GPR-350
Game Physics

Fall 2019

Sections: **GPR-350-01**: Mon & Thurs, 9:30am – 10:45am
GPR-350-02: Mon & Thurs, 11:00am – 12:15pm
GPR-350-03: Tues & Fri, 9:30am – 10:45am
Classroom: Joyce Hall 101, Game Lab

Course Instructor:

Daniel S. Buckstein
A graduate of UOIT in Ontario, Canada, Dan holds a Master's degree in Computer Science and a Bachelor's degree in Information Technology, majoring in Game Development & Entrepreneurship. He has a keen interest various areas of mathematics, including calculus and linear algebra, both of which carry over to computer animation and graphics. Dan also practices low-level programming and building clean, modular code and frameworks. This has been demonstrated in a long-standing passion for creating high- and low-level graphics and animation tools in both industry and academic contexts. His programming and development interests include game engines, tool design, virtual reality and augmented reality. Dan's passion for programming intertwines with his passion for teaching and he is always looking for innovative ways to deliver course content.

Contact Info: **dbuckstein@champlain.edu**
Office Hours: **West Hall 202; Mon 2–3pm; Tues/Thurs 2–4pm; by appointment**

Course Description:

Students learn the theory and practical application of techniques used in games to simulate real-world physical interactions. Implementation of 2D and 3D algorithms to effect movement, particle systems, collision detection, gravitational forces, kinematics, and spring systems are covered. Implementations trade-offs of efficiency for accuracy are adapted for real time games.

Prerequisite courses: GPR-200 Intro to Modern Graphics Prog; GPR-250 Game Architecture
Credit hours: 3

Course Design:

This course focuses on both theoretical and practical aspects and applications of modern game physics programming. The primary takeaways for the course are **portfolio building** and **engineering**. Theoretical content will be delivered to students through lectures, readings, discussions and tutorials, taking place during the scheduled class time. The hands-on component requires students to program algorithms and systems using C/C++. The scheduled class time will consist of a mixture of theoretical discussions, hands-on programming activities and code reviews. Students are expected to attend and participate in all scheduled classes. Furthermore, it is highly recommended that you practice C/C++ daily to keep up with practices discussed in class. This will enforce your algorithm development skills and understanding of the course materials. Evaluations consist of a

[0] Any part of this syllabus or related documents is subject to change, based on the instructor's discretion.

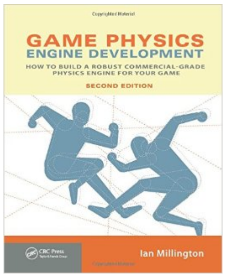
variety of practical exercises both in-class and independently, major term projects, readings and quizzes to reinforce key concepts.

Course Objectives:

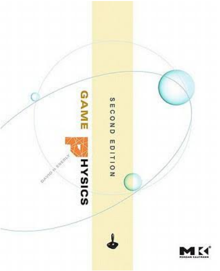
Upon completion of this course, students will be able to:

- ...implement physics simulations ranging from basic to advanced.
- ...apply physics concepts in a game programming context.
- ...utilize vector and matrix math in the context of physics.
- ...describe, create and debug simulated physics worlds.
- ...describe properties of 3D rigid bodies.

Course Books:


GAME PHYSICS
ENGINE DEVELOPMENT
HOW TO BUILD A ROBUST COMMERCIAL-GRADE PHYSICS ENGINE FOR YOUR GAME
SECOND EDITION
Ian Millington

Game Physics Engine Development
2nd edition (2010)
Ian Millington
ISBN: 978-0123819765



Recommended:

Game Physics
2nd edition (2010)
David H. Eberly
ISBN: 978-0123749031

Evaluation Method:

	Deliverable	Occurrences	Total Grade Weight
1.	**Readings & Quizzes**	3% × 10	30%
2.	**Lab Assignments & Demos**	4% × 10	40%
3.	**Midterm Project & Presentation**	10% × 1	10%
4.	**Final Project & Technical Interview**	20% × 1	20%

Evaluation Method Descriptions:

1. Readings & Quizzes (30%)
Quizzes will be assigned weekly throughout the term at the beginning of the first class each week. The quizzes will consist of multiple choice, true/false, short-answer and essay questions. Questions are pertinent to upcoming lessons and content and may include a brief review of past content. The intent of quizzes is to ensure that you are exposed to core terms and concepts, and attain the foundational knowledge needed to proceed with the following lab assignment (see below). **You must keep up with the class activities and readings to-date**. Quizzes will be administered online in Canvas and are **strictly independent evaluations**.

2. Lab Assignments & Demos (40%)
Lab assignments are designed to provide quick yet effective opportunities to gain hands-on experience with the weekly subject matter. Some of the homework assignments may be selected by students from a pool of topics designed by the instructor. Lab activities may build off of prior activities, therefore it is important to do them all and seek help where needed. Labs pertain to recent materials and require programming in C, C++, C# or other languages pertinent to selected engines. The result of each assignment will be a working implementation of a fundamental physics algorithm or principle discussed in class. Activities may also help you familiarize yourself with your selected framework or engine's capabilities. Assignment descriptions and requirements will be provided in separate briefing documents, posted on Canvas, with limited instructions on what needs to be done and how to get there. Grading for each lab is through **demonstration** during work periods, so you must be prepared to show and explain your contribution. This grading method provides you with the opportunity to demonstrate your knowledge firsthand, and to provides you with instant feedback. Specific grading requirements will be provided with each briefing document. Students may work **in pairs or independently** to complete each assignment. Labs may be due as early as one week after being assigned.

3. Midterm Project & Presentation (10%)
Students will select a project from a pool of instructor-designed topics. The project will require students to implement one or more fundamental physics algorithms; the task will require more time and critical thought to complete than the labs. Students may work **in pairs or independently** to complete the project. The projects will be graded in the same fashion as the lab assignments. Project descriptions and requirements will be provided in separate briefing documents on Canvas. Specific instructions **will not** be provided, but a certain level of expectation will be indicated. If done well, these pieces will make excellent portfolio contributions!

4. Final Project & Technical Interview (20%)
The final project is selected from a pool of instructor-designed topics. Students may also propose their own project to the instructor before beginning work. The outcome of the project is a **team-based or independent portfolio contribution**. This project is designed to be flexible and creative, but students should also strive to use this opportunity to help complete Capstone or Production projects. The final project will be evaluated through **demonstration** in the form of a **technical interview**; students are expected to show the product, discuss the code in detail and be professional; this is meant to emulate a job interview! The final project will be assigned throughout the term in stages including, but not limited to: initial proposal and justification, progress checks, and the final interview. The project briefing will be released on Canvas after the first few weeks of class.

Evaluation Policies:

Important: Pass Clause

An average of ***50% in each evaluation category*** is required to receive a passing grade.

Version Control

For all labs and projects, including the final project, some form of **version control** or **source control** must be used consistently and effectively. Popular version control software includes *Git* and *Mercurial* command line interfaces; each has a variety of supplementary GUI-based software available (e.g. *Tortoise Hg* for Mercurial). A portion of the grade for each of these assignments will go towards the use of version control. Submit a link to your public repository, or a repository shared with the instructor's account, either once per assignment or whenever a new repository is started; branching can be used to organize multiple assignments worth of work in a single repository. The use of version control does not waive other submission requirements on each of the assignments; please mind these instructions in assignment briefing documents.

Note: *Ignore filters* exist for a reason; do not track ignored files. If you are unsure of which files should and should not be tracked, *please ask.*

Note: *Branches* exist for a reason; do not do everything on one branch. If you are unsure of how to use branches, *please ask.*

Frameworks & Resources

In-class work will make use of standard game engines, primarily Unity, to demonstrate the weekly material. Assignments may be completed using the student's choice of engine, including but not limited to Unity, Unreal, DeanLib, animal3D, or other primarily C/C++ based engine.

Grading

Grading for all lab assignments will happen through **demonstration** during work periods, which is a good opportunity to explain your thought process directly and collect instant feedback. Demos should take no more than **five minutes** and include a brief show-and-tell of the assignment. Be prepared to show all pertinent components of the assignment (anything specified as to-do in the briefing) and explain how they function. All assignments and projects must also have a Canvas submission, which may or may not be used for further evaluation at the instructor's discretion. Specific grading requirements will be provided in each briefing.

Documentation

All contributions to an evaluation should be documented by the author(s). For example, any algorithm implemented can have a block comment above it that describes the step-by-step breakdown of how it works (in English or pseudocode). Documentation also includes the use of descriptive and meaningful comments in version control commits. The lack of adequate documentation in code may result in grade reductions and a longer wait time to get your grade back!

Deadlines

Deadlines for all evaluations may change with the discretion of the instructor. To avoid missing deadlines, begin work as soon as it is assigned and submit your work at least 24 hours prior to the deadline to avoid rushing to submit and to leave time to fix errors. Canvas submissions will close at the same time as the respective deadlines. Late submissions will receive a grade of **zero** and email submissions will not be accepted. Extensions and make-ups will not be granted unless deemed necessary by documented extenuating circumstances (e.g. a medical emergency).

Asking for Help & Attending Office Hours

The office hours listed above are times only the *bare minimum* times to drop in and ask for help; they do not in any way imply that your instructor is not available outside these times. If you need help outside of the hours listed above, please send an email requesting an appointment. Office hours and appointments are useful for clarifying details about class materials and assignments, getting reliable information, interpreting the syllabus, and chatting about future courses and subject matter. Also note that office hours are for clarifying problems in class, not getting a repeat lesson because you missed class. Please make the best of your education!

Debugging requests of any sort via email will be ignored; this is not an appropriate way to solve a problem and is also extremely difficult since we are dealing with living code. If you have a persistent issue that you need help with, make an effort to attend office hours. Furthermore, when visiting office hours to debug, bring your own laptop and/or ensure your repository is up-to-date.

Finally, asking for critical help with the fundamentals of an assignment within 48 hours of its deadline is last-minute and is simply unacceptable. Last-minute help will only be considered if conversation about the assignment has been sustained since the assign date. Always begin working on an assignment the same day it is assigned so you may identify potential problems immediately.

Zero Tolerance for Plagiarism

Plagiarism is the act of taking credit for graded work that is not yours. Copying a completed algorithm off of a website or out of a book and calling it your own is also plagiarism. Copying your own prior work for evaluation also counts as plagiarism. You may always ask others for help, but asking to view another student's work is a slippery slope towards plagiarism; do not do this, instead, discuss the problem with your peers and see if you can agree on a solution. It is entirely up to the party with the problem to solve the problem.

To avoid the risk of plagiarism in your work, ensure that all contributors label their own work within a submission (see 'Documentation' above). In addition to this, ensure that all borrowed material (e.g. from the internet or a literary resource) is credited appropriately. This course does not mandate the use of a standard citation format (such as APA or MLA), so it will be deemed acceptable to add a URL and/or article information (author, title, year, etc.) directly inline with code where the reference is made. **Any portion of code cited from another source will not count toward your grade**, since it is not original.

There are many reasons why you should not plagiarize. Reasons include, but are not limited to:

1. You are not learning anything by copying others' work with little to no understanding of what you are copying and why.
2. Plagiarism reduces the value and integrity of your degree for you and for your classmates.
3. Your classmates are your future colleagues and industry connections. Do not leave them with a negative impression of your work ethics; ***they will remember you***.

This course has a straightforward and ***strict policy of zero-tolerance for plagiarism***:

1. All parties involved in a case of duplicate or highly-resemblant submissions will be called in for a meeting to discuss the work in question.
2. If you are caught and confirmed plagiarizing in any form, following the usual process your **final grade in the course** will be immediately recorded as "**F**". You may visit the registrar's office to replace this with "W" if the incident occurs before the withdrawal deadline.

Please see Champlain College's Academic Honesty Policy summary below, and the complete policy posted on Canvas. Do not plagiarize. Thank you for your understanding and cooperation.

Avoiding Headaches for Everyone

The easiest way to avoid headaches for you and your instructor is simple: ***read and follow all instructions***. For example, if a project requires you to submit certain files, ensure that these files are submitted. Similarly, if a project asks you to omit certain files, do not submit these files.

In addition to following all instructions, you must ***provide instructions with all submissions on how to use your product and how to navigate the code***. This is especially true if you choose to use a framework other than one provided to complete an assignment. Please pull your weight to minimize grading time; the more you do to help your instructor, the sooner you will get your grades back!

Finally, you must ***test your work before submitting***. Any code submission that does not build and run as expected ***immediately out of the box*** will receive a score of ***zero***. Code grading is painful; not following instructions, providing information about your submission and testing your product all result in delays and therefore longer grading times.

Attendance

You are expected to attend all of the scheduled meeting times for your section since we will cover fundamental topics each week. Furthermore you are expected to show up to class on time. If you are unable to attend a class or anticipate being late, please inform your instructor before the fact. Attendance will be tracked using Canvas's roll-call feature. Unexcused absences will result in a **grade reduction of 2%** for each instance. Unexcused lateness will result in a **grade reduction of 1%** for each instance.

Missing class and showing up late for class will inherently have a negative impact on your grade (e.g., no make-up quizzes, in-class activities). Furthermore it is unprofessional and will not be tolerated in the working world. Students are responsible for all missed work, regardless of the reason for absence. It is also the absentee's responsibility to obtain all missing notes or materials. Attendance refers to both physical and mental presence, i.e. listening in class and staying focused. If at a later date you ask questions about things covered in class because you've never heard them because you were busy playing games, you're on your own.

General Course Policies:

- Put away your phones and other mobile devices for the duration of class. They are are incredibly distracting to you, your peers and the instructor. If you are expecting a call, please excuse yourself. Distracting use of such devices will result in you being asked to leave class. Your attendance is a privilege, not a right; don't waste it.

- Using computing devices for non-class related activities will result in scathing penalties. For your sake and those around you, don't do it.

- Install all required software and packages provided by the instructor. Failure to do so will greatly hinder your progress in the course.

- **Do not** package up your project's working directory with a bug (e.g. "because it just won't work") and expect your instructor to fix it via email. Email your instructor with **specific questions and evidence that you have attempted to solve the problem**.

- Independent Assignments: Students are expected to work independently. **Viewing**, **offering** and **accepting** materials are acts of **plagiarism**, which is a serious offense and **all involved parties will be penalized according to the course's and the College's academic honesty policy**. Discussion amongst students is encouraged, but when in doubt, direct questions to the professor.

- Group Assignments: Students are expected to work collaboratively. All students are expected to contribute equally and are to respect the views of other students. The professor reserves the right to distribute grades according to contribution. Please pull your own weight in your group projects and do not let others suffer due to a lack of organization or interest.

- Demonstrations are graded on-the-spot. Code grading takes at least two weeks; code grades returned in a few days is unlikely and should not change this expectation. All questions regarding any graded work **must** be brought to the attention of the professor within **one week** of when the grade is returned.

General Expectations:

- I expect you to ***read the syllabus and all related documents***. Asking for clarification is acceptable, but asking questions that are explicitly answered in any of these documents will result in a reference to said documents (e.g. Q: "When are your office hours?" A: "Syllabus.").

- I expect you to seek out and find information that you might require as we do not have enough time to cover everything. Just reading the notes is not enough to get you an A. If you are thoroughly confused about any topic or concept, **go to office hours as early as possible**.

- I expect you to be programming ***every day*** or ***every two days*** at minimum in this course.

- I expect you to have passion and to bring that passion with you to class. This includes, but is not limited to, being interactive and asking questions when appropriate.

- I expect that you have functional knowledge of essential mathematics for games and object-oriented programming concepts (classes, inheritance, virtual functions, etc.) as taught in previous courses; thus you should be reviewing these concepts.

Letter Grade Distribution:

			67.00 – 69.99	D+	
$\geq$ 93.00	A	Work is stellar, exemplary.	63.00 – 66.99	D	Work is inadequate.
90.00 – 92.99	A–		60.00 – 62.99	D–	
87.00 – 89.99	B+				
83.00 – 86.99	B	Work exceeds expectations.	$<$ 60.00	F	Work is incomplete or impertinent.
80.00 – 82.99	B–				
77.00 – 79.99	C+				
73.00 – 76.99	C	Work meets expectations.			
70.00 – 72.99	C–				

Tentative Course Outline:

Please see "***GPR350 FA2019 Topics Outline***" for the tentative course schedule, posted as a PDF on Canvas. This document also contains the schedule of assignments, deadlines, quizzes and readings. Note that quizzes are based on the respective readings, therefore they must be done **before** the specified week. Weekly coverage might change depending on the progress of the class. The schedule is provided to inform students of the topics to be covered in the upcoming lecture. Supplementary sources will be recommended before some lectures, students must familiarize themselves with the upcoming topic by doing the reading before coming to class. The meat of each lecture and tutorial will explore core, supporting and advanced algorithms, and how the topic at hand works in the context of game programming. Students should have an intermediate to comprehensive understanding of the topics discussed and be able to create their own applications. Students should constantly strive to expand their personal knowledge.

Data for Research Disclosure:

Any and all results of in-class and out-of-class assignments and examinations are data sources for research and may be used in published research. All such use will always be anonymous.

Syllabus Acknowledgment:

After reading the entirety of the syllabus and all supporting materials, please complete the "Syllabus Acknowledgment" quiz in Canvas. By acknowledging the syllabus, you are indicating that you have had the opportunity to address and resolve any questions or concerns regarding the syllabus content, and that you have read and understand all of the course requirements and policies described therein. **You must complete this acknowledgment to receive a grade in the course!**

Academic Honesty Policy Summary:

Introduction

In addition to skills and knowledge, Champlain College aims to teach students appropriate Ethical and Professional Standards of Conduct. The Academic Honesty Policy exists to inform students and Faculty of their obligations in upholding the highest standards of professional and ethical integrity. All student work is subject to the Academic Honesty Policy. Professional and Academic practice provides guidance about how to properly cite, reference, and attribute the intellectual property of others. Any attempt to deceive a faculty member or to help another student to do so will be considered a violation of this standard.

Instructor's Intended Purpose

The student's work must match the instructor's intended purpose for an assignment. While the instructor will establish the intent of an assignment, each student must clarify outstanding questions of that intent for a given assignment.

Unauthorized/Excessive Assistance

The student may not give or get any unauthorized or excessive assistance in the preparation of any work.

Authorship

The student must clearly establish authorship of a work. Referenced work must be clearly documented, cited, and attributed, regardless of media or distribution. Even in the case of work licensed as public domain or Copyleft, (See: http://creativecommons.org/) the student must provide attribution of that work in order to uphold the standards of intent and authorship.

Declaration

Online submission of, or placing one's name on an exam, assignment, or any course document is a statement of academic honor that the student has not received or given inappropriate assistance in completing it and that the student has complied with the Academic Honesty Policy in that work.

Consequences

An instructor may impose a sanction on the student that varies depending upon the instructor's evaluation of the nature and gravity of the offense. Possible sanctions include but are not limited to, the following: (1) Require the student to redo the assignment; (2) Require the student to complete another assignment; (3) Assign a grade of zero to the assignment; (4) Assign a final grade of "F" for the course. A student may appeal these decisions according to the Academic Grievance Procedure. (See the relevant section in the Student Handbook.) Multiple violations of this policy will result in a referral to the Conduct Review Board for possible additional sanctions.
The full text of the Academic Honesty Policy is in the *Student Handbook*.

Diversity, Equity & Inclusion:

As a learning community, we should be working together to create an effective and respectful space; an environment that is safe and accepting, where everyone feels like they can bring their whole self to the table; a classroom that supports the ever changing and diverse learning needs and complexities of Champlain College students; a space that maximizes learning for everyone. Differences of opinion are encouraged and should be nurtured. Discussions themselves are collaborative learning ventures, and it is often through dialogue that we come to understand our own thoughts and positions.

We all co-create this positive, inclusive learning environment. The following are guiding principles for how this can be achieved:

- Appreciate that everyone is in a different "space" of learning: how they learn, different skill level, and the unique personal context they bring to the classroom. Each of us in this class will know more in some areas and less in others.
- Try not to make assumptions; ask questions to learn more about other perspectives, especially those that are different from your own.
- Lead with "generosity" when you listen to others' perspectives and prioritize kindness over proving the other wrong. Listen to learn first, rather than to immediately judge or "win."
- Respect others' perspectives and recognize your own biases and the limits of your own knowledge.
- Show respect for others as individuals by learning and using their preferred names and pronouns.
- Share your knowledge and skills with tact.
- Applaud each other's efforts and offer constructive feedback when possible.
- Respect the speaker, even when you do not agree with or support the point the speaker is making.
- Do not monopolize the conversation, rather, give others a chance to contribute.
- Realize that no opinion or position is out of bounds (bar hate speech), but all opinions should be articulated with thoughtfulness and humility.
- Take into consideration the impact of your words on others.
- Racial, ethnic, gender or other identity-based slurs are nothing other than hurtful and will not be tolerated.

Students with Disabilities:

If you believe that you have a disability requiring accommodations in this class, please contact the Coordinator of Services for Students with Disabilities as soon as possible. You will be able to schedule a meeting with either Skip Harris or Denise Myers and have your documentation reviewed. During that meeting Skip or Denise will provide you with letters for your faculty, which will detail your needed accommodations. It is the student's responsibility to seek and secure accommodations prior to the start of an exam or project.

Contact: Courtney Cioffredi, Office of Accessibility
Office: Skiff Hall 112, Phone: 802-865-5764, Email: accessibility@champlain.edu

Class during an Extended Campus Closure:

Champlain College is taking precautionary measures to ensure that this class can continue in a "virtual environment" even during an extended emergency such as severe weather, contagious disease, physical infrastructure failure, campus closure, or similar incident. This course will continue either online through a college-provided learning management system (Canvas), or through some other process, unless cancelled. In the event of such an emergency, students are expected to continue instructor-designated class activities, as directed by the instructor. Due to the nature of the "virtual environment" learning activities may differ slightly from the on-campus course. In order for this emergency preparedness plan to be effective, you are asked to do the following:

Immediately

- Ensure that you will have a computer and Internet access at the location (home or other) in which you will reside during an extended campus closure.
- Prepare yourself with the basic skills of logging into Canvas via the my.champlain.edu dashboard, finding your course(s), and entering them.
- Participate in a "warm-up" online activity in the "virtual environment" when directed to do so by your instructor.

During an Emergency

- Test your Internet access immediately upon arriving at your chosen residence during the campus closure.
- Log into Canvas and enter your courses.
- Check for emergency information on Champlain College main website (www.champlain.edu) which will indicate the semester week and day on which college classes will resume online.
- Enter your class and go to the appropriate week of class where you will receive directions from your instructor.