

MONTGOMERY COLLEGE: Course Syllabus
Computer Science and Information & Interactive Technologies
CMSC100: Fundamentals of Programming, Prof. J. Joy, Fall 2018

I. Contact Information: Professor J. Joy

Email: Janet.Joy@montgomerycollege.edu (*This is the preferred way to contact me.*) I usually answer my email first thing in the morning and again in the evening. (*It depends on my schedule.*) On weekends it may be less often.

Office: SC 438. **Online with Zoom:** <https://zoom.us/j/4497759354>

Appointments: *just send me 3 blocks of time that work for you and I'll get back to you.*



My contact information and other class materials are available at www.zebra0.com/MC

Your Montgomery College e-mail account is the official means of communication for the college. Blackboard will use this email address to send reminders about overdue projects and other announcements. It is recommended that you check this account routinely. To check your e-mail, log into your MyMC online account and locate the e-mail icon in the upper right hand corner of the page. You can forward your MC email to your other email. Announcements sent from Blackboard may have "Do not respond" as the subject. Please take a look to see if it is important!

II. General Course Information: CMSC 100 - Fundamentals of Programming

Designed for students with no prior programming experience, this course introduces students to fundamental structures of sequence, selection, and repetition, emphasizes solving simple problems using a flowchart. With a high-level language, students code, test, and debug short programs. *Assessment Level(s):* [ENGL 101/ENGL 101A](#), [MATH 050](#), [READ 120](#). *Two hours each week. Formerly CS 100.*

2 semester hours

Accessibility: This section uses Alice3 to create 3D animation. This course is not recommended for students who cannot view and create videos. The videos used in this course have closed captions or a text version is available. Montgomery College accessibility statement:

<http://cms.montgomerycollege.edu/edu/Department.aspx?id=53990>

Preparedness: This is a fully online class. To succeed in this course you should be confident working with a computer, accessing information via the Internet, and using email as a primary means of communication. You should be comfortable with email attachments, troubleshooting an Internet connection, and downloading software. Online courses require extreme self-discipline. One must log on 3-5 times per week and be prepared to read and follow through on assignments and instructions. Students must plan to spend 4-6 hours per week preparing and submitting assignments. Initially, a great deal of time is spent becoming familiar with Blackboard and dealing with technical problems. Technology is unreliable. The plan to submit homework at the last moment can be defeated with a busy or down server.

III. Student Learning Outcomes: Upon course completion you will be able to:

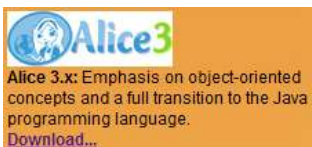
By the end of this course you will be able to:	You will demonstrate this objective through:
1. Define the software development life cycle.	By creating a storyboard for your final project and testing and debugging each of the programming assignments.
2. Describe the core features of programming languages: a) Variables, constants, and data types b) Flow control structures c) Arrays	Each week you will learn new features. You will test your understanding through quizzes. Then you will apply what you have learned by writing programs that use these core features and the ones you have learned previously.
3. Demonstrate the ability to design, develop, and test basic computer programs using the core programming features.	This is demonstrated in all programming projects, progressing from simple to more complex. In your final project you will apply these skills to create an animated movie with sound and other effects.
4. Decompose a software program into functional subprograms (procedures and functions).	As your programs get larger, you will use functions and procedures in order to work on one part at a time. This will make it easier to test and debug your programs.

Learning to program is a cumulative skill. Each week you will learn a few new skills or tools. You will share your thoughts, and what you have created using these new skills in the discussion area in Blackboard. You will also have a chance to see what your classmates have created and get new ideas and techniques from them. You will test your understanding of the new material through quizzes. Finally, you will **apply** what you have learned by creating a program that uses the new tools, along with the skills from previous weeks to create increasingly more complex 3D animation. You will use a variety of control structures to make interesting animations. The control structures allow you to specify which actions to take, whether to perform actions at the same time or in sequence and to specify how many times an action should be performed. As your programs get larger, you will learn to break them into subparts so that you can write, test and debug one piece at a time.

IV. Required materials:

There is NO required text for this class. We will use online materials and tutorials that are available for **free** on the internet. Alice is available for both the Mac and Windows computers. Alice requires the Java Development kit, JDK. The easiest way to install Alice is to download and install Eclipse first. Eclipse is a Java compiler and the installation includes the JDK.

- Download and install Eclipse from <https://eclipse.org/downloads/>
- Download and install Alice 3.5 from <http://www.alice.org> Do NOT download Alice 2.x



Alice 2.x © 1999-2016, Alice 3.x © 2008-2016, Carnegie Mellon University. All rights reserved.
Alice/Zebra0.com © 2016, Joy, J. for zebra0.com. All rights reserved.

V. Grading

Grades are earned, not given.

Course grades will be based upon the following:

- There is a final project for this course in lieu of exams. You will be required to complete a summary of your final project and evaluate 3 other projects by your class peers.
- There will be 10 programming assignments. All assignments must be uploaded to your course assignment page with all files needed to grade the assignment/practice activity to receive credit.
- Each unit has discussion questions in Blackboard. You will be required to give an initial response and you must respond to at least two other students regarding their initial postings. Responses must be substantive, relevant and constructive. Responses must be at least 25 words long using standard academic English. I drop the 3 lowest discussion grades.
- There will be a quiz in Blackboard every 2 or 3 days. I drop the 3 lowest quiz grades.
- All assignments, discussions and quizzes will be due at the end of the unit, postmarked by midnight.
- I usually grade assignments within 2 days once submitted. If I ask for corrections, they must be resubmitted within 2 days.
- All assignments lose 10% each day they are late. They will receive a grade of 0 if more than 1 week late.

The relative weights of these assignments are:

10 Programming projects	@ 5%	50%
12 Discussion questions	@ 1%	12%
19 Quizzes	@ 1%	12%
Final project design		5%
Final Project		21%

A=100-90% B= 89-80% C=79-70% D=69-60% F=60%-below

This is an online course. Active participation in the online activities and completion of all homework and online assignments is required in order to pass this course.

Due Dates: All due dates are in Blackboard. You can check the calendar in Blackboard.

Participation: Students must participate in the Blackboard discussion about 3 or 4 times per week. **If you miss 2 discussions in a row without contacting me, you are subject to being dropped from the class.**

Audit Policy: If you are auditing, you are welcome to participate in the Blackboard discussions and take all quizzes and submit programs but it is not required.

VI Class Policies

Important Student Information Link

In addition to course requirements and objectives that are in this syllabus, Montgomery College has information on its web site (see link below) to assist you in having a successful experience both inside and outside of the classroom. It is important that you read and understand this information. The link below provides information and other resources to areas that pertain to the following: student behavior (student code of conduct), student e-mail, the tobacco free policy, withdraw and refund dates, disability support services, veteran services, how to access information on delayed openings and closings, how to register for the Montgomery College alert System, and finally, how closings and delays can impact your classes. If you have any questions please bring them to your professor. As rules and regulations change they will be updated and you will be able to access them through the link. If any student would like a written copy of these policies and procedures, the professor would be happy to provide them. By registering for this class and staying in this class, you are indicating that you acknowledge and accept these policies.

<http://cms.montgomerycollege.edu/mcsyllabus/>, [Additional Montgomery College policies](#)

VII Resources

Computer problems: As a computer student, you are expected to anticipate potential computer problems. Save often! Keep backups! Allow plenty of time to complete the assignment! Computer problems are not an excuse for submitting an assignment late! I can provide help if you send me a clear explanation of the problem, plus any relevant source files or screen shots.

Netiquette: Etiquette rules for the discussion board.

The discussions are an important part of online classes. Each week you will share ideas with your classmates. You can learn a lot from your classmates and by sharing ideas. Your classmates come from many different cultures and backgrounds. You want to share ideas and tips, not offend. Please read <http://www.zebra0.com/MC/netiquette.pdf> for discussion rules.

Technical Requirements & Technical Support: You will need the following to participate online:

- Regular use of a computer with Internet access and a web browser such as Firefox, Chrome, or Internet Explorer. Expect to spend several hours online each week.
- A web browser such as Firefox, Chrome, or Internet Explorer.
- Speakers or headphone to listen to the videos and to play the sounds you add in Alice.
- See prepare yourself: <http://cms.montgomerycollege.edu/distance/prepare/>
- It is highly recommend that you have internet access at home, however, there are computer labs <http://cms.montgomerycollege.edu/oit/InTech.aspx?id=60795>

For technical assistance with college supported resources, call the Montgomery College IT Service Desk at 240-567-7222 or [://cms.montgomerycollege.edu/EDU/Department2.aspx?id=9356](http://cms.montgomerycollege.edu/EDU/Department2.aspx?id=9356)

Blackboard Help Desk: The **HELP** link on the left-hand course menu links to the **MC Blackboard Online Support Center:**

- Call the Support Center at 240-567-7222 or
- Chat with a service representative, or
- Submit a ticket.

Note: Click the **My Support** link at the top of the Blackboard Online Support Center screen to view a history of your correspondence with the Blackboard Support Center.

System Downtime: The Office of Information Technology conducts computer network maintenance on Sunday morning from 12:01 AM to 6:00 AM each week. During this time you may be not be able to access My MC to login to Blackboard. Do not rely on this time to submit course work.

Distance Learning Support: For all general distance education related questions, contact the Office of Distance Education and Learning Technologies at 240-567-6000 or dl@montgomerycollege.edu.

For all Blackboard and MyMC related questions and issues, contact the IT Service Desk at 240-567-7222 or ITServiceDesk@montgomerycollege.edu or [Blackboard Online Support Center](#).

Class Schedule and Important Dates

In order to provide the best possible learning experiences, these dates may change. Please refer to the resources in Blackboard for any announcements or changes. May 28 is the official start date for this course. All of the learning modules are required. CMSC100: Fundamentals of Programming



[Unit 1: Introduction](#)

This unit should be completed between Tuesday, May 28, 2019 and Thursday, May 30, 2019.

In this module, you will create a program with a 3D character who says "Hello World."

Goals: In this module, you will learn all of the skills necessary to succeed in this course: read the syllabus, introduce yourself to your classmates, and learn about grading and class policies. The goal of this lesson is to download and install the latest version of Alice and run it. You will learn how to navigate in Alice and learn about some of the 3D models that you will learn to animate later in the course. The goal of this lesson is to write your first program in Alice3. You will learn how to create a scene by selecting a template, then add a character and add and edit code.

Objectives:

- Know how to navigate in Blackboard

- Understand the grading policy
- Know how to contact the professor
- Download and install Alice (or find a computer lab with Alice);
- Select a template, such as grass.
- Toggle between the scene editor and the code view windows.
- In the code view window, locate each of the panels
- Name 3 classes in the Alice gallery.
- Be able to search for an object in the gallery.
- Write your first Alice program.
- You will learn to add code.
- Create and modify a scene in the design view.
- Select and add 3D characters to the scene.
- Select, add and modify people.
- Use the arrows to move, rotate, and resize the characters and other objects.
- Move the camera and change the angle.
- Add comments with your name and description of the program.
- Create code to have the character say "Hello World"
- Run the program, test, debug, and save it.

Study and Learn:

- [Course Introduction](#)
- [Introduction to Alice 3](#)
- [Your First program: Hello World](#)



Due in Blackboard by Midnight, Thursday, May 30, 2019:

- Discussion 1: Introductions
- Quiz: Syllabus
- [Final Project: Proposal](#)
- Discussion 2: Reaction to Alice
- Quiz 1: Introduction to Alice
- Discussion 3: Hello World
- Quiz 2: Hello World
- [Program 1: Hello World](#)



[Unit 2: The Virtual World](#)

This unit should be completed between Friday, May 31, 2019 and Sunday, June 2, 2019.

Goals: The goal of this lesson is to learn ways to start a new project, modify, save, save as, and reopen your project and set preferences. The goal of this lesson is to understand how objects view the world and how to move them in the Alice virtual world. You will be able to move, turn and roll the objects that make up your scene. The goal of this lesson is to learn how to change the properties and position of the objects and camera in the Alice virtual world.

Objectives:

- Instead of selecting a template choose other tabs: starters, my projects, recent, file.
- Use the arrows to move, rotate. and resize the characters and other objects.
- Move the camera and change the angle.
- Edit an existing project.
- Add parameters to commands
- Set the preferences to see Java style code
- Understand the basic principles of object motion and rotation
- How Objects see the Virtual World: Video demonstrates move, turn and roll
- Understand that each object has a sense of direction
- Be able to add code to move, turn and roll.
- Use a sequence of actions to create a dance routine, or other complex action.
- Use the handles to turn, roll, and move objects.
- View the scene from starting view, layout, top, side, or front.
- Use different camera views to position objects.
- Change the ground, opacity; add fog, light from above and below.
- Use one shots to move, turn and roll in design view.
- Be able to assign vehicles to objects.

Study and Learn:

- [Start, Save, Reopen and modify your program](#)
- [Understanding the Alice Virtual World](#)
- [Setting up the Alice Virtual World](#)



Due in Blackboard by Midnight, Sunday, June 2, 2019:

- Quiz 3: Start, Save, Reopen
- Quiz 4: Understanding the Alice Virtual World
- Quiz 5: Setting up the Alice Virtual World
- [Project Design: Scenery and Characters](#)

v



[Unit 3: Variables and User Input](#)

This unit should be completed between Monday, June 3, 2019 and Wednesday, June 5, 2019.

Goals: Learn about variables and how to use them. Your program will declare variables and assign them values.
Learning outcome #2a, Learning outcome #3 The goal of this lesson is to learn how to get input from the user and store those values in variables.

You will be able to ask the user for information such as his name and age, and then use that information later in the program.

Objectives:

- Name the different types in Alice.

- Name variables and objects correctly.
- Declare a variable and give it an initial value.
- Use the assign statement to give a variable a new value.
- Select and use functions to ask the user a question and store the result.
- Experiment with the tools available in Alice3.
- Use constants appropriately.
- Select and use functions to ask the user a question and store the result.
- Use functions to get information from the user.
- Use stored values in the code.
- Use input from users in the code.

Study and Learn:

- [Variables in Alice 3](#)
- [Input from the user in Alice 3](#)



Due in Blackboard by Midnight, Wednesday, June 5, 2019:

- Discussion 4: Variables
- Quiz 6: Variables in Alice3
- Quiz 7: User Input in Alice3
- [Program 2: Talk to the User](#)



* / [Unit 4: Math and Poses](#)

This unit should be completed between Thursday, June 6, 2019 and Sunday, June 9, 2019.

Goals: Use arithmetic operators to do some simple math. Move and pose the Alice 3D characters and the subparts such as arms and legs. You will do this in both the scene setup mode and using code. **Learning outcome #3**

Objectives:

- Use assign and math to assign a value.
- Create code to do math using just one step.
- Find a total, percent and other values.
- Use tools in design mode to move, turn and resize the 3D actors in the design mode.
- Write code to move, turn and roll the characters.
- Use tools in design mode to turn and roll the subparts such as arms and legs.
- Implement OneShots to turn and roll the subparts such as arms and legs.
- Write code to turn and roll the subparts. Learning outcome #3
- Be able to create a pose using several subparts

Study and Learn:

- [Arithmetic Operations](#)
- [Posing objects in Alice 3](#)



Due in Blackboard by Midnight, Sunday, June 9, 2019:

- [Project Design: Interactions](#)
- Quiz 8: A Little Math
- Discussion 5: Poses
- Quiz 9: Posing Objects in Alice3
- [Program 3: A Comic](#)

TRUE OR

FALSE? [Unit 5: Boolean Expressions, Animation](#)

This unit should be completed between Monday, June 10, 2019 and Wednesday, June 12, 2019.

Goals: The goal of this lesson is to learn to use Boolean expressions to make comparisons. You will learn to use the if/else control structure to create different paths or branches in the code. In this unit, you will Use the control structures do in order, do together, count and while to create animation. Learning outcome goal #2b, #3

Objectives:

- Write code to input answers of various types from the user.
- Apply if/else control structures to tell the user if his answer is right or wrong.
- Implement a counter to display the score at the end.
- Use the function getBooleanFromUser to get true or false answers.
- Write code to compare values and take appropriate actions.
- Be able to use AND and OR in Boolean expressions.
- Use the function getHeight to find the taller of two animals.
- Utilize control structures in your code.
- Determine which control structures to use for more natural movements
- Implement loop control structures **count** and **while** to wave, clap, shake the head, and stomp the feet!
- Use markers to move an object to a set position.
- Edit object markers: add, delete, re-rename, and change the color
- Create an animation that uses several subparts, such as walking.

Study and Learn:

- [Boolean Expressions: true or false](#)
- [Creating Animation in Alice3](#)



Due in Blackboard by Midnight, Wednesday, June 12, 2019:

- Discussion 6: Boolean Expressions

- Quiz 10: Boolean Expressions
- [Program 4: A Quiz](#)
- Discussion 7: Animation
- Quiz 11: Creating Animation in Alice3
- [Program 5: Animation](#)
- [Project Design: Animations](#)



[Unit 6: Top Down Design](#)

This unit should be completed between Thursday, June 13, 2019 and Sunday, June 16, 2019.

Goals: Learn to design large programs using various tools. You will break the program into small manageable parts to facilitate writing, testing and debugging the program. **Learning outcome goals #1, #3, #4** Create procedures. This will allow you to decompose a program into functional subprograms and create and test larger programs. **Learning outcome goal #4.** You will also use parameters so that we can tell the procedure how fast to wave, or how many times to wave. We can also use parameters to tell who the bunny faces when they wave. This a very powerful tool and will let you greatly improve the efficiency of your code.

Objectives:

- Algorithms: Analyze the problem and determine the best way to achieve the goals.
- Pseudocode: Decompose the problem using step wise refinement
- Storyboards: Create a graphical layout for the design.
- Flowcharts: Analyze the problem using standard symbols to illustrate the sequence of actions.
- Truth tables: Create a table of all possible combinations and the desired outcomes.
- Write code to animate a character, then convert that code to a procedure so that it can be used many times and for other characters.
- Utilize parameters for flexibility.
- Decompose a program into subprograms: myFirstMethod will be mostly calls to procedures and functions from now on.

Study and Learn:

- [The Software-Lifecycle](#)
- [Procedures in Alice3: Top Down Code and Reusable Code](#)



Due in Blackboard by Midnight, Sunday, June 16, 2019:

- Discussion 8: Software Lifecycle
- Quiz 12: The Software-Lifecycle
- [Project Design: Algorithm and Storyboard](#)
- Discussion 9: Procedures and Functions
- Quiz 13: Procedures
- [Program 6: An Exercise Class](#)

$\sqrt{a^2+b^2}$ **Unit 7: Calculations and Random Numbers**

This unit should be completed between Monday, June 17, 2019 and Wednesday, June 19, 2019.

Goals: Learn about random values. **Learning outcome goal #3.** We can use random numbers in many ways. One thing we can do with random users is to flip a coin or select a card from a deck. Those are things that we think of as random. We can also use random numbers to make movement seem more natural. When a person jumps up and down, they don't jump exactly the same amount each time, so we could have them jump a random amount within a range.

Objectives:

- Build arithmetic expressions.
- Use variables, functions, and assign statements.
- Create code to do calculations using multiple variables and steps.
- Write a function to do calculations.
- Create a function to get distance to move next to something
- Use a loop to add numbers uses a running total: total=total + number
- Write code to generate random numbers in 2 different ways.
- Use random values in your animation procedures so that characters move more naturally.
- Use loops to create a number guessing game with hints "too high" and "too low"

Study and Learn:

- [The goal of this lesson is to learn to do calculations, use counters, and find totals. These calculations will enable you to move the characters more precisely in the Alice virtual world.](#)
- [Random Numbers](#)



Due in Blackboard by Midnight, Wednesday, June 19, 2019:

- Discussion 10: Calculations and Functions
- Quiz 14: Calculations
- Discussion 11: Random Numbers
- Quiz 15: Random Numbers
- [Program 7: A Soccer Game](#)



Unit 8: Arrays: List of values

This unit should be completed between Thursday, June 20, 2019 and Sunday, June 23, 2019.

Goals: Declare and use arrays. **Learning outcome goal #2c.** You will create arrays of words, numbers, or objects, then use loops to process the entire array.

Objectives:

- Declare a variable to be an array.
- Create code to assign initial values to an array.
- Use subscripts or index to access the elements of the array.
- Utilize loop control structures to process all of the elements of the array.
- Create and process arrays of different data types, including arrays of objects.
- Understand the purpose of parallel arrays and ways to use them.
- Experiment with arrays of segments, such as the segments of an elephants trunk or tail.

Study and Learn:

- [Arrays](#)



Due in Blackboard by Midnight, Sunday, June 23, 2019:

- Discussion 12: Arrays
- Quiz 16: Arrays
- [Program 8: Arrays](#)



Unit 9: Advanced Techniques and Events

This unit should be completed between Monday, June 24, 2019 and Tuesday, June 25, 2019.

Goals: Use several advanced techniques. **Learning outcome goal #3.** In order to create interesting programs, you will add things like sound, text, background pictures, and other special effects. You will practice working with larger projects and designing, coding, and testing procedures. You will be using many of these techniques in your final project. The goal of this lesson is to learn to make your program more interactive by allowing the user to control the action using the keyboard and the mouse.

Objectives:

- Implement markers to move to specific locations.
- Utilize vehicles to move two objects together.
- Add sound to an Alice project.
- Create billboards and camera positions to change scenes.
- Use text objects to create headlines.
- Write code that uses the argument `asSeenBy` to show movement through the eyes of another object.
- Create scene procedures and change the ground and visibility of objects.
- Adding transitions to scene changes
- Use camera markers to switch from scene on the left and one on the right.
- Add properties to a class.
- Create code to implement event listeners for various events.
- Write code to determine which key on the keyboard was pressed
- Write code to determine which object was clicked.
- Use control structures to specify which actions to take when an event occurs.
- Use `addSceneActivationListener` to create a continuous action while other actions take place.
- Play background music using `initializeEventListeners`.

- Write code that uses default model manipulation in event listeners to let the user drag and drop objects at run time.
- Use collision detection to make something happen when objects collide.
- Write code for the keypress event; for example typing a letter to make a selection.
- Add object mover for an object so that the user can move it with the arrow keys.
- Use the timer event to make an action occur at random intervals.

Study and Learn:

- [Advanced Techniques](#)
- [Events: Responding to the Mouse and Keyboard](#)



Due in Blackboard by Midnight, Tuesday, June 25, 2019:

- Discussion 13: Advanced Techniques
- Quiz 17: Advanced Techniques
- [Program 9: Advanced Techniques](#)
- Discussion 14: Events
- Quiz 18: Events
- [Program 10: Events](#)



[Unit 10: Final Projects](#)

This unit should be completed between Wednesday, June 26, 2019 and Thursday, June 27, 2019.

A Typical Week in CMSC100

An online class requires quite a bit of self motivation. Active participation in the online activities and completion of all homework and online assignments is required in order to pass this course.

A Unit is every 2 or 3 days. Each unit has assigned online tutorials and activities at <http://www.zebra0.com/MC>. After watching the videos and reading the online lecture, you will create one or more programs on your own computer, and then add additional features and enhancements to show your understanding of the material and to personalize the project. After finishing the project you will share your thoughts and ideas with your classmates on Blackboard.

Write your response to the discussion question early in the unit so that there is time to exchange ideas and thoughts with classmates.

Each unit has one or more quizzes in Blackboard that is due at the end of the unit. You can take the quiz again if you miss any questions. Obviously, if you wait until the last day to take the quiz, there isn't much opportunity to take it over. Take the quiz early in the unit so that you can repeat it if necessary.

Check into the discussion board in Blackboard every day to ask questions, answer questions, and respond to your classmates.

The projects and activities are due by midnight. When you make out your schedule for the week, be sure to block out at least 20 hours when you can read and work on a computer!

Start each unit by looking in Blackboard for the assignments and discussion questions. Keep the discussion questions in mind as you study the materials.

You are expected to save all of your work on a Flash drive or other storage device. You are responsible for completing all of the work on time even if your computer crashes.

Course Resources & Technologies

Adobe Acrobat Reader, required: Download and install from <https://get.adobe.com/reader/> Copyright © 2017 Adobe Systems Incorporated. *You probably already have this if you are reading the syllabus. No privacy policy available. Adobe acrobat is accessible with a screen reader.*

Alice 3.1, required: Download and install Alice 3.2 or later from <http://www.alice.org>, Alice 3.x © 2008-2017, Carnegie Mellon University. You do not need to login or register. Alice is available for PC and Mac. No privacy policy is available. Alice is a tool for creating 3D animation. There is no comparable tool for students who cannot see the animation.

Alice Lessons at Zebra0.com, required: Available at <http://zebra0.com/alice>
Author: [Janet E. Joy](#); Publisher: Zebra0.com, This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#)
Privacy policy: <http://www.zebra0.com/resources/privacy.php> All videos have closed captions.

Blackboard, required: Copyright © 1997 - 2017. Blackboard Inc. Login to Blackboard from MyMC. Privacy policy: <http://www.blackboard.com/footer/privacy-policy.aspx> Blackboard is fully accessible.

Eclipse, optional: Download and install from <https://eclipse.org/downloads/> Copyright © 2017 The Eclipse Foundation. *If you have a problem installing Alice with the message that you need the Java JDK, install Eclipse and then retry installing Alice.* Eclipse is accessible. Privacy police: <https://eclipse.org/legal/privacy.php>

Screencast-o-matic, recommended: Download and install or run online from <https://screencast-o-matic.com/> © Screencast-O-Matic , privacy policy: <https://screencast-o-matic.com/privacy> You may use any video recording software except Flash to create the required videos for this class. Screencast-o-matic is a tool for creating videos. It is vision dependent.

Zoom, required: Copyright ©2017 Zoom Video Communications, Inc.
Join from PC, Mac, iOS or Android: <https://zoom.us/j/4497759354>
No privacy policy available. Eclipse is accessible.