

What is Understanding by Design®?

Source:

The Understanding by Design Guide to Creating High-Quality Units by Grant Wiggins and Jay McTighe

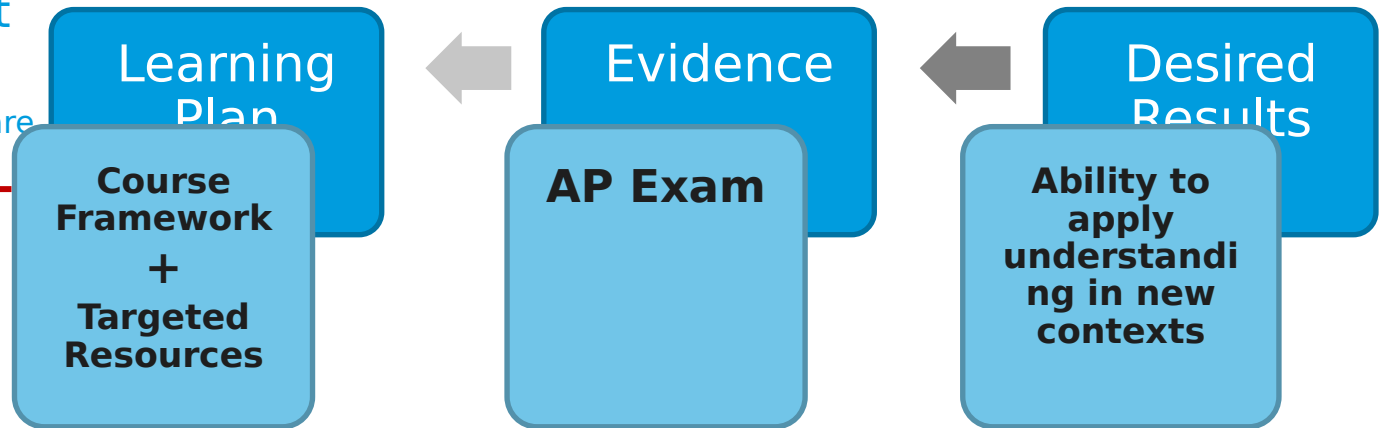
Understanding by Design® and UbD™ are trademarks owned by ASCD.

Helps to avoid:

- ✓ “textbook coverage”
- ✓ “activity-oriented teaching”

Understanding by Design (UbD) is a way of thinking purposefully about curricular planning by transforming content standards and learning goals into concrete elements of instruction and assessment.

A “UbD approach” means that curriculum is “planned backward” from long-term desired results through a three-stage process:



Understanding by Design

Turn to page 25 of the *AP Course and Exam Description*. Define each of the following terms.

- Essential Knowledge (EK) –

Essential knowledge statements describe the knowledge – including facts, definitions, etc. – required to perform the learning objectives.

- Learning Objectives (LO) –

Learning objectives define what a student should be able to do with content knowledge in order to progress toward the enduring understanding.

- Enduring Understanding (EU) –

Enduring understandings are the long-term takeaways related to the big ideas that leave a lasting impression on students. They build and earn these understandings over time by exploring and applying course content throughout the year.

- Big Idea (BI) –

The big ideas serve as the foundation of the course and develop understanding as they spiral throughout the course. The big ideas are the recurring themes that are present in the course. They are crosscutting concepts spiraled throughout the curriculum.

TOPIC PAGES
Suggested skills offers one or more possible skills related to the topic.

Where possible, available resources are listed that might help teachers address a particular topic in their classroom.

Enduring understandings are the long-term takeaways related to the big ideas that leave a lasting impression on students. Students build and earn these understandings over time by exploring and applying course content throughout the year.

Learning objectives define what a student should be able to do with content knowledge to progress toward the enduring understandings. Each learning objective is aligned to a particular skill, and that skill appears in a colored box after the learning objective.

Essential knowledge statements describe the knowledge required to perform the learning objective.

Exclusion statements define content or specific details about content that does not need to be included in the course. The content in the exclusion statements will not be assessed on the AP Computer Science Principles Exam.

AP Computer Science Principles Course and Exam Description

CSP
CED
pp. 25

Topic Pages

- **Required course content** is listed in the center of the page, in a blue box.
- The **skill** for each learning objective is listed in the upper corner. Teachers are encouraged to have students engage with this skill while learning the content of the topic.
- Skills from **Practice 6** are not explicitly aligned with a set of learning objectives and can be paired with any content to be taught across the course.
- **Optional resources** to help teach the topic are listed in the side margin. These are hyperlinked in the online version of the CED.

What students must be able to do within the context of the learning objective

SKILLS
1.C
Explain how collaboration affects the development of a solution.

6.A
Collaborate in the development of solutions.

AVAILABLE RESOURCES

- External Resources >
- Collaboration Tools from Cornell University Center for Teaching Innovation
- 4 Methods to Enhance Student Collaboration in the Classroom from Concordia Portland Online

Required Course Content

ENDURING UNDERSTANDING
CRD-1
Incorporating multiple perspectives through collaboration improves computing innovations as they are developed.

LEARNING OBJECTIVE
CRD-1.A
Explain how computing innovations are improved through collaboration. **1.C**

1.C
must do with the content

Long-term takeaway students should have

ESSENTIAL KNOWLEDGE

- CRD-1.A.1**
A computing innovation includes an integral part of its function.
- CRD-1.A.2**
A computing innovation can be physical (e.g., self-driving car), non-physical computing software (e.g., picture editing software), or non-physical computing concepts (e.g., e-commerce).
- CRD-1.A.3**
Effective collaboration produces a computing innovation that reflects the diversity of talents and perspectives of those who designed it.
- CRD-1.A.4**
Collaboration that includes diverse perspectives helps avoid bias in the development of computing innovations.
- CRD-1.A.5**
Consultation and communication with users are important aspects of the development of computing innovations.

What students must know

32 | Course Framework V.1

**BIG
IDEA
1**

Creative Development

10–13% AP Exam Weighting

1.1 Collaboration

1.2 Program Function and Purpose

TEACH
COMPUTATIONAL THINKING

COMPUTATIONAL THINKING PRACTICES

Practices spiral across big ideas.

- 1** Computational Solution Design
- 2** Algorithms and Program Development
- 3** Abstraction in Program Development
- 4** Code Analysis
- 5** Computing Innovations
- 6** Responsible Computing

20

**BIG
IDEA
2**

Data

17–22% AP Exam Weighting

2.1 Binary Numbers

2.2 Data Compression

2.3 Extracting Information from Data

2.4 Using Programs with Data

Topic Questions

Multiple-choice: ~20 questions

**BIG
IDEA
3**

Algorithms and Programming

30–35% AP Exam Weighting

3.1 Variables and Assignments

3.2 Data Abstraction

3.3 Mathematical Expressions

3.4 Strings

3.5 Boolean Expressions

3.6 Conditionals

3.7 Nested Conditionals

3.8 Iteration

3.9 Developing Algorithms

3.10 Lists

3.11 Binary Search

3.12 Calling Procedures

3.13 Developing Procedures

3.14 Libraries

3.15 Random Values

3.16 Simulations

3.17 Algorithmic Efficiency

3.18 Undecidable Problems

Topic Questions

Multiple-choice: ~90 questions
Performance Task: ~20 prompts

**BIG
IDEA
4**

Computer Systems and Networks

11–15% AP Exam Weighting

4.1 The Internet

4.2 Fault Tolerance

4.3 Parallel and Distributed Computings

Topic Questions

Multiple-choice: ~10 questions

**BIG
IDEA
5**

Impact of Computing

21–26% AP Exam Weighting

5.1 Beneficial and Harmful Effects

5.2 Digital Divide

5.3 Computing Bias

5.4 Crowdsourcing

5.5 Legal and Ethical Concerns

5.6 Safe Computing

Topic Questions

Multiple-choice: ~20 questions

CSP
CED
pp. 20 -
21

Activity: Topic 1.2 Program Function and Purpose

Each page is intended to represent a teachable “chunk” of content – a chunk that will require teachers to plan which unit / module it will be taught in.

THINK PAIR SHARE

- What are the most valuable pieces of information portrayed on the topic page?
- What parts of the topic page represent required course content or skills?
- What parts of the topic page represent suggested or non-required content?

BIG IDEA 1 Creative Development

10–13% AP Exam Weighting

1.1 Collaboration

1.2 Program Function and Purpose

1.4 Identifying and Correcting Errors

Required Course Content

ENDURING UNDERSTANDING

CRD-2
Developers create and innovate using an iterative design process that is user-focused, that incorporates implementation/feedback cycles, and that leaves ample room for experimentation and risk-taking.

LEARNING OBJECTIVE

CRD-2.A
Describe the purpose of a computing innovation. **1.A**

ESSENTIAL KNOWLEDGE

CRD-2.A.1
The purpose of computing innovations is to solve problems or to pursue interests through creative expression.

CRD-2.A.2
An understanding of the purpose of a computing innovation provides developers with an improved ability to develop that computing innovation.

CRD-2.B
Explain how a program or code segment functions. **4.A**

CRD-2.B.1
A program is a collection of program statements that performs a specific task when run by a computer. A program is often referred to as software.

CRD-2.B.2
A code segment is a collection of program statements that is part of a program.

CRD-2.B.3
A program needs to work for a variety of inputs and situations.

CRD-2.B.4
The behavior of a program is how a program functions during execution and is often described by how a user interacts with the program.

20 | Course Framework V.1

AP Computer Science Principles

Big Idea Opener

- **Developing Understanding** provides an overview that contextualizes and situates the key content of the big idea within the scope of the course.
- The **Building Computational Thinking Practices** section describes specific aspects of the skills that are appropriate to focus on in that big idea.
- The **Preparing for the AP Exam** section provides helpful tips and common student misunderstandings identified from prior exam data.
- The **essential questions** are thought-provoking questions that motivate students and inspire inquiry.

AP COMPUTER SCIENCE A

BIG IDEA 1 10-13% AP EXAM WEIGHTING

Creative Development

Preparing for the AP Exam

Students will be expected to design and implement a program of their choice for the Create performance task. While students select their own topic for this task, they are required to include certain elements, such as lists and procedures, in their program code. Providing students with exemplars may help them consider the types of programs that can be developed while still meeting this requirement.

Students will need practice identifying and correcting errors to prepare AP Exam. One way to give students this practice is to provide them with prewritten program code to correct.

ESSENTIAL QUESTIONS

CRD-1

Building Computational Thinking Practices

1.B 3.A 4.A 4.C

When designing a solution to a problem, programmers consider both the problem itself and the way the user will interact with the program: the user interface. A well-designed user interface makes it easy for the user to understand and use the program. A well-designed user interface is required as input for the program to complete its tasks.

CRD-2

When creating diagrams of a program, students will benefit from considering what they want their program to do and how they want it to be identified inputs. Planning ahead helps them determine what abstract concepts are developed and can help identify errors early in development.

When implementing program design, programmers often use documentation to explain the purpose of various code segments and describe how they function together in the program. Students' diagrams

AP 10-13% AP EXAM WEIGHTING

Big Idea at a Glance



AP COMPUTER SCIENCE PRINCIPLES

Computational Thinking Practices: Skills

Practice 1	Practice 2	Practice 3	Practice 4	Practice 5	Practice 6
Computational Solution Design 1	Algorithms and Program Development 2	Abstraction in Program Development 3	Code Analysis 4	Computing Innovations 5	Responsible Computing 6
Design and evaluate computational solutions for a purpose.	Develop and implement algorithms.	Develop programs that incorporate abstractions.	Evaluate and test algorithms and programs.	Investigate computing innovations.	Contribute to an inclusive, safe, collaborative, and ethical computing culture.

SKILLS

<p>1.A Investigate the situation, context or task.</p> <p>1.B Determine and design an appropriate method or approach to achieve the purpose.</p> <p>1.C Explain how collaboration affects the development of a solution.</p> <p>1.D Evaluate solution options.</p>	<p>2.A Represent algorithmic processes without using a programming language.</p> <p>2.B Implement an algorithm in a program.</p>	<p>3.A Generalize data sources through variables.</p> <p>3.B Use abstraction to manage complexity in a program.</p> <p>3.C Explain how abstraction manages complexity.</p>	<p>4.A Explain how a code segment or program functions.</p> <p>4.B Determine the result of code segments.</p> <p>4.C Identify and correct errors in algorithms and programs, including error discovery through testing.</p>	<p>5.A Explain how computing systems work.</p> <p>5.B Explain how knowledge can be generated from data.</p> <p>5.C Describe the impact of a computing innovation.</p> <p>5.D Describe the impact of gathering data.</p> <p>5.E Evaluate the use of computing based on legal and ethical factors.</p>	<p>6.A Collaborate in the development of solutions.</p> <p>6.B Use safe and secure methods when using computing devices.</p> <p>6.C Acknowledge the intellectual property of others.</p>
--	--	---	--	---	---

BIG IDEA 1
Creative Development

10–13% AP Exam Weighting

1. Collaboration
2. Program Function and Purpose

BIG IDEA 1

Creative Development

BIG IDEA AT A GLANCE

Learning Objective	Topic	Skills	Unit/Module
1. Collaboration CSD-1.A, CSD-1.B, CSD-1.C	1.1 Collaboration	<p>1.C Explain how collaboration affects the development of a solution.</p> <p>6.A Collaborate in the development of solutions (not assessed).</p>	
2. Program Function and Purpose 1. CSD-2.A, 2. CSD-2.B	1.2 Program Function and Purpose	<p>1.A Investigate the situation, context, or task.</p> <p>3.A Generalize data sources through variables.</p> <p>4.A Explain how a code segment or program functions.</p>	

Skills

Unit/Module

<p>1.C Explain how collaboration affects the development of a solution.</p> <p>6.A Collaborate in the development of solutions (not assessed).</p>	
--	--

Skill

Identify and address any student misunderstandings.

30 | Course Framework V.1

AP Computer Science Principles Course Framework

CSP
CED
p. 30



based on the learning objective-skill pairing.

Sample Instructional Activities

- The Sample Instructional Activities page includes **optional activities** that can help tie together the content and skills of a particular topic.
- The activity descriptions are not meant to serve as prescriptive sets of steps; rather, **teachers should feel free to modify the activities** in ways that would best suit the needs of their students.
- The sample activities listed use common **instructional strategies**, all of which are listed in the Instructional Approaches section of the CED.

BIG IDEA 1

Creative Development

SAMPLE INSTRUCTIONAL ACTIVITIES

The sample activities on this page are optional and are offered to provide possible ways to incorporate instructional approaches into the classroom. They were developed in partnership with teachers from the AP community to share ways that they approach teaching some of the topics and skills in this big idea. Please refer to the Instructional Approaches section.

Activity	Topic	Sample Activity
1	1.1	Sharing and responding <p>Have students develop a list of three questions that they would like to use data to answer. Then, in small groups, ask each student to share one of their questions. The group will respond with feedback to improve the quality of the question. Students should take turns sharing their questions until all questions have been considered. Finally, ask each group to come to a consensus on which three questions they will answer with data.</p>

Language and Logic of Computing: Algorithmic Thinking, Teaching and Assessing
Module in the Professional Development section of AP Classroom for a more detailed lesson plan and video example.

Big Idea Planning Notes

Use the space below to plan your approach to the topics in this big idea. Consider what resources and instructional strategies you might want to use.

.....

.....

.....

AP Computer Science Principles Course and Exam Description

**CSP
CED
p. 31**

Topic Pages

- Each topic page represents **a teachable “chunk” of content and skills**. This chunk might take anywhere from slightly less than a single class period to several class periods.
- Teachers and endorsed providers will need to scaffold the **content** and **skills** found in each topic across their units / modules.

BIG IDEA 1

SKILLS
1.A
Explain how collaboration affects the development of a solution.
1.A.1
Collaborate in the development of solutions.

AVAILABLE RESOURCES

- External Resources >
 - Collaboration
- Tools from Cornell University Center for Teaching Innovation
- 4 Methods to Enhance Student Collaboration in the Classroom from Concordia Portland Online

Creative Development

TOPIC 1.1
Collaboration

Required Course Content

ENDURING UNDERSTANDING
CRD-1
Incorporating multiple perspectives through collaboration improves computing innovations as they are developed.

LEARNING OBJECTIVE
CRD-1.A
Explain how computing innovations are improved through collaboration. **1.A.1**

ESSENTIAL KNOWLEDGE
CRD-1.A.1
A computing innovation includes a program as an integral part of its function.
CRD-1.A.2
A computing innovation can be physical (e.g., self-driving car), non-physical computing software (e.g., picture editing software), or non-physical computing concepts (e.g., e-commerce).
CRD-1.A.3
Effective collaboration produces a computing innovation that reflects the diversity of talents and perspectives of those who designed it.
CRD-1.A.4
Collaboration that includes diverse perspectives helps avoid bias in the development of computing innovations.
CRD-1.A.5
Consultation and communication with users are important aspects of the development of computing innovations.

continued on next page

32 | Course Framework V.1

AP Computer Science Principles Course and Exam Description

BIG IDEA 1

Creative Development

LEARNING OBJECTIVE
CRD-1.A
Explain how computing innovations are improved through collaboration. **1.A.1**
CRD-1.B
Explain how computing innovations are developed by groups of people. **1.B**
CRD-1.C
Demonstrate effective interpersonal skills during collaboration. **1.C**

ESSENTIAL KNOWLEDGE
CRD-1.A.1
Information gathered from potential users can be used to understand the purpose of a program from diverse perspectives and to develop a program that fully incorporates these perspectives.
CRD-1.B.1
Online tools support collaboration by allowing programmers to share and provide feedback on ideas and documents.
CRD-1.B.2
Common models such as pair programming exist to facilitate collaboration.
CRD-1.C.1
Effective collaborative teams practice interpersonal skills, including but not limited to:

- communication
- consensus building
- conflict resolution
- negotiation

AP Computer Science Principles Course and Exam Description

**CSP
CED
pp. 32-33**

9