Course Syllabus

Jump to Today ✎ Edit

CSC 384 H1F Introduction to Artificial Intelligence

University of Toronto, St. George Campus, Fall 2024

Table of Contents

Your Professor

Hi! I am Professor Alice Gao. I will be your instructor for this course. My pronouns are she/her/hers. Feel free to call me Professor, Professor Gao or Alice. I will answer to all three.

I am an assistant professor, teaching stream in the computer science department. This year (Sep 2024 to Apr 2025), I will be teaching Intro to AI (CSC384), Intro to ML (CSC311) and Deep Learning (CSC413). If you enjoy this course, please consider taking another one with me! I would love the chance to get to know you better! Besides teaching, I also work on research projects on computer science education, games, and AI. If you want to learn about research, consider taking a CSC494/495 course with me.

Take a look at **my website** (https://www.cs.toronto.edu/~axgao/) for more information on my past teaching experience, research projects, and instructions on how to *request a reference letter* from me (read this now if you are interested in pursuing graduate studies!).

Best Way to Reach Alice

Please do NOT email me regarding course-related matters. Instead, please *post on Piazza*, send an email to *the course email account*, and/or *attend my office hours*. I receive too many emails on various matters (teaching, research, service, etc.). These channels (Piazza and course email account) help me keep my teaching-related and other emails separate and allow me to respond to you in a timely manner.

My office is in **Bahen Center (https://goo.gl/maps/dNEP1wq7vWkDcFi66)** 4240. If my door is open, feel free to come and chat with me.

Your TAs

I am fortunate to have nine excellent teaching assistants working with me this term. They will hold office hours, answer Piazza questions, and invigilate and mark the term tests and the final exam. We will post the TA office hour information on Piazza.

Your TAs are graduate students in computer science. If you are (even remotely) interested in pursuing graduate studies, you might want to chat with them. They should have excellent insights into what graduate school is all about.

Course Description

↑ Contents ↑

In this course, you will learn about the theories and algorithms that capture some core elements of computational intelligence. I have designed this course to achieve the *course outcomes* below.

This course aims *to combine theory and practice*. You will learn about the theory in lectures and demonstrate your understanding of the theory on the term tests and the final exam. Moreover, you will apply the theory to solve realistic problems by completing the programming assignments.

Course Outcomes

1. *Explain* the properties of an artificial intelligence algorithm.
2. *Execute* an artificial intelligence algorithm to solve a problem.
3. *Justify* whether an artificial intelligence algorithm is appropriate for solving a problem.
4. *Formulate* a problem using an abstract representation so that we can solve the problem using an artificial intelligence algorithm.
5. *Design* a significant program to solve a problem using an artificial intelligence algorithm.
6. *Implement* an artificial intelligence algorithm to meet correctness and efficiency requirements.

How Are These Course Outcomes Relevant To You? This course provides a first and shallow introduction to artificial intelligence. It is an excellent starting point for anyone curious about artificial intelligence. In this course, we will teach you some core concepts in artificial intelligence and prepare you to dive deeper into a particular topic in advanced artificial intelligence courses.

Course Topics

Specifically, you will be learning about algorithms in 5 topics:

1. Search
2. Game Tree Search (a.k.a. Adversarial Search)
3. Constraint Satisfaction Problems (CSP)
4. Uncertainty
5. Knowledge Representation (KR)

Recommended Textbook

The course materials will largely follow the recommended textbook below.

Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition.

However, there is no need to purchase this textbook for the course. The lecture slides include all the materials you need to know to succeed in this course. I will post the lecture slides before class and the annotated lecture slides after class.

Course Schedule

↑ Contents ↑

I have compiled a course schedule for you below. This schedule outlines all the lectures, assignment due dates, and term test dates for the whole term.

I highly recommend that you take some time to put all the important dates in your calendar right now. See the **Tips for Success** for more details.

Course Schedule

Monday	Tuesday	Wednesday	Thursday	Friday
Mon, Sep 2 Labour Day		Wed, Sep 4 **Introductions** **(https://q.utoronto.ca/courses/354350/files/3000806?wrap=1)** **(https://q.utoronto.ca/courses/354350/files/3000806/download?download_frd=1)**		Fri, Sep 6 **Definitions of AI and Intro to Search** **(https://q.utoronto.ca/courses/354350/files/3000808?wrap=1)** **(https://q.utoronto.ca/courses/354350/files/3000808/download?download_frd=1)**

Mon, Sep 9 Uninformed Search 1		Wed, Sep 11 Uninformed Search 2		Fri, Sep 13 Heuristic Search 1
Mon, Sep 16 Heuristic Search 2	Tue, Sep 17 A1 Posted	Wed, Sep 18 Games 1		Fri, Sep 20 Games 2
Mon, Sep 23 Games 3	Tie. Sep 24 A1 Early Deadline 1	Wed, Sep 25 Games 4		Fri, Sep 27 **Test 1 Review**
Mon, Sep 30 **Test 1**	Tue, Oct 1 A1 Early Deadline 2 A2 Posted	Wed, Oct 2 Games 5		Fri, Oct 4 CSP 1
Mon, Oct 7 CSP 2	Tue, Oct 8 A2 Early Deadline 1 **A1 Due @ Noon**	Wed, Oct 9 CSP 3		Fri, Oct 11 CSP 4
Mon, Oct 14 **Thanksgiving Day**	Tue, Oct 15 A2 Early Deadline 2	Wed, Oct 16 CSP 5		Fri, Oct 18 CSP 6
Mon, Oct 21 CSP 7	Tue, Oct 22 **A2 Due @ Noon** A3 Posted	Wed, Oct 23 Uncertainty 1		Fri, Oct 25 Uncertainty 2
Mon, Oct 28 **Reading Week**	Tue, Oct 29 **Reading Week**	Wed, Oct 30 **Reading Week**	Thu, Oct 31 **Reading Week**	Fri, Nov 1 **Reading Week**
Mon, Nov 4 Uncertainty 3	Tue, Nov 5 A3 Early Deadline 1	Wed, Nov 6 Uncertainty 4		Fri, Nov 8 **Test 2 Review**

Mon, Nov 11 **Test 2**	Tue, Nov 12 A3 Early Deadline 2 A4 Posted	Wed, Nov 13 Uncertainty 5		Fri, Nov 15 Uncertainty 6
Mon, Nov 18 Uncertainty 7	Tue, Nov 19 A4 Early Deadline 1 **A3 Due @ Noon**	Wed, Nov 20 KR 1		Fri, Nov 22 KR 2
Mon, Nov 25 KR 3	Tue, Nov 26 A4 Early Deadline 2	Wed, Nov 27 KR 4		Fri, Nov 29 KR 5
Mon, Dec 2 Conclusion	Tue, Dec 3 **A4 Due @ Noon**	Wed, Dec 4	Thu, Dec 5	Fri, Dec 6
Mon, Dec 9	Tue, Dec 10	Wed, Dec 11	Thu, Dec 12	Fri, Dec 13
Mon, Dec 16	Tue, Dec 17	Wed, Dec 18	Thu, Dec 19	Fri, Dec 20

Let's take a look at the course components individually.

Lectures

The lecture will occur on Mondays, Wednesdays, and Fridays. The lecture times and locations are given below.

Lecture Times and Locations

Section	Time	Monday	Wednesday	Friday
101/2001	12-1pm	BA 1190	BA 1190	BA 1190
201/2101	2-3pm	MP 103	MP 103	MP 103

Lecture recordings will be generated and posted automatically on the OCCS Student App. *Remember that course videos and materials belong to your instructor and the University and are protected by copyright.* You are permitted to download videos and materials for your own personal academic use, but *you may not copy, share, or otherwise distribute them* without explicit permission from the instructor.

Term Tests

There will be two term tests. Each test covers roughly two of the first four topics. The 2 tests is roughly equivalent to 1 midterm.

The tests will occur *on Mondays during lecture time slots (50 minutes long)*. See the course schedule above for the test dates. The tests will be graded and returned through Crowdmark.

The main purpose of having 2 tests instead of 1 midterm is *to motivate you to study regularly*! See "Study Regularly" in the **Tips for Success** section below.

How do I prepare for tests? The tests contain *theoretical* questions only. There will NOT be programming questions. You can expect two types of questions on the test. The conceptual questions (multiple-choice or short-answer) test your understanding of the properties of the algorithms. The execution questions (long-answer) ask you to show the steps of executing an algorithm to solve a problem. Education research has shown that *the best way to prepare for tests is to solve practice problems (ideally under time pressure).* See "Solve Practice Problems" in the **Tips for Success** section below.

Programming Assignments

As part of this course, you will complete 4 substantial programming assignments. These assignments provide excellent opportunities to solve fun and realistic problems by applying the theoretical concepts learned in lectures. You may find these assignments challenging, but I promise they will be rewarding! These assignments are my favourite part of this course 😄.

The assignments will be *due on Tuesdays at NOON ET on **Markus** **(https://markus.teach.cs.toronto.edu/markus)*. They are to be completed individually.

A Roadmap to Succeeding on the Assignments

In the past, many students found the programming assignments challenging, often due to debugging and testing. As a result, we have updated the programming assignments to provide more support to you while completing them.

- For each assignment, we will provide some starter code which models the problem. This way, you can focus on implementing the algorithms to solve the problem.
- We have broken down the work into several functions. The functions start simple and gradually become more complicated and build on the previous ones. You will need to complete the functions based on the provided function headers.
- We will grade your program by running unit tests on the functions individually. As a result, if you cannot complete the program by the deadline, you can still get partial marks for implementing some functions.

This design provides you with a pathway to success. We have broken down a large task into smaller, more manageable tasks so that each small task does not look intimidating. Ideally, we envision you spending a few hours on the assignment each week to complete the functions in order of increasing

difficulty. Moreover, you can test each function as soon as you complete it rather than waiting to test the entire program.

To encourage you to follow the pathway to success, we have introduced *two early deadlines* for each assignment. We took all the functions you need to implement and divided them into three groups. The first group has the simplest helper functions. The second group has more complex functions that build on the functions in the first group. The third group has the remaining functions. Each assignment is available for three weeks. During each week, we will provide unit tests for the functions in the respective group.

We created the two early deadlines for several reasons: to e*ncourage you to start the assignments early and to help you tackle the assignment incrementally.* To encourage you to meet the early deadlines, for each assignment, meeting both early deadlines is worth 10% of the final assignment grade and the final submission is worth 90% of the final assignment grade.

Surveys

You will complete five short surveys (as Quercus quizzes). The first survey asks for some background information on you and helps us get to know you better. The next four surveys collect information on your experience with the four programming assignments and help us improve the assignments for the future.

Final Exam

The final exam will cover all 5 topics in the course. The questions on the first four topics will likely be more challenging than those on the tests since you already had some practice. Since there is no test on the last topic (KR), we will provide some practice questions to help you prepare for the final exam. I encourage you to use the past exams in the **Old Exam Repository (https://login.library.utoronto.ca/index.php?url=https://exams.library.utoronto.ca)** since doing practice questions is one of the best ways of preparing for an exam.

The final exam schedules will be available in November on **the A&S page (https://www.artsci.utoronto.ca/current/faculty-registrar/exams-assessments/exam-assessment-schedule)** .

Estimated Student Workload

We estimate that you will need to spend an average of 8 hours on this course weekly. A rough breakdown is 3 hours of lectures, 3 hours working on the programming assignments, and 2 hours preparing for the tests.

However, we anticipate you will spend more than 3 hours on the programming assignments each week. See "Tackle Each Assignment Incrementally" in the **Tips for Success** section below.

Seeking Help

↑ Contents ↑

The course staff are ready and eager to help you inside and outside class. Remember that *seeking help is NOT a sign of weakness*. To the contrary, we encourage you to *seek help from us early and often* so that we can help you succeed in this course. You can get in touch with us through the following channels: Piazza, Professor office hours, TA office hours, and course email account.

Piazza

Piazza is one of the best ways to reach the course staff. It gives you the flexibility to ask a question anywhere, anytime. Moreover, all the students in the course can benefit from your conversation with the course staff. We encourage you to post your questions on Piazza publicly. However, please make your post *private* if your question contains *sensitive* information (e.g., hints for assignments/tests or personal information).

The teaching assistants and I will aim to respond to your Piazza posts within 48 business hours. Note that if you post a question outside of business hours, we cannot guarantee responding to your question until the next business day.

We will make *important announcements on Piazza* rather than Quercus.

See the **Tips for Success** on "Check Announcements Regularly" and "Search Before Post."

Professor Office Hours

I have set aside three weekly hours to meet with you. Feel free to chat with me about this course, my experience, or anything else I can help you with. If there are a lot of students (and there will be when a deadline is near =), I will prioritize answering course-related questions.

- In-person office hours:
  - Wednesdays: 3:30-5pm in Bahen 4240
  - Thursdays: 9:30-11am in Bahen 4240 (exceptions: 10-11am on Sep 12 and Sep 26)
- Online office hours by appointment (on Friday mornings)

TA Office Hours

The TAs will hold in-person and online office hours to answer your questions. Each TA will specialize in answering questions for one topic so that they can provide the best help for you. The in-person office hours will happen in the Help Centre (BA 2270). The online office hours will occur on Zoom. The TA office hours will not follow a regular schedule. We will announce the TA office hour information on Piazza, usually two weeks before the office hours take place.

Course email address

csc384-2024-09@cs.toronto.edu (mailto:csc384-2024-09@cs.toronto.edu)

If you have an administrative issue, please send a message to the course email address above.

However, if you have a *remark request* or *special consideration* request, it is sufficient to fill out the respective online form. Please avoid sending us an email unless you must provide information that cannot be included in the form responses.

Course Policies

↑ Contents ↑

Grading Scheme

Please take a careful look at the grading scheme chart below.

Since the final exam covers all the topics in the course, you must obtain *a minimum grade of 40%* on the final exam to *pass* this course. If your final exam grade is less than 40%, the maximum final grade you can obtain for the course will be 47%.

Grading Scheme

Assessment	Percentage
4 Assignments	40%, 10% per assignment. For each assignment, 1% for the two early deadlines and 9% for the final submission.
2 Tests	20%, 10% per test.
Final Exam	39%
5 Surveys	1%

Test Attendance Policy

You must *attend each test in your registered section*. If you must attend the test in another section due to extenuating circumstances, please email us at the course email account.

Special Consideration Policies

↑ Contents ↑

If you need an assignment deadline extension, cannot complete an assignment, or cannot write a test due to extraordinary circumstances beyond your control, please submit a special consideration request as soon as possible. Special consideration will be evaluated on a case-by-case basis and is not granted automatically. Sometimes, we cannot grant you exactly the special consideration you seek.

Special Consideration Policies for Tests:

- If you *miss one test* for approved reasons, we will shift the weights of the tests to the final exam. After missing one test, you have missed valuable opportunities to get feedback and are possibly in danger of failing the course. Therefore, we will require you to make an appointment with your **College**

Registrar (https://registrar.utoronto.ca/list-of-faculty-colleges-and-campus-registrar-offices/) to create a concrete plan for the rest of the term. We *will require confirmation from your College Registrar* that you have met with them and that your plan for the rest of the term is realistic *before* we approve special consideration requests for the second test.

Special Consideration Policies on Assignments:

- *Minor Deadline Extension:* If you require a minor (2-day) deadline extension for any assignment, simply fill out the special consideration request form, and we will grant the request. You may use this policy on as many assignments as you require. We created this policy because we recognize that unexpected problems, illness, and disability-related barriers sometimes make it difficult to submit assignments on time.
- *Major Deadline Extension or Missing an Assignment:* If you require *a major (7-day) deadline extension* or *cannot complete an assignment* for approved reasons, we will grant your requests for *at most two* assignments as long as you provide reasonable supporting documentation. After missing or falling behind on two assignments, you have missed valuable learning opportunities and are possibly in danger of failing the course. Therefore, we will require you to make an appointment with your **College Registrar (https://registrar.utoronto.ca/list-of-faculty-colleges-and-campus-registrar-offices/)** to create a concrete plan for the rest of the term. We *will require confirmation from your College Registrar* that you have met with them and that your plan for the rest of the term is realistic *before* we approve special consideration requests for future assignments.

Special Consideration Policies if you are registered with Accessibility Services:

- Your accommodation letter will allow for an extension of up to 7 full days. However, due to the incremental nature of CS courses, granting such a long extension from the onset may cause you to fall behind and be disadvantaged. As such, we will start by suggesting an initial 3-day extension. We will grant the 7-day extension later if necessary.

Please complete and submit **the special consideration request form (https://forms.office.com/r/iD1CbDYZkD)** . We will aim to respond to you within 48 business hours.

Make sure to *include supporting documentation* with your request. Please read the new **Student Absences (https://www.artsci.utoronto.ca/current/academics/student-absences)** page from the Faculty of Arts & Science carefully. It contains detailed information on the recognized forms of documentation and the circumstances under which you should use the Absence Declaration tool.

Below are some **invalid** reasons to apply for a special consideration request. These reasons are invalid because the circumstances are **neither unexpected nor outside of your control**.

- Heavy course load
- Multiple assignments are due in the same week.
- Multiple tests are scheduled in the same week.
- I need to catch up on missed work.

If you have difficulty managing stress and time, don't hesitate to contact your College Registrar, who can suggest wellness counselling, academic advising, and/or learning strategist services.

Remark Requests

↑ Contents ↑

If you believe there was an error in the marking of an assignment or a test, or if you simply have questions about how we marked your work, you may submit a remark request. We do **NOT accept remark requests in the first 24 hours after the grades are released**. The reason is that you should spend this time reading and understanding your assessment results, the sample solutions and the marking scheme.

Below are some examples of **valid** reasons for requesting a remark.

- My answer was marked incorrectly based on the marking scheme.
- There was an error in adding up the marks.
- I should get more marks based on the marking scheme.
- I interpreted the question differently, which caused my answer to differ from the sample solutions.

After 24 hours, you will have **one week** to submit a remark request by filling out the appropriate form below. Please provide a detailed justification --- this will help us process your request efficiently. We will process all the remark requests after the one week is over.

Remark Request Forms

Course Component	Form Link
Term Test	**remark request form for term tests (https://forms.office.com/r/5kGPrWxbyd)**
Assignment	**remark request form for assignments (https://forms.office.com/r/FQrZg0YE9X)**

Academic Integrity

↑ Contents ↑

All work you submit must be your own. It is an academic offence to copy the work of someone else *unless you explicitly and clearly attribute the work to its source*. This includes words, sentences, entire documents, and even ideas. Whether you copy or let someone else copy, it is an offence. Academic offences are taken very seriously and can have correspondingly serious consequences.

At the same time, we want you to benefit from working with other students. For the programming assignments in this course, you cannot submit the same code as another student. However, you can

discuss how to solve the problems with anyone you wish. The purpose of the assignments is to allow you to practice implementing an algorithm to solve a real problem. Even if you did not figure out all the implementation yourself, you could still receive full credit for writing up a program *with a list of ALL sources you consulted*: textbooks, web pages, students with whom you discussed the problem, etc. Include all the citations in the Python files that you submit.

You are also welcome to discuss course material and technology related to assignments with each other, and we encourage you to do so. For example, you may work through examples that help you understand course material or new technology or help each other configure your system to run a supporting piece of software.

Please take a few minutes to consult the **Academic Integrity at U of T (https://www.academicintegrity.utoronto.ca/)** website: it contains good information and concrete strategies to help support your learning in ways that follow the principles of academic integrity, in addition to references to formal policies and procedures.

Generative AI Policies

In this course, you may use generative artificial intelligence (AI) tools (like ChatGPT and GitHub Copilot) as learning aids and to help complete the programming assignments. *You will NOT be permitted to use generative AI on the term tests or final exam.* While some generative AI tools are currently available for free in Canada, these tools have not been vetted by the University of Toronto. They may not meet University guidelines or requirements for privacy, intellectual property, security, accessibility, and records retention. Generative AI may produce content that is incorrect, misleading, or inconsistent with the expectations of this course. They may even provide citations to sources that don't exist — and submitting work with false citations is an academic offence. These tools may be subject to service interruptions, software modifications, and pricing changes during the semester.

Generative AI is NOT required to complete any aspect of this course, and we caution you not to rely on these tools to complete your coursework. Instead, we recommend treating generative AI as a supplementary tool only for exploration or drafting content — **always remembering to cite any resource you used to generate your answers**. Ultimately, you (and not any AI tool) are responsible for your learning in this course and for all the work you submit for credit. It is your responsibility to evaluate the content generated critically and to assess your learning independent of generative AI tools regularly. Overreliance on generative AI may give you a false sense of how much you have learned, which can lead to poor performance in this course, later courses, or your future career.

Tips for Student Success

↑ Contents ↑

Tips for Student Success

Title	Details
Put Important Dates In Your Calendar at the Beginning of the Term	I highly recommend that you take a few minutes to record the important dates (such as test dates and assignment due dates) in your calendar. Also, I recommend you do so for all the courses you are taking this term. Once you have completed this, take a careful look at your calendar. Do you have any test times or assignment due dates that are conflicting or close to each other? I suggest planning for these times in advance so you can complete these assessments to the best of your abilities without leaving the work to the last minute.
Check Announcements Regularly	We will make important announcements on Piazza. I prefer Piazza to Quercus for announcements since Piazza provides a way for you to respond to the announcement immediately. I strongly suggest you check Piazza a few times each week to ensure you know the important announcements.
Search Before Post	Before posting a question on Piazza, please search to see if other students have asked similar questions. Doing this is especially important when the number of questions skyrockets near an assignment deadline. If a student has asked a similar question already, we would prefer that you post a follow-up question rather than creating a new post.
Study Regularly	I recommend viewing the tests as learning opportunities rather than considering them exams. The purpose of having more frequent tests is to *motivate you to study regularly.* The tests are designed to be *low-stakes* and *low-stress* opportunities to *get feedback* on your learning.
Solve Practice Problems	The tests contain *theoretical* questions only. You can expect two types of questions: conceptual and execution. The conceptual questions will mostly test you on the properties of the algorithms, whereas the execution questions will ask you to solve a problem by executing an algorithm. For conceptual questions, I recommend summarizing the algorithm properties in your own words using flashcards or concept maps. For execution questions, I recommend practising executing an algorithm. Education research has shown that the best way to prepare for a test is to solve practice problems. I recommend practising under time pressure. Doing this will ensure that you can solve the problems correctly and reasonably quickly.
Tackle Each Assignment Incrementally	We anticipate you will likely spend more than 3 hours on the programming assignments each week. A programming project often comes with unexpected debugging and testing issues, and it is difficult to estimate the time required to solve these issues. Fortunately, each programming assignment will be available for three weeks. We strongly recommend that you plan to work on the

Title	Details
	programming assignments for a few hours every week to account for unexpected testing and debugging issues.

Student Support Resources

↑ Contents ↑

UofT Mental Health Portal (https://mentalhealth.utoronto.ca/)

If you go into Explore Our Care Model, One-On-One Sessions, View Services and Resources, you will get to **Mental Health Clinical Services** (https://studentlife.utoronto.ca/service/mental-health-clinical-services/) . The first option allows you to book **same-day counselling appointments** (https://studentlife.utoronto.ca/service/same-day-counselling-appointment/) . Book an appointment by calling Health & Wellness at 416-978-8030 (select option 5).

If you are in a crisis, UofT Telus Health Student Support (formerly U of T My SSP) provides *real-time, confidential*, *24-hour* support for any school, health, or general life concern at no cost to you. Call 1-844-451-9700 or 001-416-380-6578 (if outside of North America).

You can also contact the Good2Talk Student Helpline (Call 1-866-925-5454 or text GOOD2TALKON to 686868).

Accessibility Services (https://studentlife.utoronto.ca/service/accessibility-services-registration-and-documentation-requirements/)

Check out the **Accessibility Services registration & documentation requirements – St. George Campus** (https://studentlife.utoronto.ca/service/accessibility-services-registration-and-documentation-requirements/) . Register with Accessibility Services before 5 p.m. on Friday, October 13, 2023, for final assessment accommodations.

Student Life Portal (https://studentlife.utoronto.ca/)

Check under **Health and Wellness** (https://studentlife.utoronto.ca/department/health-wellness/) for a range of programs and services.

Take a look at the **Academic Success** (https://studentlife.utoronto.ca/task_levels/academic-success/) section. There are many resources on topics such as **Better note-taking** (https://studentlife.utoronto.ca/task/better-note-taking/) , **Studying, concentration and memory** (https://studentlife.utoronto.ca/task/studying-concentration-and-memory/) , **Goal setting and motivation** (https://studentlife.utoronto.ca/task/goal-setting-and-motivation/) , **Reboot after an academic setback** (https://studentlife.utoronto.ca/task/reboot-after-an-academic-setback/) , etc. They also offer a free online

course on **5 Keys to Succeed at UofT (https://studentlife.utoronto.ca/program/5-keys-to-succeed-at-u-of-t-online-course/)** .

Recognized Study Groups (https://sidneysmithcommons.artsci.utoronto.ca/recognized-study-groups/)

Recognized Study Groups (RSG) are student-led study groups of up to eight classmates enrolled in the same Faculty of Arts & Science course. RSGs can meet online or in person on the St. George Campus.

Course Summary:

Date	Details	Due
	Assignment 1 Search (https://q.utoronto.ca/courses/354350/assignments/1322146)	
	Assignment 2 Games (https://q.utoronto.ca/courses/354350/assignments/1322147)	
	Assignment 3 CSP (https://q.utoronto.ca/courses/354350/assignments/1322137)	
	Assignment 4 Bayes (https://q.utoronto.ca/courses/354350/assignments/1322149)	