!)**

Last day to add: Monday, September 11

Last day to drop: Friday, September 15

Last day to withdraw (with ‘W’): Friday, October 20

|  |  |  |
| --- | --- | --- |
| **Week** | **Dates** | **Topics, Activities, Exams** |
| **Week 1** | Aug 13, 15, 17 | Introduction;  Asymptotic Analysis and Big-O Notation |
| **Week 2** | Aug 20, 22, 24 | Recurrence Relations;  Substitution, Recursion tree, and Master methods; Strassen’s algorithm |
| **Week 3** | Aug 27, 29, 31 | Sorting Algorithms; Sorting in Linear Time |
| **Week 4** | Sept 3, 5, 7 | **Quiz1**; Medians and Order Statistics |
| **Week 5** | Sept 10, 12, 14 | Dynamic Programming; Greedy Algorithms |
| **Week 6** | Sept 17, 19, 21 | Graph Algorithms |
| **Week 7** | Sept 24, 26, 28 | (Huffman codes, Network flows) |
| **Week 8** | Oct 1 | Finite Automata (DFAs and NFAs); |
| **Week 9** | Oct 8 | **Fall Break** |
| **Week 10** | Oct 15 | Regular Expressions  Closure Properties of Regular Languages |
| **Week 11** | Oct 22 | Pumping Lemma for Regular Languages; |
| **Week 12** | Oct 29 | Context-Free Grammars (CFGs)  Pushdown Automata |
| **Week 13** | Nov 5 | Pumping Lemma for CFLs |
| **Week 14** | Nov 12 | Turing Machines  Church-Turing Thesis |
| **Week 15** | Nov 19 | the Halting Problem  Non-Recognizability |
| **Exam Period** | Nov 26 |  |