**AP Computer Science Principles Course**

**Instructor Notes**

**UNIT NOTES: Introduction to Python Programming Using Processing**

There are many excellent curricula currently available for teaching beginners computer programming. The [Khan Academy Javascript course](https://www.khanacademy.org/cs) and [Codecademy Python course](https://www.codecademy.com/learn/learn-python) are two excellent free online courses. Apple’s *The Swift Programming Language* is a free interactive iBook designed for use by high school students and would be a great choice for classrooms with Apple computers. Any of these courses could be used to teach the programming component of the new AP Computer Science Principles course. Processing gives students a large set of graphics tools that encourages creativity. You could easily mix and match the content of this course with the more traditional Codecademy Python course.

In this unit you will spend roughly one week per programming assignment except for Tron which takes two. There is a multiple choice quiz based on the AP exam every other week.

Whatever programming language and curriculum you choose, here’s what to you will need to cover:

* variables
* arithmetic operators +, -, \*, / and MOD (you will need to teach integer division and modulus)
* logical operators AND OR and NOT
* relational operators < <=, > >=, = (not an assignment), !=
* if statement
* while (or “repeat until”) loops
* lists including the operations insert, append, remove and replace. Note that on the APCSP exam, the first index in a list is 1, not 0 like in Python and JavaScript.
* for each loops
* functions with arguments and return values
* Algorithms, specifically linear and binary search
* Limits of computing, i.e. there are some problems computers can’t solve like the halting problem
* Simulations, including strengths and weaknesses and why we use them
* Data visualizations

**Answers to practice quiz questions**

Slide 72: !. C

Slide 73: 2. API 3. iii

Slide 93: A. 2 B. 1 C. 10 D. 1 E. 0 F. 5

Slide 110: 4

Slide 111: The last line of code should be changed from **DISPLAY(lucky)** to **DISPLAY(luck)** OR the third line of code could be changed **luck <- 11** to **lucky <- 11**

Slide 120: [12,1,-45,37,6]

Slide 127: C

Slide 139: F

Slide 158: C

Slide 178: B

Slide 202: C

Slide 208: D

Slide 224: D

Slide 239: C

Slide 250: B

Slide 254: B

Slide 271: second

 third

Slide 285: x2y

 3

Slide 312: A and E

Slide 327: First

 Fifth

 Seventh

Slide 338: 1. False

 2. True

 3. **mouseX** (varies)

 4. **mouseY** (varies)

 5. **key** (varies)

Slide 347:

def setup():

 size(300,300)

 strokeWeight(2)

def draw():

 background(0)

 stroke(255)

 line(150,150,mouseX,mouseY)

 stroke(0,255,0)

 fill(0,255,0)

 ellipse(average(150,mouseX),average(150,mouseY),40,40)

def average(x,y):

 **return (x+y)/2 #Better than (mouseX + mouseY)/2**

Slide 348:

def setup():

 size(300,300)

def draw():

 background(0)

 stroke(0,255,0)

 fill(0,255,0)

 ellipse(mouseX,mouseY,40,40)

 stroke(255,0,0)

 fill(255,0,0)

 ellipse(half(mouseX),third(mouseY),40,40)

def half(num):

 **return num/2**

def third(num):

 **return num/3**

Slide 349: 1. True

2. False

3. False

4.True

5. False

Slide 361: **True**

 **False**

 **True**

 **False**

Slide 368: 17

Slide 373: 17

Slide 395: C (The others can be calculated with a simple algorithm)

Slide 396: D

Slide 398: E

Slide 401: D

Slide 418: D

Slide 419: A

Slide 420: A and D

Slide 436: 16:00 -122.503728

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Week # (approx five 55min classes) | Lab and/or Activity | PowerPoint slides | Enduring Understanding | Learning Objective | Computational Thinking Practice | Essential Knowledge |
| 1 | [Olympic Rings](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/OlympicRings.odt) | 1 - 33 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP5: CommunicatingP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1E1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.4.1B5.4.1E5.5.1A |
| 2 | [Penguin](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/Penguin.odt) | 33 - 73 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP5: CommunicatingP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1E1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1M5.3.1N5.3.1O5.4.1A5.4.1B5.4.1E5.4.1L5.4.1M5.4.1N5.5.1A |
| 3 | [Lists](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/Lists.odt) | 74 - 133 | 2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages4.2Algorithms can solve many, but not all, computational problems5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language4.2.4 Evaluate algorithms analytically and empirically for efficieny, correctness, and clarity5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: Abstracting | 2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1A4.1.1B4.1.1D4.1.1F4.1.1G4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I4.2.4B5.1.1A5.1.1B5.1.2A5.1.2C5.1.2F5.1.2J5.2.1A5.2.1B5.2.1C5.2.1D5.2.1E5.3.1K5.3.1L5.3.1M5.4.1B5.4.1E5.5.1A5.5.1H5.5.1I5.5.1J |
| 4 | [Make a Repeating Pattern with a while loop](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/RepeatingPattern.odt) | 134-212 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1E1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1M5.4.1B5.4.1E5.5.1A |
| 5 | [Functions and Animations](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/FunctionsAndAnimations.odt) | 213-257 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: Abstracting | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1E1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1M5.4.1B5.4.1E5.5.1A |
| 6 | [User Input](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/UserInput.odt) | 258-352 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.2Create a computational artifact using computing tools and techniques to solve a problem1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.3.1 Use abstraction to manage complexity in programs5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1D1.2.1E1.2.2A1.2.2B1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1A5.3.1B5.3.1C5.3.1D5.3.1D5.3.1E5.3.1F5.3.1H5.3.1I5.3.1J5.3.1K5.3.1L5.3.1M5.3.1N5.3.1O5.5.1A5.5.1G5.5.1H5.5.1I5.5.1J |
| 7-8 | [Tron](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/Tron.odt) | 353-406 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.2Create a computational artifact using computing tools and techniques to solve a problem1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP4: Anayzing Problems and ArtifactsP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1D1.2.1E1.2.2A1.2.2B1.2.3A1.2.3B1.2.3C1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1A5.3.1B5.3.1C5.3.1D5.3.1D5.3.1E5.3.1F5.3.1H5.3.1I5.3.1J5.3.1K5.3.1L5.3.1M5.3.1N5.3.1O5.4.1B5.4.1C5.4.1D5.4.1E5.4.1F5.4.1G5.4.1H5.4.1K5.4.1L5.5.1A5.5.1G5.5.1H5.5.1I5.5.1J |
| 9 | [Computer Simulations](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/ThreeHeadsInARow.odt) (three heads in a row) | 407-425 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages4.2 Algorithms can solve many, but not all, computational problems5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.2.1 Create a computational artifact for creative expression1.2.2Create a computational artifact using computing tools and techniques to solve a problem1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.2.4 Collaborate in the creation of computational artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP3: AbstractingP4: Anayzing Problems and ArtifactsP1: Connecting Computing | 1.2.1A1.2.1B1.2.1C1.2.1D1.2.1E1.2.2A1.2.2B1.2.3A1.2.3B1.2.3C1.2.4A1.2.4B1.2.4C1.2.4D1.2.4E1.2.4F1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1A5.3.1B5.3.1C5.3.1D5.3.1D5.3.1E5.3.1F5.3.1H5.3.1I5.3.1J5.3.1K5.3.1L5.3.1M5.3.1N5.3.1O5.4.1B5.4.1C5.4.1D5.4.1E5.4.1F5.4.1G5.4.1H5.4.1K5.4.1L5.5.1A5.5.1D5.5.1E5.5.1F5.5.1G |
| 10 (1 day) | [Linear and Binary Seary](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/LinearAndBinarySearch.docx) | 426-430 | 4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages4.2Algorithms can solve many, but not all, computational problems | 4.1.1 Develop an algorithm for implementation in a program4.2.1 Explain the difference between algorithms that run in a reasonable amount of time and those that do not run in reasonable time4.2.4 Evaluate algorithms analytically and empirically for efficiency, correctness and clarity | P6: CollaboratingP4: Analyzing Problems and ArtifactsP5: Communicating | 4.1.1H4.2.1B4.2.4A4.2.4D4.2.4H |
| 10 | [Solve a Mystery with a Data Visualization](file:///E%3A/APCSParthurSimonCurriki/IntroToPythonProgrammingWithProcessing/DataVizMystery.odt) | 431-441 | 1.1 Creative development can be an essential process for creating computational artifacts1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.1.3 Computing can extend traditional forms of expression and experience2.2Multiple levels of abstraction are used to write programs or create other computational artifacts4.1Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages4.2Algorithms can solve many, but not all, computational problems5.1 Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society)5.2 People write programs to execute algorithms5.3 Programming is facilitated by appropriate abstractions5.4 Programs are developed, maintained and used by people for different purposes5.5 Programming uses mathematical and logical concepts | 1.1.1 Apply a creative development process when creating computation artifacts1.2.1 Create a computational artifact for creative expression1.2.2Create a computational artifact using computing tools and techniques to solve a problem1.2.3 Create a new computational artifact by combining or modifying existing artifacts1.2.4 Collaborate in the creation of computational artifacts1.3.1 Use computing tools and techniques for creative expression2.2.2 Use multiple levels of abstraction to write programs2.2.3 Identify multiple levels of abstraction that are used when writing programs4.1.1 Develop an algorithm for implementation in a program4.1.2 Express an algorithm in a language5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge5.2.1 Explain how programs implement algorithms5.3.1 Use abstraction to manage complexity in programs5.4.1 Evaluate the correctness of a program5.5.1 Employ appropriate mathematical and logical concepts in programming | P2: Creating Computational ArtifactsP6: CollaboratingP3: AbstractingP4: Analyzing Problems and ArtifactsP5: CommunicatingP1: Connecting Computing | 1.1.1A1.1.1B1.2.1A1.2.1B1.2.1C1.2.1D1.2.1E1.2.2A1.2.2B1.2.3A1.2.3B1.2.3C1.2.4A1.2.4B1.2.4C1.2.4D1.2.4E1.2.4F1.3.1C1.3.1E2.2.2A2.2.2B2.2.3A2.2.3B2.2.3C4.1.1B4.1.1H4.1.1I4.1.2C4.1.2D4.1.2I5.1.1A5.1.1B5.1.2A5.2.1B5.3.1A5.3.1B5.3.1C5.3.1D5.3.1D5.3.1E5.3.1F5.3.1H5.3.1I5.3.1J5.3.1K5.3.1L5.3.1M5.3.1N5.3.1O5.4.1B5.4.1C5.4.1D5.4.1E5.4.1F5.4.1G5.4.1H5.4.1K5.4.1L5.5.1A5.5.1G5.5.1H5.5.1I5.5.1J |