Intro To Computer Science

Table of Contents

[Assessment 2](#__RefHeading___Toc1818_1614263175)

[10% Homework 2](#__RefHeading___Toc1820_1614263175)

[10% Quizzes 2](#__RefHeading___Toc1822_1614263175)

[20% Tests 2](#__RefHeading___Toc1824_1614263175)

[15% Participation 2](#__RefHeading___Toc1826_1614263175)

[15% Projects 3](#__RefHeading___Toc1828_1614263175)

[30% Exams 3](#__RefHeading___Toc1830_1614263175)

[Absences 3](#__RefHeading___Toc1832_1614263175)

[Miscellaneous 3](#__RefHeading___Toc1834_1614263175)

[JXFLS Grading Scale 3](#__RefHeading___Toc1836_1614263175)

[Schoolwide Classroom Expectations: 4](#__RefHeading___Toc1850_1614263175)

[Explore Project 4](#__RefHeading___Toc296_2095273207)

[Overview 4](#__RefHeading___Toc298_2095273207)

[Components 5](#__RefHeading___Toc260_2095273207)

[Computational Artifact (CA) 5](#__RefHeading___Toc262_2095273207)

[Effective artifacts include: 5](#__RefHeading___Toc264_2095273207)

[Ineffective Artifacts Include: 5](#__RefHeading___Toc266_2095273207)

[Written Responses (WR) 5](#__RefHeading___Toc268_2095273207)

[Create Project – Computer Science Applications From Ideas 6](#__RefHeading___Toc300_2095273207)

[Overview 6](#__RefHeading___Toc302_2095273207)

[Components 6](#__RefHeading___Toc304_2095273207)

[Students are required to 6](#__RefHeading___Toc306_2095273207)

[Lessons 7](#__RefHeading___Toc246_2095273207)

[Week 1 – Computer Science Big Ideas 7](#__RefHeading___Toc1518_886379824)

[Overview 7](#__RefHeading___Toc1520_886379824)

[Cooperative Learning Structures: 7](#__RefHeading___Toc1522_886379824)

[Quiz 7](#__RefHeading___Toc1524_886379824)

[Homework 7](#__RefHeading___Toc1526_886379824)

[Week 2 – What is Python 7](#__RefHeading___Toc1528_886379824)

[Overview 7](#__RefHeading___Toc1530_886379824)

[Cooperative Learning Structures: 8](#__RefHeading___Toc1532_886379824)

[Quiz 8](#__RefHeading___Toc1534_886379824)

[Homework 8](#__RefHeading___Toc1536_886379824)

[Week 3 – Python Data Types: 8](#__RefHeading___Toc1538_886379824)

[Overview 8](#__RefHeading___Toc1540_886379824)

[Cooperative Learning Structures: 8](#__RefHeading___Toc1542_886379824)

[Quiz: 8](#__RefHeading___Toc1544_886379824)

[Homework: Pair Programming 9](#__RefHeading___Toc1546_886379824)

[Week 4 – Introduction to Abstraction: OOP & Classes 9](#__RefHeading___Toc248_2095273207)

[Overview 9](#__RefHeading___Toc1548_886379824)

[Cooperative Learning Structures 9](#__RefHeading___Toc1550_886379824)

[Quiz 9](#__RefHeading___Toc1552_886379824)

# Assessment

## 10% Homework

Homework completion is very important for a number of reasons. First, it helps you to understand the concepts from the lesson that you understood or did not understand. It gives you questions to present in class the next day that initiates discussion of homework. Second, it helps the teacher to know what concepts you understand. The attention to detailed, written explanations sheds light on the level of understanding that you have of course concepts. Additionally, your detailed work and written explanations set you apart from other students, and it is only those students who work the very hardest and who master the content with deep understanding who deserve to make a top grade of an A in an AP course.

## 10% Quizzes

Quizzes are given periodically and are always announced in advance. If the assignment sheet says that there is supposed to be a quiz, then you will have a quiz whether or not I told you in class that there will be a quiz. Quizzes are typically a combination of multiple choice problems and short answer and possibly one free response problem. Quizzes let me know what you very well and what areas we may need to revisit.

## 20% Tests

All tests will be announced. You will receive point grades for each test. Because tests contain cumulative questions that assess not only material learned for a particular unit but also material from past units, an adjusted scale may be used. It is important to understand that you will only be graded according to this scale if *all of your homework for the unit has been completed and on time*. Any lapse in this expectation and your tests will NOT be graded according to this adjusted scale.

## 15% Participation

Active participation in class activities is essential for your learning. This includes speaking in English, bringing your materials every day, taking good notes, doing out of class assignments for discussion the next day, answering questions when asked, and asking questions when needed. Failure to do any of these things detracts from your learning and the learning of those around you. This is an easy 15% **IF** you do what is expected. Many students lose points here by being unprepared for class, not completing the work, sleeping, not paying attention, interrupting the teacher or other students when they are talking, being disrespectful to others, or interrupting the class.

## 15% Projects

Projects are activities that enhance and deepen content knowledge and may be completed in-class or out-of-class. In either case, projects will require you to create a product. Past projects have included students solving a free-response question, creating a scoring guide for their solution, discussing their scoring guides with their classmates and revising them as needed, comparing them to the AP scoring guide, and then reflecting on the process identifying new insights into how they will be assessed on the AP exam; Writing a paper describing how a course concept can be applied in the real-world; creating a presentation or lesson and teaching a topic to the class.

## 30% Exams

**These are cumulative exams as the course builds on prior learning.**

# Absences

I understand you all have busy lives and are often required to leave school to take various assessments including TOFEL, SAT, and SAT 2. It is **YOUR** responsibility to notify me in advance if you are going to miss class due to a planned absence. It is also **YOUR** responsibility to see me *during office hours* to make arrangements to get the work you will miss. If you have a planned absence and email me after the fact, you will not receive credit for any in-class participation grades and your homework will follow the late work policy. In all cases, any tests or quizzes missed must be taken within two days of your return during your lunch break. Please make the appropriate preparations for this.

If you have an unplanned absence due to illness, you will need to make arrangements to get your missing work either by email, through a classmate, or the day you return. If you are going to be out of school for an extended period of time, I suggest you email me or contact your banzhuren to get the assignments you will miss. As this course is preparing you for the AP exam and indicates to universities that you are capable of doing college level work, ***NOTHING is excused, omitted, exempted, or forgiven in this course that directly relates to the content***. I will not give you a passing grade if you miss three weeks of class and do not complete the content that we covered during that time.

# Miscellaneous

You are required to be an active member in the discussion groups. This is included in your participation grade.

Group projects will be done periodically throughout the course. These are designed to improve your academic language skills and to deepen your understanding of the content. They are not frivolous activities to boost your grade and I take them very seriously. It takes a lot of effort to create and monitor these and you are expected to perform at your best. There are no opportunities to make-up or redo these projects.

Concept maps are required for every major topic in this course (4). They are helpful in developing comprehension and deepening your understanding of the material and for review purposes.

# JXFLS Grading Scale

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A+ 97-100 | B+ 87-89 | C+ 77-79 | D+ 67-69 | F <60 |
| A 93-96 | B 83-86 | C 73-76 | D 63-66 |  |
| A- 90-92 | B- 80-82 | C- 70-72 | D- 60-62 |  |

**The table below highlights each of the six grading categories for the school and examples of what is included in each category from this course**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Class Participation (15%)** | **Homework (10%)** | **Quizzes (10%)** | **Tests (20%)** | **Exams (30%)** | **Projects (15%)** |
| Participation in class discussion. | Weekly Problems | Formal Quiz | Chapter Tests | Quarter Exams | Individual Projects |
| Use of English | AP Free-Response questions | Occasional in-class review quiz | Peer Instruction Assigned Day | Final Exams | Group Projects |
| Problems for Discussion | Interim project work | Homework Quiz |  |  | Concept Maps |
| Note Taking |  |  |  |  | Writing Assignment |
| Being Prepared for Class |  |  |  |  |  |
| Group Interactions |  |  |  |  |  |

# Schoolwide Classroom Expectations:

Student desks are clear of all materials except subject-class learning materials.

Desks are cleared of food and drinks.

White board is cleaned before and after class.

English is spoken at all times.

Students listen when teacher is talking.

Students who talk while the teacher or another student is talking are being disrespectful and are subject to disciplinary action.

Students should follow instructions and directions given.

Students bring all materials required for this class.

Students behave appropriately.

**NOTE:** Nothing in this document supersedes any school policies. You are expected to follow all school policies that are enacted including the late work policy and technology policy. Cell phones are not permitted in this class. Additionally, any decisions by the administration are final. I cannot give you make-up work or allow a re-take if the administration has decided against it.

# Explore Project

### **Overview**

Computer Science innovations impact our lives in ways that require considerable study and reflection for us to fully understand them. In this performance task, students will explore a computer science innovation of their choice. The close examination of a computer science innovation will deepen the students’ understanding of computer science.

In this task, students select and investigate a computer science innovation that has had or has impacts on society, economy, or culture, and consumes, produces, and/or transforms data. A computer science innovation is an innovation that includes an algorithm or program code as an integral part of its function. Examples of AI innovations could include facial recognition software, or adding Bunny Ears on faces when taking a picture.

As students analyze a computer science innovation, they will explain its intended purpose or function, describe harmful and beneficial effects, describe data storage, data privacy, and data security concerns. Students will also produce a **computer science artifact** that illustrates, represents, or explains the innovation’s intended purpose, its function, or its effect, and provide **written responses** to each of the given prompts.

### Components

The following components are formally assessed and must be submitted for the Explore performance task.

#### Computational Artifact (CA)

There are a number of widely available tools students can use to create Computer Science artifacts for this task. A computer science artifact is a visualization, a graphic, a video, a program, or an audio recording that students create using a computer. The students’ creations could solve a problem, show creative expression, or provide the viewer with new insight or knowledge. Students must be able to attest that their computational artifact and written responses they are submitting are their own. Students can use work that is not originally created by them (including but not limited to images, video, or program code segments) as long as they provide appropriate acknowledgments

##### **Effective artifacts include:**

* visual, graphical, and/or audio content to help a reader understand the purpose, function, or effect of a computing innovation; and
* the use of communications media, such as animations, comic strips, infographics, and/or public service announcements, to illustrate the purpose, function, or effect of a computing innovation.

##### Ineffective Artifacts Include:

* artifacts that repeat information supplied in the written responses;
* multi-slide presentations with paragraphs of text or bullets; and
* artifacts that have not been created by the student.

#### Written Responses (WR)

Written responses must be based on relevant, credible, and easily accessible sources. Students are required to provide in-text citations for at least three sources that helped them create their computational artifact and/or formulate their written responses. At least two of the sources must be available online or in print; the third source may be either online, in print, or a personal interview with an expert on the computing innovation. At least two of the sources must have been created after the end of the previous academic year. Students must avoid plagiarism by acknowledging, attributing, and/or citing sources throughout their responses and including a bibliography. Sources that should be acknowledged include text, images, video, music, graphs, and program code that are used in the creation of their computational artifacts

# Create Project – Computer Science Applications From Ideas

## Overview

Data Science is a collaborative and creative process that brings ideas to life through the analysis of data and the creation of models to describe or solve problems. Data Science programs can help solve problems, enable innovations, or express personal interests. In this performance task, students will be developing a program of their choice. The students’ development process should include iteratively designing, implementing, and testing their program. Students are strongly encouraged to work with another student in their class. Students are permitted to collaborate with another student peer. However, a significant portion of the data science program must be developed independently. Students must provide program code segments that they developed independently as part of their written response. Students must be able to attest to the originality of the program code and the written response they are submitting. Students can use program code segments that are not originally developed by them provided they have included appropriate acknowledgment for these code segments.

## Components

The following components are formally assessed and must be submitted for the Create performance task:

* A video of your program running (V)
* Individual written responses about your program and development process (IWR)
* Program code (PC)

## Students are required to

* iteratively design, implement, and test their program;
* independently create at least one algorithm and one abstraction that are central to the purpose of the program and that can be used to meet the requirements for the written response;
* create a video that displays the running of their program and demonstrates its functionality;
* write responses to questions about their program to demonstrate their understanding of programming concepts; and

# Lessons

## Week 1 – **Computer Science Big Ideas**

### Overview

* Introduction (10 minutes)
* Learn about bubble sort and binary search (20 minutes)
* Learn how to find the longest possible time a binary search could take. (10 minutes)
* Learn about **Big Ideas (**Creativity, Abstraction, Data, Algorithims, Programming, Impact, and Internet, Big Ideas) of computer science. (10 minutes)
* Learn about course learning structures (15 minutes)
* Computer Science Big Ideas Quiz (15 minutes)

### Cooperative Learning Structures:

* Kinetic Learning: Standing Birthday Sort (Bubble Sort) & Search (Binary Search)
* Peer instruction – What Learning Structures will we use?

### Quiz

* Computer Science Big Ideas Quiz

### Homework

Sign up to Udacity and start The Intro to Computer Science course.

## Week 2 – **What is Python**

### Overview

* What is programming? (10 minutes)
* What knowledge do you know about the things computer science can do? (10 minutes)
* What programming is not (5 minutes)
* Test out the python interpreter (10 minutes)
* Open Sublime Editor and run first hello world script. (5 minutes)
* Understanding the syntax of python (20 minutes)
* Intro to Python Quiz (20 minutes)

### Cooperative Learning Structures:

* Group Round-robin – Each person states a programming language they know, then share with class.
* Ask, share, exchange – What cool technologies do you know of? (What can Siri do?)
* Lava Seats: When done writing your script get up off your seat and help someone else.

### Quiz

* Python Data Types Quiz

### Homework

Write a python script that instantiates the following variables: your first name, last name, age, dream college, student\_id number, and one embarrassing story. Save it as “myself.py” .

## Week 3 – **Python Data Types:**

### Overview

* What is big data? (10 mintues)
* Video - Humans Need Not Apply (15 minutes)
* Thoughts on the video (5 minutes)
* We will learn about the important python data types (booleans, strings, integers, floats, dictionaries, lists, and tuples). (10 minutes)
* We will create a lebron\_james.py script. (30 minutes)
* Python Data Types Quiz (10 minutes)

### **Cooperative Learning Structures:**

* Free-for-all – Reactions to the Humans Need Not Apply Video
* Pair Think-Write-Share (Why is the term big-data all of a sudden so popular? What’s the excitement?)
* Pair Programming – Make Your Own Data Types tutorial script.
* Peer Instruction – **Student 1** will instruct the class on writing the Lebron James script.

### **Quiz:**

* Python Data Types Quiz (10 minutes)

### Homework:

**Pair Programming:** With a **partner** make your own data\_types.py like script. It should contain the description of a person using all the data types and printing their introduction as we did in class.

## Week 4 – **Introduction to Abstraction: OOP & Classes**

### **Overview**

* What is abstraction? How can we abstract away the details of an NBA Player or another given object/thing/person. (10 minutes)
* Abstracting away the complexity of an NBA with Python Classes (30 minutes)
* We will create a python model/class that takes the complexity out of a real life object, person, or idea. (20)
* Abstraction: Python OOP & Classes Quiz

### **Cooperative Learning Structures**

* Continuous Round-Robin – Come up with real life objects whose details we can abstract away.
* Round Robin – How can we abstract away the details of an NBA Player or another given object/thing/person.
* Peer Instruction – **Student 2** - Abstracting away the complexity of an NBA with Python Classes (30 minutes)
* Pair Programming – Come up with your own abstraction of a real life person, object, or thing.

### Quiz

* Abstraction: Python OOP & Classes Quiz

### Homework

Watch Tutorial on Python Classes & OOP.