# End-of-Course Assessment Blueprint

## Computer Science Principles

PLTW offers summative End-of-Course (EoC) Assessments that measure both subject-matter knowledge and the real-world transportable skills students need to thrive, no matter which college and career paths they choose.

These assessments are designed to measure knowledge and skills covered in the course curriculum. The content for each EoC assessment is based on an assessment blueprint specifying the extent to which course content will be represented in the assessment. Assessment blueprints were thoughtfully designed by a panel of industry experts, faculty from higher education institutions, PLTW Teachers, and PLTW Curriculum and Assessment Team Members.

Skill clusters were developed for each blueprint to guide the assessment creation process. Skill clusters represent an overarching category of related course objectives measured by the assessment. Table 1 articulates the alignment between the skill clusters and course objectives. In addition, each skill cluster was assigned a weight to determine the degree to which course objectives are addressed in the assessment.

*Table 1: Assessment Blueprint – Computer Science Principles*

| Skill Cluster | Framework Objective | % Weight by Skill Cluster |
|---|---|---|
| Professional Practice | **CAR-1.A:** Describe career paths within the computing specialties.<br>**ERM-1.A:** Abide by professional standards when creating value for people and society.<br>**IOC-1.A:** Explain how an effect of a computing innovation can be both beneficial and harmful.<br>**IOC-1.B:** Explain how a computing innovation can have an impact beyond its intended purpose.<br>**IOC-1.C:** Describe issues that contribute to the digital divide.<br>**IOC-1.D:** Explain how bias exists in computing innovations.<br>**IOC-1.E:** Explain how people participate in problem-solving processes at scale.<br>**IOC-1.F:** Explain how the use of computing could raise legal and ethical concerns.<br>**IOC-2.A:** Describe the risks to privacy from collecting and storing personal data on a computer system.<br>**IOC-2.B:** Explain how computing resources can be protected and can be misused.<br>**IOC-2.C:** Explain how unauthorized access to computing resources is gained. | 5% - 20% |

*(Table continued on next page)*

*Table 1 (continued): Assessment Blueprint – Computer Science Principles*

| Skill Cluster | Framework Objective | % Weight by Skill Cluster |
|---|---|---|
| Collaboration and Communication | **COL-1.A:** Collaborate when processing information to gain insight and knowledge.<br>**COL-1.B:** Apply project management strategies effectively as part of a team.<br>**COM-1.A:** Communicate ideas, processes, and products to optimize audience perception and understanding. | 5% - 20% |
| Problem Solving | **CCP-1.A:** Apply a creative development process<br>**CCP-1.B:** Acknowledge moments where persistence and the positive aspect of failure played an important role in gaining understanding about a problem.<br>**CCP-1.C:** Engage stakeholders in a problem and use their perspectives to shape the course of your development. | 5% - 20% |
| Data, Algorithms, and Development | **DAT-1.A:** Explain how data can be represented using bits.<br>**DAT-1.B:** Explain the consequences of using bits to represent data.<br>**DAT-1.C:** For binary numbers: (a) Calculate the binary (base 2) equivalent of a positive integer (base 10) and vice versa. (b) Compare and order binary numbers.<br>**DAT-1.D:** Compare data compression algorithms to determine which is best in a particular context.<br>**DAT-2.A:** Describe what information can be extracted from data.<br>**DAT-2.B:** Describe what information can be extracted from metadata.<br>**DAT-2.C:** Identify the challenges associated with processing data.<br>**DAT-2.D:** Extract information from data using a program.<br>**DAT-2.E:** Explain how programs can be used to gain insight and knowledge from data.<br>**CRD-1.A:** Explain how computing innovations are improved through collaboration.<br>**CRD-1.B:** Explain how computing innovations are developed by groups of people.<br>**CRD-1.C:** Demonstrate effective interpersonal skills during collaboration.<br>**CRD-2.A:** Describe the purpose of a computing innovation.<br>**CRD-2.B:** Explain how a program or code segment functions.<br>**CRD-2.C:** Identify input(s) to a program.<br>**CRD-2.D:** Identify output(s) produced by a program.<br>**CRD-2.E:** Develop a program using a development process.<br>**CRD-2.F:** Design a program and its user interface. | 35% - 50% |

| | | |
|---|---|---|
| Data, Algorithms, and Development *(Continued)* | **CRD-2.G:** Describe the purpose of a code segment or program by writing documentation.<br>**CRD-2.H:** Acknowledge code segments used from other sources.<br>**CRD-2.I:** For errors in a program: (a) Identify the error. (b) Correct the error.<br>**CRD-2.J:** Identify inputs and corresponding expected output or behaviors that can be used to check the correctness of a program.<br>**AAP-1.A:** Represent a value with a variable.<br>**AAP-1.B:** Determine the value of a variable as a result of an assignment.<br>**AAP-1.C:** Represent a list or string using a variable.<br>**AAP-1.D:** For data abstraction: (a) Develop data abstraction using lists to store multiple elements. (b) Explain how the use of data abstraction manages complexity in program code.<br>**AAP-2.A:** Express an algorithm that uses sequencing without using a programming language.<br>**AAP-2.B:** Represent a step-by-step algorithmic process using sequential code statements.<br>**AAP-2.C:** Evaluate expressions that use arithmetic operators.<br>**AAP-2.D:** Evaluate expressions that manipulate strings.<br>**AAP-2.E:** For relationships between two variables, expressions, or values: (a) Represent using relational operators. (b) Evaluate expressions that use relational operators.<br>**AAP-2.F:** For relationships between Boolean values: (a) Represent using logical operators. (b) Evaluate expressions that use logic operators.<br>**AAP-2.G:** Express an algorithm that uses selection without using a programming language.<br>**AAP-2.H:** For selection: (a) Represent using conditional statements. (b) Determine the result of conditional statements.<br>**AAP-2.I:** For nested selection: (a) Represent using nested conditional statements. (b) Determine the result of nested conditional statements.<br>**AAP-2.J:** Express an algorithm that uses iteration without using a programming language.<br>**AAP-2.K:** For iteration: (a) Represent using iteration statements. (b) Determine the result or side effect of iteration statements. | |

| | | |
|---|---|---|
| | **AAP-2.L:** Compare multiple algorithms to determine if they yield the same side effect or result.<br>**AAP-2.M:** For algorithms: (a) Create algorithms. (b) Combine and modify existing algorithms. | |
| Data, Algorithms, and Development *(Continued)* | **AAP-2.N:** For list operations: (a) Write expressions that use list indexing and list procedures. (b) Evaluate expressions that use list indexing and list procedures.<br>**AAP-2.O:** For algorithms involving elements of a list: (a) Represent using iterative statements to traverse a list. (b) Determine the result of an algorithm with list traversals.<br>**AAP-2.P:** For binary search algorithms: (a) Determine the number of iterations required to find a value in a data set. (b) Explain the requirements necessary to complete a binary search.<br>**AAP-3.A:** Determine the result or effect of a procedure call.<br>**AAP-3.B:** Explain how the use of procedural abstraction manages complexity in a program.<br>**AAP-3.C:** Develop procedural abstractions to manage complexity in a program by writing procedures.<br>**AAP-3.D:** Select appropriate libraries or existing code segments to use in creating new programs.<br>**AAP-3.E:** For generating random values: (a) Write expressions to generate possible values. (b) Evaluate expressions to determine the possible results.<br>**AAP-3.F:** For simulations: (a) Explain how computers can be used to represent real-world phenomena or outcomes. (b) Compare the use of simulations with real-world contexts.<br>**AAP-4.A:** For determining the efficiency of an algorithm: (a)Explain the difference between algorithms that run in reasonable time and those that do not run in reasonable time. (b) Identify situations where a heuristic solution may be more appropriate.<br>**AAP-4.B:** Explain the existence of undecidable problems in computer science. | |

*(Table continued on next page)*

*Table 1 (continued): Assessment Blueprint – Computer Science Principles*

| Skill Cluster | Framework Objective | % Weight by Skill Cluster |
|---|---|---|
| The Internet | **CSN-1.A:** Explain how computing devices work together in a network.<br>**CSN-1.B:** Explain how the Internet works.<br>**CSN-1.C:** Explain how data is sent through the Internet via packets.<br>**CSN-1.D:** Describe the differences between the Internet and the World Wide Web.<br>**CSN-1.E:** For fault-tolerant systems, like the internet: (a) Describe the benefits of fault-tolerance. (b) Explain how a given system is fault-tolerant. (c) Identify vulnerabilities in a system.<br>**CSN-2.A:** For sequential, parallel, and distributed computing: (a) Compare problem solutions. (b) Determine the efficiency of solutions.<br>**CSN-2.B:** Describe benefits and challenges of parallel and distributed computing. | 5% - 20% |