TABLE OF CONTENTS

CPSC 106 DIGITAL STORYTELLING SYLLABUS 2

CPSC 110 INTRODUCTION TO COMPUTER SCIENCE SYLLABUS 9

CPSC 220: COMPUTER SCIENCE I SYLLABUS 12

CPSC 230 DATA STRUCTURES SYLLABUS 16

CPSC 310 COMPUTER INFORMATION SYSTEMS SYLLABUS 18

CPSC 330 OBJECT ORIENTED DESIGN SYLLABUS 23

CPSC 430 SOFTWARE ENGINEERING SYLLABUS 27

CPSC 440 GAME PROGRAMMING SYLLABUS 33

CPSC 444 COMPUTER GRAPHICS SYLLABUS 36

CPSC 106 Digital Storytelling Syllabus

Professor: Jennifer A. Polack
Location: The Internet
Email: jenniferpolack@gmail.com
Office: The Internet.
Office Hours: As needed; there will be weekly online drop in office hours

Course Description The Wikipedia articles on Digital Storytelling defines it rather succinctly as "using digital tools so that ordinary people can tell their own real-life stories." It then goes on to elaborate as follows:*Digital Storytelling is an emerging term, one that arises from a grassroots movement that uses new digital tools to help ordinary people tell their own 'true stories' in a compelling and emotionally engaging form. These stories usually take the form of a relatively short story (less than 8 minutes) and can involve interactivity.The term can also be a broader journalistic reference to the variety of emergent new forms of digital narratives (web-based stories, interactive stories, hypertexts, fan art/fiction, and narrative computer games).As an emerging area of creative work, the definition of digital storytelling is still the subject of much debate.*There are a number of ideas and assumptions here that we will be interrogating over the course of this semester, namely the idea of "ordinary people," "true stories," and the debate around the meaning of this term. The above article is rather vague about the details surrounding this emerging genre of narrative, and it is our responsibility to examine the term digital storytelling within the cultural context of our moment. This means each of you will be experimenting with your own digital platform for storytelling, as well as placing yourself within a larger narrative of networked conversation on the internet at large.This course will require you to both design and build an online identity and narrate your process throughout the five week semester. Given this, you will be expected to openly frame this process and interact with one another throughout the course as well as engage and interact with the world beyond.In many ways this course will be part storytelling workshop, part technology training and, most importantly, critical interrogation of the digital landscape all around us that is increasingly defining the the way we communicate with one another.

Course Objectives

- To develop skills in using technology as a tool for networking, sharing, narrating, and creative self-expression
- To frame a digital identity wherein you become both a practitioner in and interrogator of various new modes of networking
- To critically examine the digital landscape of communication technologies as emergent narrative forms and genres

Course Materials

- Internet: There is no textbook for this class, however individual readings/videos will be assigned and will all be available online. Success in this class is very much dependent on a reliable, fast Internet connection.
- Computer: Do we need to even list this?

- o Web Accounts/Software: You will need to set up accounts on various social media sites we will be using for class. For the most part, no specific software is required; you will need to use what you have or choose from web-based/trial versions of software to create media. See the Packing List
- o Web Hosting Account: You will be expected to manage a web hosting account with http://umw.domains
- o Class Web Site: The location of the course's online activity will behttp://ds106.jenpolack.com You should always use this URL to enter the course; it is where you will find information about assignments and activities. Over the course of the semester, we will also make use of two other important ds106 sites:

◦ ds106 Assignment Repository: This collection of digital storytelling assignments has been developed over the course of the last few years. We will frequently be drawing upon this collection for course assignments. You will also be creating assignments as part of your coursework.

◦ The Daily Create: These daily creative assignments ask you to spend no more than 10-15 minutes experimenting with either photography, video, audio, or text based on a pre-defined assignment.

Department of Computer Science Grading Scale

If applicable, here is the grading scale:

A 92-100% | A- 89- 91% | B+ 87-88% | B 82-86%| B- 79-81% | C+ 77-78% | C 72-76% | C- 69-71% | D+ 67-69% | D 60-66% | F 0-59%

Grading Breakdown

Participation (15% of Final Grade)

This class will in many ways be anchored around your ongoing, regular participation through the various technologies you will be experimenting with. If you are not present, you will compromise the success of the class (as well as YOUR success in it). We expect active and engaged participation.

For the purpose of this entirely online version of ds106, presence and participation are determined by the degree to which you are actively and thoughtfully engaged with your classmates and the course materials via the various online spaces used for the class. Participation will be evaluated based upon the following kinds of activities:

- o Narrating your course experience. Throughout the summer, you are required to use your blog to regularly provide updates about your course activities. These posts should be thorough and reflective.
- o Commenting upon your classmates' work. You are expected to respond thoughtfully and critically to the work that others in the class are creating.

This will be accomplished in several ways, primarily through regular, thoughtful blog comments and feedback on Twitter. You should focus your commenting energies most explicitly upon the work of your proximal zone mates. But we encourage you to actively review and comment upon the work of anyone in ds106.

Keep in mind comments are distributed, and while we have certain mechanisms for tracking comments, they are imperfect. We will be expecting you to keep track of your feedback on your classmates work and share it with us in the assessment conferences we have during the semester.

- Engagement with social media. The online nature of this course requires us all to work especially hard to build a learning community. In large part, we expect this community to emerge out of various spaces and tools that you will be asked to use. We will be looking for your regular presence in spaces like Twitter, Flickr, and YouTube. Complaining that you "don't understand" the tool is not a suitable excuse. You will only begin to understand by using and engaging.

The Daily Create (15% of Grade)

Regular, creative exercises are at the heart of ds106, and to this end over the course of the semester we will be expecting every student to complete four (4) Daily Create each week. In order to get full credit for this assignment you will need complete it the day the assignment was posted as well as tag it according to the directions given with the prompt.

Digital Storytelling Assignments (40% of Final Grade)

Throughout the semester, we will assign a number of digital storytelling projects using a variety of tools, techniques, and technologies. You are expected to complete all of these assignments in a timely fashion and share them on your blog. Your grade on these will reflect both your success at completing these assignments as well as a detailed commentary on your blog describing your process and any difficulties you encountered. In other words, you will be expected to not only complete an assignment, but also share with everyone how you did it. What's more, if you have difficulty with an assignment we will always expect you to attempt it, but you can use your blog to share insight into what you found challenging and how you negotiated the requirements.

Generally speaking, as long as we see a commitment to completing an assignment creatively and sharing your process thoroughly, you can expect to do well on it. If you don't complete an assignment, you will receive a zero. If you complete an assignment, but you have failed to document your process (and have not explained to us why you did not meet the requirements), you can expect to receive partial credit.

Also, keep in mind each assignment in the ds106 assignment repository has two tags. You are required to use both tags from each assignment correctly to receive credit. It is your responsibility to double check the spelling of the tags and ensure they are correct for each and every assignment you create.

You are expected to review the course site regularly and to complete all assignments on-time.

Creating (and Completing) Your Own Assignments (10% of Final Grade)

Over the course of the semester we would like each of you to create at least two assignments. Each assignment you create must be for a different section of the course (i.e., visual, design, audio, video, and mashup/remix)–feel free to create more than two, but that is your minimum. The assignments should be relatively short and creative. In addition, you must do the assignment you create and document your own process for completing it. You can submit the ones you create here.

Remember, each assignment has to be tagged correctly to receive credit—and those tags will be created immediately after you submit the assignment. Don't forget to tag your example of the assignment you complete.

Creating Tutorials (10% of Final Grade)

In addition to creating at least two assignments, you will be required to create at least two tutorials for either assignments you create or pre-existing assignments in the repository. These tutorials can be blog posts with specific instructions or screenshots, screen casts walking an audience through the process, or some other approach to helping others complete the task.

Like assignments, tutorials have tags that need to be added to the post on your blog in order for it to be associated with the proper assignment. You need to check the tutorial tag for the assignment you are writing the documentation for. You will need to correctly use the tutorial tags to get full credit.

Also use the tag **ds106tutorials** so that we can aggregate the tutorial into one page.

Story Challenge (10% of Final Grade)

This will be a digital story of your making in a response to a challenge given out the final weeks of class. You may tell the story using any of the tools or techniques we discuss in the class; the only requirement is that you share it publicly on your blog and it be something you are interested in.

Other Non-Graded but Important Course Elements

Conferences

Twice during the semester (week 2 and week 4), we will set up a meeting to discuss your progress in the class. The meetings generally should not take longer than 10 minutes and will be conducted via a tool like Skype or Google Hangout. These conferences are a valuable opportunity for you to receive one-on-one feedback about your work. **These appointments are required.**

Attendance

As this class is online, attendance is not required for synchronous sessions, but you are expected to be regularly active in your own blog, commenting and offering critiques on others, and sharing ideas and resources via Twitter.

Weekly readings or reviews of video/web sites are expected to be written up as blog posts.

Twitter

You must use Twitter for this class. If you already have an account, you may use it. Otherwise, creating an account is easy! Simply tweet class-relevant content with the hashtag **#ds106**. These tweets will be harvested and displayed on the course website. In addition, Twitter can and should be integrated with your class blog. For example, when you complete a new entry, post a link on your twitter account.

Contacting Me

I can be contacted many ways, but e-mail is the easiest Jennifer Polack: jenniferpolack@gmail.com. Our correspondence will be much more productive if you follow a few simple guidelines:

- First, consider whether you really need to e-mail me. If you're experiencing a technical problem, make every effort to solve it first on your own (though a Google search, a call for help blog post, etc.). If you do need to ask for technical help, your message should indicate that you've already tried available means to solve the problem, including specific steps you've already taken.
- Don't forget to identify yourself. If you have a question about an assignment, please make sure we know who you are, what section you're in, and the exact assignment about which you have a question.

- Please send a followup. If an explanation helped, or if the technical suggestion worked, please send a note. This way, I know whether or not to make the same suggestion to someone else when they come to me with a similar problem.

Decorum

Students are expected to treat the instructor and fellow students with the appropriate degree of respect in all interactions. Communication, either in person or through electronic media, that is deemed abusive, threatening, or harassing in nature will not be tolerated.

The Honor Code

Students are expected to conduct themselves in a manner consistent with the letter and spirit of the Honor Code. A violation of the Honor Code is a very serious matter.

Online Accessibility

The University of Mary Washington is committed to ensuring that all students have the same opportunities to successfully participate and learn in online courses as they do in traditional, face-to-face courses.

To this end, we will make every effort to make sure the media we create in this online courses should be developed and presented in ways that are universally accessible

- Images should be optimized and include descriptive "alt" tags
- Written transcripts of audio files and video files should be made available.
- Whenever possible, alternative formats of materials should be made available to students who require them (e.g. optional print packet of extensive online reading materials, CD of audio clips)
- Web sites and Web-based tools should adhere to accessibility "best practices"
- Mechanisms should be available for including "alt" texts when images are uploaded or used.
- Text should be legible and re-sizable
- Use of color should add interest and indicate interface choices, but should not disadvantage those with color blindness.
- When approached by a student with an unanticipated access concern, online faculty should make every attempt to address the concern by adjusting requirements, providing extensions, or making additional accommodations.

Disability Service Statement

The Office of Disability Services has been designated by the University as the primary office to guide, counsel, and assist students with disabilities. If you receive services through the Office of Disability Services and require accommodations for this class, make an appointment with me as soon as possible to discuss your needs. I will hold any information you share with me in strictest confidence unless you give me permission to do otherwise.

Course Calendar

The following schedule lays out the basic structure of the class and the units and topics we'll cover over the semester.

Week 1 (Boot Camp)

The first week of our "Boot Camp" for getting in shape for ds106; setup of your blogs and creation other social media accounts. Complete Introductions via blog, twitter. Review of a range of media with exploration question of what kind of stories do they tell. Explore concepts of personal cyberinfrastructure

Week 2 (Boot Camp cont'd)

This week you go deeper into customizing your blog with themes/templates, plugins/widgets, and other key settings. Also, this will be the week you get introduced to the Daily Create. As part of Boot Camp activity this week, you will need to write blogs posts that embed media. You will also need to begin commenting on your classmates' blogs and soliciting comments on your own. This week's discussion and reflection will be around the ideas of sharing media and creative commons.

Week 3 Introduction to Storytelling

Storytelling… it's part of the title this course, and you likely have some idea what it means from your childhood or school years. This week we will explore it in the framing of what you will be doing for the next 13 weeks in ds106. You will hone in on your own understanding of what the Digital part adds, and try your hand at a few creative exercises.

Week 4 Audio Intro (Listening)

This week in ds106 we're going to be diving into our first storytelling genre: audio. Working with audio can be a bit daunting and unfamiliar, so we'll be easing you into it this week. We'll ask you to do some listening exercises as well as begin to create your own

Week 5: Visual

In doing your ds106 Daily Creates you've already been using photography and drawing skills, plus you've had some practice on doing visual stories for our introduction to storytelling. In this week we go a bit deeper and give you the opportunity to practice telling stories in primarily visual form.

Week 6: Design

We've wrapped up our week on visual storytelling and photography, and this week we're diving into design. You'll spend some time this week thinking about the way the world around you is designed, as well as creating some of your own design projects. This is when we start forming groups for doing the mid term radio show project

Weeks 7 & 8: Advanced Audio (radio show)

The next two weeks are all about audio and radio. The majority of your work during this time should be working on your radio show.

Week 9: Radio Shows Go Live and Telling Stories Within the Web

All the work your groups have done the last two weeks pays off. This week you get to broadcast your shows live on ds106radio. In addition, we move to a different kind of storytelling, one that uses the space of existing web sites as a place for you to assert your own stories into them.

Table of Contents

Week 10: Reading Movies

This week we enter what most students find the most challenging yet rewarding portion of ds106: video. Working with video presents challenges with file formats and using more complex software. But the end rewards are often the most rewarding. Before we jump into editing, we want to spend some time first looking critically at the video form and do some pre-planning for your first video editing.

Weeks 11 & 12: Movie Time

It's time to make movies! Video is perhaps the most rich of storytelling forms, and we want you to focus explicitly now on video storytelling. The only assignments on your plate are to work on video assignments, and all of which are due in two weeks time.

Week 13 & 14: Remix

For the next two weeks, you'll be exploring the ideas of remixes and mashups, the artistic recasting of existing media into new forms by creative combination and editing. This will build off of your previous work in all media forms. And we will even remix assignments.

Weeks 15 & 16: Final Project and Wrap Up

CPSC 110 Introduction to Computer Science Syllabus

M, W, F, Trinkle Hall, B7
Professor Jennifer A. Polack, Ph.D.
Phone: 654-1318
Email:**jenniferpolack@gmail.com**
Very Important to Email me at Gmail, school mail is not checked regularly
Check you email daily, I email notices and assignment regularly

Description: This course provides a foundation in computer science for students who have little or no prior computing experience. This course provides sufficient support to permit the student to continue in the major program. This course introduces problem solving methods and algorithm development as appropriate to Computer Science. Upon completion each student should gain the knowledge of how to design, code, debug, and document computer programs using techniques of good programming style.

- **Prerequisite**: n/a
  Text: An Introduction to Programming with Java Applets Third Edition by Elizabeth Sugar Boese
- **Office Hours:** MTWR 9-10am, F 11-12pm
- **Grading:** Labs, In-Class Work & Homework 50% Exams 50 %

	Tentative Topics
1	Chapter 1
2	**Chapter 2**
3	Chapter 3
4	Chapter 4
5	Exam
6	Chapter 5
7	Chapter 5
8	Chapter 6
9	Chapter 7
10	Chapter 8
11	Chapter 9
12	Exam
13-5	TBD

- We will be doing in-class assignments that may not be in this packet.
- Anything not in this packet will be handed out at the beginning of class.
- **LABS WILL NOT BE ACCEPTED LATE**

In Class Lab Grading Scheme

- 100% Credit: The lab was turned in by the end of class and works.

- 80% Credit: The lab was turned in by the end of class and almost does what it is suppose to.
- 75% Credit: The lab was turned in by the end of the class and sort of works
- 50% Credit: The lab was turned in by the end of the class and compiles but does not do what it is suppose to do.
- 25% Credit: The lab was turned in by the end of class but it does not compile.
- 0%: The lab was not turned in on the due date

Out of Class Labs Grading Scheme

	Unsatisfactory	Satisfactory	Good	Excellent
Delivery	• Completed less than 70% of the requirements. • Not delivered on time or not in correct format (disk, email, etc.)	• Completed between 70-80% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)	• Completed between 80-90% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)	• Completed between 90-100% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)
Coding Standards	• No name, date, or assignment title included • Poor use of white space (indentation, blank lines). • Disorganized and messy • Poor use of variables (many global variables, ambiguous naming).	• Includes name, date, and assignment title. • White space makes program fairly easy to read. • Organized work. • Good use of variables (few global variables, unambiguous naming).	• Includes name, date, and assignment title. • Good use of white space. • Organized work. • Good use of variables (no global variables, unambiguous naming)	• Includes name, date, and assignment title. • Excellent use of white space. • Creatively organized work. • Excellent use of variables (no global variables, unambiguous naming).
Documentation	• No documentation included.	• Basic documentation has been completed including descriptions of all variables. • Purpose is	• Clearly documented including descriptions of all variables. • Specific purpose is noted for each	• Clearly and effectively documented including descriptions of all variables. • Specific purpose is

		noted for each function.	function and control structure.	noted for each function, control structure, input requirements, and output results.
Runtime	• Does not execute due to errors. • User prompts are misleading or non-existent. • No testing has been completed.	• Executes without errors. • User prompts contain little information, poor design. • Some testing has been completed.	• Executes without errors. • User prompts are understandable, minimum use of symbols or spacing in output. • Thorough testing has been completed	• Executes without errors excellent user prompts, good use of symbols, spacing in output. • Thorough and organized testing has been completed and output from test cases is included.

CPSC 220: Computer Science I Syllabus

Monday & Wednesday
MW: 10:00 am - 11:50 am Trinkle Hall B6
Professor Jennifer A. Polack, Ph.D.
Phone: 654-1318
Email:jenniferpolack@gmail.com
Very Important to Email me at Gmail, school mail is not checked regularly. Check your email daily, I email notices and assignment regularly

Description: Continued coverage of disciplined problem-solving and algorithmic development including emphasis on procedural and data abstraction. Topics include elementary data structures such as arrays, files, and classes. The notions of data modeling and the linking of data type definitions with their associated operations is introduced. Study of program design, coding, debugging, testing, and documentation in a higher level language that supports the object-oriented paradigm. Intended for students who have had previous programming experience.

Prerequisite: Computer Science 110 or Equivalent
Required Text: *Starting Out with Games & Graphics in C++* 2nd Edition by Tony Gaddis, ©2014, 2010 Pearson Education - Addison Wesley, ISBN: 978-0-13-312807-9
Gaming Website: http://darkgdk.com
Office Hours: TBA

Goals and Objectives:

1. Students will demonstrate an ability to interpret quantitative/symbolic information.
2. Students will have the ability to convert relevant information into various mathematical/analytical forms (e.g., equations, graphs, diagrams, tables, words)
3. Students will be able to apply analytical techniques or rules to solve problems in a variety of contexts.
4. Students will gain an appreciation for how analytical techniques or rules are used to address real-world problems across multiple disciplines.

Topics:

- Decision Statements
- Repetition Statements
- Recursion
- Methods
- Parameter passing
- Overloaded method names
- Concept of a Class
- Using classes
- Creating a data type
- The purpose of class members and methods
- Intro to information hiding, encapsulation, inheritance, & polymorphism
- Arrays as a data structure and abstract data type
- Searching algorithms - linear search, vs. binary search
- Sorting algorithms
- The List as an abstract data type
- Exceptions & Exception handling
- Using an IDE
- Input & Output (including file I/O)

Tentative Calendar

Plan	Topic
Week 1	Course Plan (syllabus) General Course Information Appendix A Downloading and Installing the Required Software Chapter 2 C++ Fundamentals
Week 2	Chapter 3 Decision Structures and Boolean Logic
Week 3	Chapter 4 Repetition Structures
Week 4	Exam 1
Week 5	Chapter 5 Functions
Week 6	Chapter 5 Functions Chapter 6 Arrays
Week 7	Chapter 6 Arrays
Week 8-9	Part 2 Intro to Game Programming with the App Game Kit Chapter 7 Using the App Game Kit with C++
Week 10	Exam 2 Chapters 5-7
Week 11	Chapter 8 Input, Animation and Sound
Week 12-13	Chapter 9 Text, Collisions and the Vulture Trouble Game
Week 14	Chapter 10 Using Files and Arrays with the AGK
Week 15	TBD
Final	Wednesday, May 1 8:30-11:00 am

Course Participation

Each student's *active participation* in CPSC 220 is of utmost importance. An important part of each student's intellectual and creative growth takes place through effective communication between the instructor and the student and between the student and his or her peers. It is the responsibility of every student to stay in close touch with the professor throughout the semester and to actively participate in class discussions and activities.

Grading:

Anything below a 70% will be reported as unsatisfactory on mid-semester reports.
All Programming Assignments and Labs 50% Exams 50 %. Final Grade are based on the following:

- 92-100 A
- 89-91 A
- 87-88 B+
- 82-86 B
- 79-81 B-
- 77-78 C+
- 72-76 C
- 69-71 C-
- 67-68 D+
- 60-66 D
- <60 F

Mid term, grades will be based on a straight average of all grades given to that point. Students with averages of less than 70% and students who have not turned in all required assignments will receive an unsatisfactory grade.

LABS and PROGRAMMING ASSIGNMENTS WILL NOT BE ACCEPTED LATE

In Class Lab Grading Scheme

- 100% Credit: The lab was turned in by the end of class and works
- 75% Credit: The lab was turned in by the end of the class and sort of works
- 50% Credit: The lab was turned in by the end of the class and compiles but does not do what it is suppose to do.
- 25% Credit: The lab was turned in by the end of class but it does not compile
- 0%: The lab was not turned in on the due date

Out of Class Labs Grading Scheme

A 0% is given if the code does not compile. If the majority of the program works but one section is causing it to crash comment it out. If you comment out 51% or more of the program because it doesn't compile you will receive 0%.

	Mastery (4)	Accomplished (3)	Developing (2)	Beginning (1)
Is the appropriate data structure and implementation specifics used (40 %)	You used everything I required.	You used the correct data storage but not all the methods. You used multiple files.	You did not use the data storage and/or methods. You use multiple files.	You put everything in one file.
Is the code easy to read? (10 %)	Code is tabbed Plenty of comments explaining blocks of code	Code is tabbed, but few comments	Code is generally in a straight line	Code is jumbled
Does the code accomplish all tasks? (60%)	Code completes all tasks, including special cases.	Code completes all tasks but special cases	Code completes some tasks.	Code completes no tasks

Honor Code

I run all of my classes on the Honor Code system; in other words, I trust you to do your own work at all times. However, I expect students to work together in laboratory sections, which means you can help each other correcting syntax and logic errors. You cannot copy any code from one person's screen to another, however they can reference your notes for comparison and they can point out mistakes on your screen. All outside programming assignments must be done on your own. You can discuss logic is generally terms but no code specifics. You can ask for classmates if they recognize your syntax errors and give you advice on how to fix them but they cannot copy any code from you. If you have even the slightest doubt that an activity violates the Honor Code - don't do it.

Disability Statement: The Office of Disability Services has been designated by the University as the primary office to guide, counsel, and assist students with disabilities. If you already receive services through the Office of Disability Services and require accommodations for this class, make an appointment with me as soon as possible to discuss your approved accommodations needs. Please bring your accommodation letter with you to the appointment. I will hold any information you share with me in the strictest confidence unless you give me permission to do otherwise.

If you have not contacted the Office of Disability Services and need accommodations, I will be happy to refer you. The office will require appropriate documentation of disability. Their phone number is 540-654-1266. The office is located in Lee Hall, Room 401.

CPSC 230 Data Structures Syllabus

Monday and Wednesday
MW Section : 10:00 am - 11:50 am Trinkle Hall B7

Professor Jennifer A. Polack, Ph.D.
Phone: 654-1318
Email:jenniferpolack@gmail.com
Very Important to Email me at Gmail, school mail is not checked regularly
Check you email daily, I email notices and assignment regularly

Description: Study of data modeling and incorporation of abstract data types including linked lists, stacks, queues, heaps, trees, and graphs. Study of advanced sorting and searching techniques. Provides experience in the use of algorithm analysis. Continued study of program design, coding, debugging, testing, and documentation in an object-oriented higher level language.

Prerequisite: Computer Science 220
Text: C++ Programming: Program Design Including Data Structures. MALIK.

- Office Hours: 9-10am MTWR and 12-1pm M
- Grading: All Programming Assignments, Homework and Labs 50% Exams 40 % Quizzes 10%
- Tentative Schedule (3 Exams and will be announced a week in advance)

Week	Topic
1	Pointers, Dynamic Arrays (List) and Unix and VIM
2	Pointers, Dynamic Arrays (List) and Unix and VIM
3	Makefiles, Dynamic Arrays (List) and Templates
4	Makefiles, Dynamic Arrays (List) and Templates
5	Linked List and Operator Overloading
6	Linked List and Operator Overloading
7	Double Linked List
8	Midterm
9	Stacks and Composition
10	Queues and Inheritance
11	Trees
12	Tree Lab: Monday Graphs
13	Graphs and Priority Queues
14	Sorting and Searching
15	Doubly Linked List
	Finals MW Class will use the 11am timeslot for exam

	Monday, April 27	Noon - 2:30 p.m.	Classes taught at 11:00 - 11:50 p.m., MWF

- LABS and PROGRAMMING ASSIGNMENTS WILL NOT BE ACCEPTED LATE

In Class Lab Grading Scheme

- 100% Credit: The lab was turned in by due date and works.
- 80% Credit: The lab was turned in by due date and has the majority of work
- 75% Credit: The lab was turned in by due date and sort of works
- 50% Credit: The lab was turned in by due date and compiles but does not do what it is suppose to do.
- 25% Credit: The lab was turned in by due date but it does not compile.
- 0%: The lab was not turned in on the due date

Out of Class Labs Grading Scheme

		Mastery (4)	Accomplished (3)	Developing (2)	Beginning (1)
	Is the appropriate structure and implementation specifics used (40 %)	You used everything I required.	You used the correct methodology but not all the methods.	You did not use the discussed methodology and/or methods	You use only the display method
	Is the code easy to read? (10 %)	Code is tabbed Plenty of comments explaining blocks of code	Code is tabbed, but few comments	Code is generally in a straight line	Code is jumbled
	Does the code accomplish all tasks? (50%)	Code completes all tasks, including special cases.	Code completes all tasks but special cases	Code completes some tasks.	Code completes no tasks

CPSC 310 Computer Information Systems Syllabus

Professor: Jennifer Polack

Office: B21 Trinkle Hall

Email: jenniferpolack@gmail.com

Office Hours: M T W 9:00 – 10:00am, TR 12:00 – 1:00pm

Pre-Requisites: CPSC 220

Course Goals Objectives

- The primary goals and objectives are:
  - Explain the role of technology as a business enabler
  - Identify and explain applications in a business setting
  - Evaluate the organizational fit and suitability of business applications
  - Interpret the interaction between technology, customers, processes, data, infrastructure, participants, and environment an organization.
- The secondary goals and objectives are:
  - Discriminate and synthesize between different sources of information as part of application acquisition or development
  - List ethical and intellectual property challenges that arise from the use of technology
  - Explain the evolving role of MIS in the organization, the MIS professional, and careers in MIS

Textbook

Required

1. Information Systems Today: Managing in the Digital World by Valacich and Schneider 6th

Grading	
Item	Percent of Total Points
Exams	50%
Group Project	25%
Individual Projects	25%
In Class Work	25%
Total	100%

Grading Scale			
94-100	A	73-76	C
90-93	A-	70-72	C-
87-89	B+	67-69	D+
83-86	B	63-66	D
80-82	B-	60-62	D-
77-79	C+	Below 60	F
Anything below a 70 with get a Midterm Unsuccessful Grade Report			

Table of Contents

Tentative Schedule

The schedule below is tentative and is subject to updates and modifications as the course progresses, particularly if guest speakers are arranged.

eek	Topics	Reading	Deliverable
1	se Introduction Managing in a Digital World Vorld Is Flat video	Valacich: Chapter 1	Group Formation
2	rs p Project Discussion and E-Portfolio Discussion		Group Proposal
3	ng Competitive Advantage Through Information Systems	Valacich: Chapter 2	Elevator Pitch
4	ging the Information Systems Infrastructure & Services	Valacich: Chapter 3	Project Charter
5	ling Business-to-Consumer Electronic Commerce	Valacich: Chapter 4	Project 3 Phase 1 – e-Portfolio due
6	ncing Organizational Communication & Collaboration Using Social Media	Valacich: Chapter 5	Solution Proposal
7	ncing Business Processes Using Information Systems	Valacich: Chapter 6	Project 1 – Business Applications
8	ncing Business Processes Using Enterprise Information Systems	Valacich: Chapter 7	Business Model
9	gthening Business-to-Business Relationships via Supply Chain and Customer Relationship Management	Valacich: Chapter 8	Project 3 Phase 2 – Google Analytics Status Update
10	loping and Acquiring Information Systems	Valacich: Chapter 9	Status Update
11	ing Information Systems	Valacich: Chapter 10	Status Update Project 2 – ERP
12	Up		Status Update Project 3 Phase 3 – Google Analytics Results
13	UP PROJECT WRAP UP		Status Update
14	UP PROJECT PRESENTATION		Final Project Presentation and Prototype
15	Friday, December 12	Noon - 2:30 p.m.	Classes taught at 11:00 - 11:50 p.m., MWF

Attendance Policy	• Class discussion is intended to be an integral part of the course. Accordingly, I expect full attendance by every member of the class. • If you are absent from class, speak with your classmates to catch up on what you have missed.
Class Discussions	• Come to class prepared to discuss the required reading from your textbook! • The focus of class will be to discuss the material in the text, NOT TO COVER the material in class. • Students will be called upon to share their thoughts on the required reading in a random fashion. Students may share their understanding of the material or, if they had questions about the material, ask insightful questions about the aspects they didn't understand. Either way, students must demonstrate that they have completed the required reading and have thought about the material. • If a student is not prepared for class and is not in a position to discuss the material intelligently, they may simply respond "Pass" and I will move on to another student and will not call on this student again during this class. However, if you choose to "Pass" during one class it will be extremely likely that you will be called upon during the next class. • Being absent from class or failure to be prepared for class when called upon will impact your final grade. For every three times you are called on and you are absent or not prepared for class, you final grade will be reduced by one fraction of a letter grade. • Being absent from class when called upon will be equivalent to using a "Pass."
Class Etiquette	• Please be respectful of the class environment. • Class starts promptly at the start time. Please make EVERY effort to be on time, as I will communicate important information in the first few minutes of class. • Cell phones must be turned off and put away during class. • Refrain from personal discussions during class. Please leave the room if you need to speak to another student for more than a few words. If a student cannot refrain from engaging in private conversation and this becomes a pattern, the students will be asked to leave the classroom to allow the remainder of the students to work.
Exams	• There will be three examinations during the semester. The exams cannot be made up.

Late Assignment Policy	• All assignments are due at the beginning of class. As you will note in the tentative schedule, we will typically discuss your deliverables on the due date. Accordingly, I cannot accept any late deliverables. A deliverable is considered late if it is turned in after the beginning of class. This time will be strictly enforced. • Equipment failure is not an acceptable reason for late submission of a project. During the semester, computers will fail, printers will be out of service, and your e-mail will mysteriously not work. You should always make a backup of your files (if it is really important, make two backups). You should make sure you print out your work early enough that you can find an alternate location to print.
Submission of Work	• Please submit all assignments on Canvas unless otherwise instructed.
Reading and Class Participation	• The primary source of material for this course is the textbook. In addition, supplemental materials will be provided to you as either hyperlinks to documents on the web, or soft versions posted to the class blog. • During many classes, we will have an interactive discussion of a case or a scenario. Without reading the assigned material, you will not be able to participate and you will find yourself lost.
Appropriate use of Technology in the classroom	• Please turn off cell phones at the start of class. If you have an urgent, personal situation and may be receiving an important phone call during class, please let me know this at the beginning of class, sit near the door, and step out of the classroom if you need to take a call. • While the use of laptop computers in the classroom is permitted for taking notes, using a laptop for any other purpose is prohibited. This distracts the students sitting around you. If I find that you are using a laptop for something other than taking notes, you will be asked to put your laptop away and you will no longer be permitted to use a laptop in the classroom.

Plagiarism, Academic Dishonesty and Citation Guidelines

Plagiarism and academic dishonesty can take many forms. The most obvious is copying from another student's exam, but the following are also forms of this:

- Copying material directly from the Internet (or another source) without a proper citation crediting the author
- Turning in an assignment from a previous semester as if it were your own
- Having someone else complete your lab assignment and submitting it as if it were your own
- Signing someone else's name to an attendance sign-in sheet
- Use of assignments completed in one class as any part of a project assigned in another class

- Sharing/copying homework assignments.
- Use of unauthorized notes during an examination
- In cases of cheating, both parties will be held equally responsible, i.e. both the student who shares the work and the student who copies the work.

Of course, behavior like this will not be tolerated in this class. Penalties for such actions are given at my discretion, and can range from a failing grade for the individual assignment, to a failing grade for the entire course.

If you use text, figures, and data in reports that was created by others you must identify the source and clearly differentiate your work from the material that you are referencing. If you fail to do so you are plagiarizing. There are many different acceptable formats that you can use to cite the work of others (see some of the resources below). The formats are not as important as the intent. You must clearly show the reader what is your work and what is a reference to somebody else's work.

How to successfully quote and reference material:
University of Wisconsin Writers Handbook
http://www.wisc.edu/writing/Handbook/QuotingSources.html

How to cite electronic sources:
Electronic Reference Formats Recommended by the American Psychological Association
http://www.apastyle.org/elecmedia.html

Office of Disability Statement

I am committed to the principle of universal learning. This means that our classroom, our virtual spaces, our practices, and our interactions be as inclusive as possible. Mutual respect, civility, and the ability to listen and observe others carefully are crucial to universal learning.

Any student with particular needs should contact the Office of Disability at 401 Lee Hall 1301 College Avenue Fredericksburg, VA 22401 Phone: 540-654-1266 Fax: 540-654-2155 E-mail: odr@umw.edu, at the start of the semester. The Dean of Students' office will forward any necessary information to me. Then you and I can work out the details of any accommodations needed for this course.

Table of Contents

CPSC 330 Object Oriented Design Syllabus

Spring 2010

Instructor:	Dr. Jennifer A. Polack-Wahl	Phone:	654-1318
Office:	B18 Trinkle Hall	Email:	japwahl@gmail.com
Office Hours:	TBA		

Course Description

This course will take you to the next level as a programmer. You've mastered the basics in 220 and 230, you know all about how to use methods and write a list class and you've worked on some challenging assignments, but perhaps it all felt a little contrived and theoretical. In this class we're going to focus on essential skills required to be an effective professional software developer. We're going to learn about design patterns, using the Java API, and how to present your ideas in UML. You're going to build your skills in communication and working in a group. You'll learn about using debuggers and version control. Get ready to get professional!

Catalog Description

Prerequisite: Computer Science 125 and 230.

Theory and practice of the object-oriented software development paradigm. Focus is on major design principles such as abstraction, encapsulation, inheritance, polymorphism, aggregation, and visibility. Modeling notations for capturing and critiquing designs. Introduction to the concept of design patterns, and coverage of a catalog of common patterns. Students work in team projects to develop collaborative software solutions in an object-oriented language.

Course Objectives:

- To provide opportunities that require the design, coding, implementation, debugging, and documentation of original programs using techniques of good programming style design that emphasize object oriented design.
- To introduce the concept of design patterns and become familiar with the application of common patterns,
- To introduce the Java programming language and become familiar with the Java API.
- To provide writing experience as appropriate to computer science.
- To provide an examination of a topic with social, ethical and professional relevance.

Course Topics:

Introduction to Java Swing

330 Required Patterns

1. Strategy Chapter 1 Heads First
2. Observer Chapter 2 Heads First
3. Decorator Chapter 3 Heads First
4. Factory Chapter 4 Heads First
5. Singleton Chapter 5 Heads First
6. Template Method Chapter 8 Heads First
7. Composite Chapter 9 Heads First

UML

- class diagrams
- sequence diagrams
- collaboration diagrams

Required Texts

DEVELOPING SOFTWARE WITH UML
OESTEREICH
ISBN:9780201756036

HEAD FIRST DESIGN PATTERNS
FREEMAN
ISBN:9780596007126

JAVA SWING
LOY
ISBN:9780596004088

I do NOT require you to own a Java reference book. We're going to be using the Java API online heavily. If you like to have things in a printed format, you may benefit from purchasing:
Java in a Nutshell, 5th edition (ISBN: 0 596007736). You can get a new copy for about $30 or a used copy for around $10. It's enormous and has most of the API printed in it.

How to reach me:

By far the best way is to come to office hours or e-mail me (to japwahl@gmail.com).

How I reach you:

I will be communicating with you outside of class time via e-mail (using the e-mail address that Blackboard has on file for you.) In order to get this information in a timely manner, **you need to check your e-mail every single day!** If you're not already in the habit of doing this, figure out some consistent time during your day (for example, first thing before class, right before lunch, last thing before you go to bed, or whatever) and just log on to your preferred email for a few minutes. Otherwise you'll miss important updates, including help on the assignments!

Rules of the game:

1. Absolutely, positively, NO question is stupid!! Your job is *not* to already know everything before you start the course. Your job is to try hard to learn, and part of that involves asking questions.
2. This class will be interactive. Be prepared to answer questions during class and participate during class activities and discussions. (Don't worry if you don't "know all the answers".) Come to class, participate, and take good notes.
3. Don't skip class. Just don't. It's bad form. I work hard on preparing for class, to make it compelling and relevant. Plus you miss out on important stuff, and you'll end up falling behind if you skip lecture. So come every time. Come happy, fresh, excited, and ready to think and to participate.
4. Don't cheat. **If you ever feel tempted to cheat, in this class or any other, come and talk to me about it.** It's not wrong to feel tempted, and we can find other ways out of whatever dilemma you're facing without compromising your moral character.

Late policy:

Stuff does happen sometimes, I know. You can't always control it. There are times when you can't complete an assignment on time for reasons that have nothing to do with taking the class seriously. The good news is that these times are very rare, and so here is my policy: **each student, during the semester, will have one free, no-questions-asked extension that they can use for any INDIVIDUAL assignment. (Late passes CAN NOT**

Table of Contents

be used with group assignments.) If you use your extension for an assignment, the assignment in question will be due within 48 hours of the original due date. **This 48 hour period INCLUDES WEEKENDS and BREAKS**. If something is due on Friday at 1pm, you must turn it in by Sunday at 1pm if you use your "late pass". If you need to turn in an assignment late, by using your free extension, please tell me via email on the day that the assignment is due.

ALL OTHER ASSIGNMENTS MUST BE SUBMITTED ON TIME. NO LATE WORK WILL BE ACCEPTED without a late pass.

You cannot barter, trade, or sell your late pass to other members of the class.

Course Work Details

Group Project

Most software development professionals spend their time working in groups. They design software in groups. They split up software tasks and develop the pieces in tandem. Working with others is hard. It's a skill that you need to practice. This course is your first opportunity to practice group work skills.

All group assignments will receive a group grade. You soar or fail as a team. There will be an opportunity towards the end of the course, however, to evaluate yourself and your peers confidentially, and this will factor into your project grade as well. (This allows superstars and slackers to be identified, and to be modestly compensated.)

Writing Assignments

CPSC 330 is a writing intensive course and earns you credits toward the writing-across-the-curriculum graduation requirement. In this course, you will receive significant practice writing in the computer science discipline. There will be 3 formal writing assignments during the semester. All formal writing assignments must be typed.

You must earn an average of 60% or better on the formal writing assignments in order to pass the course.

Students may obtain help with the organization and/or the writing mechanics of any writing assignment from the UMW Writing Center (located on the first floor of Trinkle Hall). You can make up to two visits to the Writing Center with any paper for this course. The Writing Center staff is incredibly friendly and helpful. You're also welcome to schedule appointments with me to discuss writing assignments.

Individual Programs

Individual programming assignments will be used to give you hands-on practice with concepts covered during class-time. There will likely be 3-4 of these assignments. The programs are your time to practice skills in a solo environment. Take advantage of the opportunity. If you put something into your program, you had better understand it inside and out.

Homework

You'll be completing some homework assignments from both the *Heads First Design Patterns* book. Short, informal writing will also be included in the homework category. The assignments are intended to provide practice with and further exploration of topics covered during class and reading assignments. Homework is graded on a credit/no credit basis.

Grade Determination

Final grades will be based upon the following assignments.

Group Project	20%
Writing Assignments	15%
Individual Programming Projects	20%
Homework Assignments	5%
Exams	40%

Course policy stipulates that in order to receive a passing grade in this course, a student must satisfy **all** of the following:

1. Earn an average of 60 or better on all tests and quizzes.
2. Successfully complete all programming projects and earn an average of 60 or better.
3. Successfully complete all formal writing assignments and earn a writing assignment average of 60 or better.

Final letter grades will be determined according to the following departmental scale.

A	92-100%	B+	87-88%	C+	77-78%	D+	67-68%
A-	89-91%	B	83-86%	C	73-76%	D	60-68%
		B-	79-82%	C-	69-72%	F	below 60%

Disabilities:
The Office of Disability Services has been designated by the University of Mary Washington as the primary office to guide, counsel, and assist students with disabilities. If you have a disability that may affect your participation in this course, please contact me as soon as possible to discuss your approved accommodations. I will hold any information you share with me in strictest confidence unless you give me permission to do otherwise.

If you have not made contact with the Office of Disability Services and have reasonable accommodation needs, I will be happy to help you contact them. The office will require appropriate documentation of a disability.

Office of Disability Services - 209 George Washington Hall
540-654-1266
ods@umw.edu

Table of Contents

CPSC 430 Software Engineering Syllabus

Instructor: Jennifer Polack Phone: 654-1362
Office: B19 Trinkle Hall Email:JenniferPolack@gmail.com
Office Hours: M T W R 9-10am, M 12-1

Catalog Description

Techniques for modeling, designing, implementing, and managing large- scale computer programs are studied. Studies include requirements analysis, modeling using UML, and application development with CASE tools. Students work in groups and apply the techniques studied to semester-long projects.

Course Description

Writing a computer program is a challenging and creative experience that is motivated by the desire to solve problems. The task of developing even a small computer program is not an easy one. Programmers are continually required to keep their attention focused on many different aspects of both problems and solutions. This course attempts to help students prepare for application development and their future jobs by teaching object oriented analysis and design, Unified Modeling Language (UML), and Computer Aided Software Engineering (CASE) Tools. This class requires that students work in groups as if a company employed them. Experience in group work is essential to a computer science student's training because in the real world it is rare for programmers to design and program large-scale applications alone.

This course is both writing and speaking intensive. It also satisfies the experiential learning requirement.

Prerequisites: CPSC 330

Course Goals and Objectives

At the end of the semester, you will be able to

- Identify and describe a specific problem requiring a software solution.
- Design an efficient system to solve the problem.
- Use CASE tools to assist in designing and developing a solution.
- Locate and learn to use useful tools.
- Define a timeline for team members and project milestones.
- Present and communicate the current standings of a project.
- Maintain documentation.
- Develop a test plan.

General Education Goals and Objectives

This course satisfies the Experiential Learning Requirement. The following goals are associated with this requirement.

- Students will be able to apply what was learned in coursework to new scenarios outside standard university courses
- Students will be able to identify their personal values and learning goals and direct themselves by creating personalized learning experiences that may include alternative means of learning
- Students will be able to clarify and refine their understanding of their strengths and weaknesses in content of relevant disciplines and in skills such as time management, organization, professionalism, and so forth
- Students will be able to recognize their knowledge and lack of knowledge
- Students will be able to connect their undergraduate experiences and their post-graduation lives

Across the Curriculum Goals and Objectives

Writing Across The Curriculum

- to enhance students' understanding of course material by having them write frequently about that material and (b) to help students become better writers

Speaking Across the Curriculum

- Students will understand and be able to explain the conventions and expectations of oral communication as practiced within the discipline of the course taken.
- Students will apply theories and strategies for crafting messages (verbal, nonverbal, and visual) for particular audiences and purposes.
- Students will be able to craft oral messages after a conscious process in which various options are reviewed and will be able to explain and support their choices.
- Students will be able to metacommunicate about their own communication patterns.

Required Materials

This course will use two required textbooks.

Head First Software Development ISBN-13:
978-0-596-52735-8

UML Distilled
ISBN-13: 978-0321193681
The UML Distilled text is available as an ebook and as a Kindle version.

Course information, selected handouts, and all assignments are available electronically on Canvas.

Class Attendance and Responsibilities

Your attendance is expected at each class meeting. You are responsible for the content of reading assignments, lectures and handouts, as well as announcements and schedule changes made in class. If you must miss a class, check with your instructor or another student to get what you missed.

Attendance is particularly important in this course as presentations and group assignments play key roles in the learning experience. There will be days when your project group will have time to meet or discuss issues during class. If you are not there, it not only affects you, it affects your other group members as well.

Students will be making presentations in class on Thursdays starting on January 30. It is essential that other members of the class are present to provide valuable peer feedback. Additionally, you will learn key facts about other projects during the presentations and watching others will help you to improve your own presentation style. There is a no laptop policy during student presentations.

There is no grade provided for class participation, but at the end of each project cycle all students will have the opportunity to evaluate their other group members' participation and contribution to the project. These evaluations will form 40% of your project grade at the end of the semester.

You are responsible for checking your email DAILY for updates or modifications to assignments, lectures, etc. I use Canvas to send announcements. It's your responsibility to edit your profile in Canvas so that messages are sent to your preferred contact method.

Assignments

All assignments are posted on Canvas. I recommend using the "Syllabus" link which provides a running list of what's due for this course.

Homework/Labs

Occasionally homework and labs will be assigned. Unless the instructor specifies otherwise, the assignment should be completed on your own and turned in on the date indicated. Electronic submissions are preferred unless otherwise stated.

Programming Project & Writing Assignments

You will work in a group on a large software development project. The primary focus of this course will be on the DESIGN and DOCUMENTATION of the project; a secondary focus will be on developing the software for the project. You will be creating a project plan, requirements document, system specification and design, user interface report, and test plan for your project. Additionally, you'll submit a written weekly status report. A list of specific deliverables and due dates is available on Canvas.

Speaking Assignments

As you work on your programming project during the semester, you will have the opportunity to orally share your team's progress with the class. You are required to make two short presentations related to specific topics and your project. A presentation topic will be announced each Thursday beginning on Jan 16. Presentations on that topic will be made on Jan 30 and a new topic will be announced for the following week's presentations. You may select any two presentation weeks during the course of the semester. Due to the large course enrollment, you will be required to sign up for your presentation times. The signup sheet will be maintained in a shared document that will be accessible through Canvas in the Collaborations area. You must make one presentation before fall break. You can choose to make the second presentation before or after break.

Examinations

NO makeup tests will be given except in the case of an unavoidable absence that can be verified as legitimate through the Office of Academic Services. In the case of a planned absence, alternate exam arrangements must be made in advance by notifying the instructor.

Grade Determination

Final grades will be based upon the following assignments.

Exams	35%
Writing Assignments	20%
Individual Presentations	10%
Group Presentation	5%
Final Version of Project	10%
Peer Feedback Collected at Project Phases	10%
Homework/Lab Assignments	10%

Departmental and course policy stipulates that in order to receive a passing grade in CPSC 430, a student must satisfy all of the following:

1. Take all tests and earn an average of 60 or better.
2. Successfully complete all programming projects and earn an average of 60 or better.
3. Complete all writing assignments and earn an average of 60 or better.
4. Complete all speaking assignments and earn an average of 60 or better.
5. Final letter grades will be determined according to the CPSC department grading scale. I reserve the right to lower these standards, but will not raise them.

A	93-100%	B+	87-89%	C+	77-79%	D+	67-69%
A-	90-92%	B	83-86%	C	73-76%	D	60-68%
		B-	80-82%	C-	70-72%	F	below 60%

Midsemester Grades

The University provides the opportunity to provide grading feedback midway through the semester. Any student who earns a 65% or below on an exam, presentation, or writing assignment will receive a midsemester unsatisfactory (U). Additionally, any

Table of Contents

student receiving significant negative peer reviews before the middle of the semester will receive a U. Any student receiving a U in this course should meet with the instructor to develop a performance improvement plan.

The Honor Code and CPSC 430

Programming projects, homework assignments, writing assignments, and exams fall under the conditions of the Honor Code. Individual assignments should be completed individually. Group assignments will require you to consult and work with other members of your group. You should never share information or solutions with members of other groups.

If you have any questions or concerns about how the Honor Code applies to this course, please contact me at any time.

Disabilities

The Office of Disability Services has been designated by the college as the primary office to guide, counsel, and assist students with disabilities. If you receive services through the Office of Disability Services and require accommodations for this class, make an appointment with me as soon as possible to discuss your approved accommodation needs. Bring your accommodation letter with you to the appointment. I will hold any information that you share with me in the strictest confidence unless you give me permission to do otherwise.

If you have not made contact with the Office of Disability Services and have reasonable accommodation needs (note taking assistance, extended time for tests, etc.), I will be happy to refer you. The office will require appropriate documentation of disability.

Tentative Course Schedule

This schedule is meant to provide an overview of topics covered during the semester. All assignments and due dates are posted on Canvas. Refer to the Canvas schedule for precise information.

Week+	Topic/Reading
1	Intro to Software Engineering; Software Process;
2	Requirements Gathering
3	Requirements; Use Cases; User Stories
4	Project Management; Project Scheduling
5	User Interface Design
6	Software Design
7	UML & Class Diagrams; System Models; Exam on Thurs Feb 27
8	Version Control; Implementation

9	More UML & Sequence Diagrams
10	Implementation; Test Driven Development
11	Verification & Validation; Testing
12	Bugs; Additional Topics TBA
13	Additional Topics TBA; Final project due
14 &15	Final Presentations
	Tuesday, April 28 Noon - 2:30 p.m. Classes taught at 11:00 – 12:15 p.m.

Table of Contents

CPSC 440 Game Programming Syllabus

Fall 2008
Tuesday and Thursday 3:30 – 4:55
Professor Jennifer A. Polack-Wahl

Contact Information:

- Phone: 540-654-1318,
- Email polack@umw.edu,
- Office: Trinkle B18

Office Hours:

- TBA

Course Objectives

1. Learn fundamental foundations of the field of Game Programming
2. Sharpen C++ programming skills while implementing concepts in the development of gaming applications
3. Sharpen presentation skills by giving formal presentations of individually designed and implemented games

Required Text

- NA

Grading

Grading Criteria	Weights
Programming Assignments	45%
Presentation	15%
Exams	50%

Final letter grades will be determined according to the following department grading scale. I reserve the right to lower these standards, but will not raise them.

A 92-100%
A- 89-91%
B+ 87-88%
B 83-86%
B- 79-82%
C+ 77-78%

C 73-76%
C- 69-72%
D+ 67-68%
D 60-68%
F below 60%

Table of Contents

Responsibilities

- Your attendance is expected at each class meeting.
- You are responsible for the content of reading assignments, lectures, in-class activities and handouts.
- You are responsible for arriving to class on time to hear announcements and schedule changes.
- If you must miss a class or arrive late, check with me or another student to obtain information that you missed.
- In addition to attending class, you are responsible for checking your campus email for updates or modifications to assignments, lectures, etc.

Programming Projects

- There will be approximately 5 programming projects assigned during the semester. Each programming project should be completed without assistance with other students.
- Please refer to the attached honor code statement for details of the honor code's application to this course.
- Each submitted assignment must compile and run on dev-c paprika. Please compile and run your program on this compiler prior to submission. Programs that do not compile will not be accepted for grading.
- In order to pass this course, all programming assignments must be submitted with an average of 60% or higher.

Examinations

- There will be two in class exams.
- NO makeup tests will be given except in the case of an unavoidable absence that can be verified as legitimate through the Office of Academic Services.
- In the case of a planned absence, alternate exam arrangements must be made in advance by notifying the instructor.

Final Exam

- TBA
- If you have 3 scheduled exams on this date, you may petition to have your exam time moved. Please contact me if you need to make arrangements.

Weekly Schedule (Tentative, subject to change without notice)

Week(s)	Chapter	TOPICS
1	Overview	
	What are games? What it takes to make a game? Game Types.	
2	Starting Points, Design Principles, Video Game Software	
3	Getting Started with Dev-C++ and Allegro	
	Basic 2D Graphics Programming with Allegro	
4	Writing Your First Allegro Game	
5	Programming the Keyboard, Mouse, and Joystick	
6	Introduction to Game Design	
7	Basic Bitmap Handling	
8	Basic Sprite Programming	

9 Advanced Sprite Programming (Animation, Collision Detection)
10 Programming Tile-Based Backgrounds with Scrolling
11 Timers, Interrupt Handlers, and Multi-Threading
12 Creating a Game World: Editing Tiles and Levels
13 Vertical Scrolling Arcade Games
14 Horizontal Scrolling Platform Games
15 Mastering the Audible

CPSC 444 Computer Graphics Syllabus

Professor: Jennifer A. Polack, PHD Term: Summer 2014

Objective:	Computer graphics is concerned with all aspects of producing pictures or images using a computer. We will explore the fundamentals used in computer graphics and use a graphics software system called OpenGL.
Prerequisite:	CPSC 330
Text:	NONE
Website:	http://www.opengl.org http://www.glprogramming.com/red/ The OpenGL Programming Guide - this online version documents older versions of OpenGL but still should be useful

Homework and Programming Assignments: Each assignment will have a specified due date. Work turned in late will incur a penalty of 10% per day, counting weekends and holidays. Thus, in addition to the quality of your work, your grade will be based partly on your ability to turn your work in on time. Homework/programming assignments will be submitted electronically, and must be received by midnight on the stated due date. Independent of late penalties, you will not pass this course unless *all* assignments are turned in.

Grading

50% Exams
50% Programs (May not be weighted equally)

Programming Exercises:

- You may use either C or C++.
- Your programs should compile and link with no warnings.
- You must use Visual C++.
- Your programs are to be written in good programming style and to be readable and understandable. For that purpose, follow the guidelines
  - Documentation:
    - Document your code!
    - In the head of each file write its name and a short description of the file's content.
    - Before declaring or implementing functions / methods describe it's purpose.
    - Document your code whenever you think it's not understandable enough.

    - Make your documentation short and clear.
- Modulation:
    - Don't write too long functions.
    - Divide large blocks of code to several functions, to make things more readable.
    - Design your program before you begin coding it.
    - Avoid code repetition
- Naming conventions:
    - Use meaningful names for functions and variables, even if you have to use long names.
    - Be consistent with your naming conventions. e.g., if you start functions with capital letter, keep it so with all your functions.
    - Note: the conventions of OpenGL, GLU and GLUT are to prefix their functions, types and macros with gl / GL / glu / glut. Therefore do not use these prefixes for the names you give in your code.

Week	Material Covered	Due
Week 1	What is Computer Graphics? Image Formation. An Interactive Introduction	Program 1
	Programming with OpenGL: 1,2,3 Input and Interaction Elementary Rendering	
	Better Interactive Programs Geometry Representation	Program 2
Week 2	Transformation OpenGL Transformations Transformations An Interactive Introduction to OpenGL	Program 3
	Building Models Classical Viewing	
	Computer Viewing Projection Matrices	
Week 3	Lighting	
	Review and Midterm	Program 4
	Shading	
Week 4	Shading in OpenGL	
	Animation and Depth Buffering	Program 5
	Buffers Texture Mapping	

Table of Contents

Week 15	OpenGL Texture Mapping Composition	
	Final Program In-Class Demo	Program 6
	Final Program In-Class Demo	

Out of Class Labs Grading Scheme

	Unsatisfactory	Satisfactory	Good	Excellent
Delivery	• Completed less than 70% of the requirements. • Not delivered on time or not in correct format (disk, email, etc.)	• Completed between 70-80% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)	• Completed between 80-90% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)	• Completed between 90-100% of the requirements. • Delivered on time, and in correct format (disk, email, etc.)
Coding Standards	• No name, date, or assignment title included • Poor use of white space (indentation, blank lines). • Disorganized and messy • Poor use of variables (many global variables, ambiguous naming).	• Includes name, date, and assignment title. • White space makes program fairly easy to read. • Organized work. • Good use of variables (few global variables, unambiguous naming).	• Includes name, date, and assignment title. • Good use of white space. • Organized work. • Good use of variables (no global variables, unambiguous naming)	• Includes name, date, and assignment title. • Excellent use of white space. • Creatively organized work. • Excellent use of variables (no global variables, unambiguous naming).
Documentation	• No documentation included.	• Basic documentation has been completed including descriptions of all variables.	• Clearly documented including descriptions of all variables. • Specific purpose is	• Clearly and effectively documented including descriptions of all variables. • Specific

		- Purpose is noted for each function.	noted for each function and control structure.	purpose is noted for each function, control structure, input requirements, and output results.