# Chief Reader Report on Student Responses:

## 2017 AP® Computer Science Principles

- Number of Students Scored: 44,330
- Number of Readers: 335
- Score Distribution:

| Exam Score | N | %At |
|---|---|---|
| 5 | 6,115 | 13.8 |
| 4 | 9,607 | 21.7 |
| 3 | 17,320 | 39.1 |
| 2 | 8,101 | 18.3 |
| 1 | 3,187 | 7.2 |

- Global Mean: 3.17

The following comments on the 2017 free-response questions for AP® Computer Science Principles were written by the Chief Reader, Paul Tymann, Professor, Rochester Institute of Technology. They give an overview of each free-response question and of how students performed on the question, including typical student errors. General comments regarding the skills and content that students frequently have the most problems with are included. Some suggestions for improving student preparation in these areas are also provided. Teachers are encouraged to attend a College Board workshop to learn strategies for improving student performance in specific areas.

| **Task:** Create Performance Task | **Topic:** Applications of Ideas | |
|---|---|---|
| | **Mean Score:** | **Max. Points:** |
| **Developing a Program with a Purpose** (weighted 20%) | 2.83 | 3 |
| **Developing a Program with a Purpose** (weighted 20%) | 2.18 | 3 |
| **Applying Algorithms** (weighted 30%) | 1.46 | 3 |
| **Applying Abstraction** (weighted 30%) | 1.43 | 3 |

***What were students expected to demonstrate in their response to this question?***

Students were expected to:

- Design and create a program for a particular purpose. The program may be used to solve a problem or for students' own personal creativity.
- Describe the development process of the program, focusing on two distinct points in that process. Students were asked to discuss difficulties or opportunities that were encountered during the development and how those difficulties were resolved or incorporated into the program.
- Demonstrate their understanding of how to develop and implement algorithms in a program. In particular, students were expected to recognize and use algorithms as building blocks by integrating and combining them to create a new algorithm.
- Use abstraction in the development of their program to manage complexity. Students needed to identify abstraction in their program and explain how this abstraction helped to manage the complexity of their program, or how the program might function differently if this abstraction was not used.

***How well did the students address the course content related to this question? How well did the studentsintegrate the skills required on this question?***

- Students were asked to create a video that demonstrated the running of at least one feature of their program. Students were able to create a video that clearly and accurately illustrated the purpose of the program.
- Students were asked to describe the process they used to develop the program, focusing on two distinct points in the process. While most students were able to identify an opportunity or a difficulty, several did not indicate clearly how these were addressed.
- Students were asked to select an algorithm that integrates two or more algorithms, accurately identify its purpose, and describe how the two algorithms work both independently and in combination. While many students were able to select an algorithm that integrates two or more algorithms, they struggled to explain in detail what each algorithm does independently and in combination in the program.
- Students were asked to select an abstraction that integrated mathematical and/or logical concepts and explain how this abstraction served to manage complexity of the program. While many students were able to select an appropriate abstraction, they struggled to write a detailed explanation of how the abstraction manages complexity and why this is important in the program.

***What common student misconceptions or gaps in knowledge were seen in the responses to this question?***

| *Submission Requirement* | *Common Misconceptions/Knowledge Gaps* | *Responses that Demonstrate Understanding* |
|---|---|---|
| 1, 2a: Developing a Program with a Purpose | • Submitting screenshots of program code, or static images of the program will result in a low score. Student videos should clearly show a significant portion of the program running.<br><br>• Failing to describe the purpose of the program clearly, either in the video or in the written response. | • High-scoring videos clearly showed any text contained in the video in a form that was easy to read. Sound or music in videos was clear. Narration was a highly effective method to help readers understand what they were seeing in the video. |
| 2b: Developing a Program with a Purpose | • Referring to working with a partner or "we" throughout an entire response. This will result in a score of 0. Students are required to work independently on a significant portion of their program code, and must include a description of at least one opportunity or difficulty they encountered while programming independently.<br><br>• Writing exclusively about time spent brainstorming ideas prior to beginning the design and implementation of the program.<br><br>• Some responses neglected to discuss the incremental and iterative development process of their entire program.<br><br>• Describing difficulties with the creation or submission of the video was not part of the development process.<br><br>• Students must identify a difficulty and an opportunity encountered during the development of their program and clearly indicate what was done to address the situation. | • High-scoring submissions included details about portions of program code that weren't functioning properly. The descriptions included how errors were identified and resolved, either independently or by working with a partner.<br><br>• High-scoring submissions clearly described what was done in the development process. For example, "I used a debugger to determine that my loop count was off by one. I corrected the corresponding code and the program worked as expected," as opposed to "I fixed a problem in my program." |

| | | |
|---|---|---|
| 2c: Applying Algorithms | • The response must identify three algorithms;the selected algorithm and the two algorithms that are included in the selected algorithm.<br><br>• Selection of algorithms that do not integrate two or more algorithms. For example, while an if/else might incorporate two algorithms (e.g., processing data vs. error handling). The if/else by itself does not, but it could incorporate two others (and likely will). | • High-scoring submissions included algorithms that integrated at least two different algorithms, described what each algorithm does independently, then described how the two algorithms worked together to complete a task. Submissions can include the code segments for each algorithm individually with their description, and then the code for how the algorithms work together to create a new algorithm.<br><br>• High scoring submissions explicitly identified the algorithm. Responses that highlighted more than the selected algorithm, resulted in lower scores because the reader could not clearly determine the code segment containing the algorithm. |
| 2d: Applying Abstraction | • Provided individual lines contained in the abstraction without explaining the purpose of the abstraction or how it worked to manage complexity in the program.<br><br>• The abstraction must be developed by the student and must have the potential for re-use to manage complexity.<br><br>• The selected abstraction failed to include mathematical or logical concepts. | • High-scoring submissions include a comparison of how the program would function with and without the abstraction, as well as illustrate how the abstraction works to manage complexity in the program.<br><br>• High-scoring submissions explicitly identified the abstraction. Responses that highlighted more than the selected abstraction, resulted in lower scores because the reader could not clearly determine the code segment containing the abstraction. |

***Based on your experience at the AP® Reading with student responses, what advice would you offer to teachers to help them improve the student performance on the exam?***

Developing a Program with a Purpose – Submission Requirement 1 and 2a

- Ensure that students have access and opportunity to practice on how to use video recording equipment prior to administering this task. A video consisting of screen shots does not satisfy this requirement.
- Students need practice creating videos so that any text contained in the video is clearly visible and readable.
- Students need practice creating videos so that any sound or music included in the program can be clearly heard.
- Submissions with narration were highly effective in aiding understanding of what was shown in the video.

Developing a Program with a Purpose – Submission Requirement 2b

- Students were expected to write a significant portion of their program code independently. Any program code written by another person should be acknowledged by including the author's name in the comments.

Applying Algorithms and Applying Abstraction – Submission Requirement 2c and 2d

- Students can and should include multiple program code segments to demonstrate the full utility of their algorithm and abstraction. These program code segments may come from more than one area of the program code. For example, if the algorithm calls a procedure, the student should also include the program code segment that includes the procedure in their submission.

***What resources would you recommend to teachers to better prepare their students for the content and skill(s) required on this question?***

Suggested Resources Include:

- Teachers should provide the Reproducible for Students section of the Course and Exam Description to students. The submission template and scoring guidelines are helpful tools to use alongside the official performance task directions and guidelines, but are not an adequate substitute.
- Have students use a journal to record their development process, including when iteration occurred during the process. This will help students recall the incremental and iterative process used when developing their program. The journal is also a helpful tool when completing the written response.
- The Assessment Overview section of the Course and Exam Description provides a section called "Preparing for the Task." This section provides teachers with a suggested series of steps that students can use to develop their program.

| **Task:** Explore Performance Task | **Topic:** Impact of Computing Innovations | |
|---|---|---|
| | **Mean Score:** | **Max. Points:** |
| **Criteria 1** (weighted 20%) | 0.97 | 1 |
| **Criteria 2** (weighted 10%) | 0.97 | 1 |
| **Criteria 3** (weighted 15%) | 0.94 | 1 |
| **Criteria 4** (weighted 15%) | 0.24 | 1 |
| **Criteria 5** (weighted 15%) | 0.63 | 1 |
| **Criteria 6** (weighted 15%) | 0.15 | 1 |
| **Criteria 7** (weighted 10%) | 0.50 | 1 |

***What were students expected to demonstrate in their response to this question?***

Students were expected to:

- Investigate and analyze a computing innovation that has or has had both beneficial and harmful effects on society, the economy, or culture. The computing innovation should consume, produce, and/or transform data, and raise a data concern.
- Create a computational artifact that represents a computing innovation's purpose, function, or effect.
- Determine and justify, with evidence, beneficial and harmful effects of a computing innovation.
- Analyze the data a computing innovation uses, manipulates, or produces, as well as any storage, privacy, or security concerns connected to the computing innovation.

***How well did the response address the course content related to this question? How well did the responses integrate the skills required on this question?***

- Students were asked to investigate a computing innovation and create a computational artifact to represent its intended purpose, function, or effect. With the use of video and video presentations, students were able to create computational artifacts that conveyed a computing innovation's purpose, function, or effect. Some students struggled to convey a computing innovation's purpose, function, or effect when using other mediums.
- Students were asked to identify and describe both a beneficial and a harmful effect of a computing innovation. While most students were able to identify beneficial effects of a computing innovation, many struggled to fully describe the effect and provide specific evidence tying it to society, economy, or culture. Additionally, many students did not include a harmful effect of a computing innovation.
- Students were asked to identify the data used in a computing innovation and describe how it used or transformed this data. Additionally, students were asked to describe, in detail, a data storage, privacy, or security concern. While most students were able to identify the data used or stored by a computing innovation, many struggled to describe in detail how it used or transformed this data. Students were also able to identify a data concern, but many did not connect this concern to a computing innovation.

***What common student misconceptions or gaps in knowledge were seen in the responses to this question?***

| *Submission Requirement* | *Common Misconceptions/Knowledge Gaps* | *Responses that Demonstrate Understanding* |
|---|---|---|
| 1, 2a: Computational Artifact | • Some students presented innovations that were not computing innovations. A computing innovation is one that includes a computer or program code as an integral part of its function. According to this definition, a self-driving car would be a computing innovation, whereas a fiber optic cable would not be. The response should clearly identify and describe the computer or program code that is part of the innovation.<br><br>• Confusion between technology innovations, such as a car, with computing innovations, such as the computers and sensors used in a self-driving car.<br><br>• Use of images and videos taken from the Internet to create a computational artifact without providing any acknowledgement that they were not the student's work. This is considered plagiarism, and will result in a score of 0 on the entire performance task. | • High-scoring submissions included descriptions of physical computing innovations such as Google glasses, non-physical computing software such as cell phone applications, and e-commerce, which relies on transactions conducted on the Internet.<br><br>• All submissions must include acknowledgement of the source or author of any information or evidence taken from the work of someone else. Acknowledgements can be done by adding a citation to the computational artifact itself, by adding a credits page to a video, or by including these with the references in submission requirement 2e. |
| 2b: Development Process | • Written description about how to run a presentation or video, rather than the steps taken to create the presentation or video. For example, a written description on how to run a Prezi presentation, including how to advance the Prezi to the next animation. | • High-scoring submissions demonstrated understanding of the design process by describing the steps to create the computational artifact; sometimes being iterative in nature, and included how specific features of the computing tool helped to achieve a desired effect on the computational artifact. |

| | | |
|---|---|---|
| 2c: Analyzing Impact of Computing | • Sometimes responses confused the effect of the use of a computing innovation and an outside influence that might serve to break the innovation. Some examples of harmful effects that would not be considered direct effects of a computing innovation include the cost of a computing innovation (e.g., being too expensive) or the ability of a computing innovation to be hacked.<br><br>• Stating a fact about a computing innovation, without making a claim as to whether it is a beneficial or harmful effect. For example, stating that e-commerce encourages people to shop at home rather than going to a store is not enough to conclude this is a beneficial effect on society. | • High-scoring submissions included a clear and detailed analysis of harmful and beneficial effects of a computing innovation, including the consequences of using an innovation and tying this to society, economy, or culture. These submissions included evidence to support their claims. For example, a submission claiming the benefits to our economy of self-driving cars could support the economic impact by including the following statement: "According to the Association of Unmanned Vehicle Systems International (AUVSI), the cost of a car crash is $576M and 42 lives daily." [3] Since self-driving cars are designed to obey traffic laws, the response can further explain how self-driving cars avoid fatalities and losses. Therefore, the self-driving car impacts the economy by saving the user from unforeseen financial hardship in the form of car repairs and medical and legal expenses.<br><br>• High-scoring submissions explicitly tie their claim to society, economy, or culture, and support the claims by providing evidence. For example, e-commerce has a negative impact on society by encouraging shoppers to become addicted. This addiction happens because "features that emphasize attractive product stimuli … may make shoppers less attentive to their purchasing behavior by generating a sense of |

| | | |
|---|---|---|
| | | excitement about the product that banishes reasoned self-observation of the damage to one's credit card balance."[2] |
| 2d: Analyzing Data and Information | • Neglecting to demonstrate understanding of how a computing innovation uses and transforms data. For example, a submission might identify that a computing innovation stores video, but it neglects to explain how the video is being used.<br><br>• Stating a data privacy concern, such as hacking, without an analysis that connects the concern to a computing innovation while offering a description of how the data may be used to violate privacy rights.<br><br>• Confusion between the devices used to collect data and the data itself. For example, a camera on a self-driving car is not data, but the images that cameras collect are the data. | • High-scoring submissions identify the type of data used by a computing innovation and demonstrates the understanding of how a computing innovation stores and processes this information. Submissions use evidence to support the analysis of how a computing innovation is using the data. For example, a submission might identify that a computing innovation stores video of car license plates, and further explains that "the number and letter images on license plates are scanned and matched with on-board, real-time databases" [4] to match current police bulletins for wanted criminals.<br><br>• High-scoring submissions incorporate an analysis of data storage, data privacy, or data security aspects of a computing innovation. For example, these submissions go beyond stating the data security concern, such as an app that stores patient information. This app may be meant to help first responders, but patient information is protected under HIPAA. A security breach could result in patient information being "passed on to a lender who could then deny the patient's application for a home mortgage…" [1] |

| | | |
|---|---|---|
| 2e: References | • Neglecting to cite references in the written responses, or providing ambiguous citations.<br><br>• Not providing attributions for the sources of information in their written responses, either by saying, "According to…," including the name of a source and/or link, or quoting and putting a numeral to the citation. | • See examples in previous row of this table. |

References:

[1] "188-Why is the HIPAA privacy rule needed," HHS.gov, 2015. [Online]. Available: http://www.hhs.gov/hipaa/for-professionals/faq/188/why-is-the-privacy-rule-needed/index.html. Accessed: Sep. 8, 2016.

[2] R. LaRose Ph.D., Journal of Computer-Mediated Communication, 2001. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.116.2169&rep=rep1&type=pdf. Accessed: Sep. 8, 2016.

[3] H. Logic, "Daily impact of self driving cars in the United States - association for unmanned vehicle systems international," in AUVSI All Things Unmanned. [Online]. Available: http://www.auvsi.org/auvsiresources/knowledge/dailylossesinaworldwithoutselfdrivingcars. Accessed: Sep. 8, 2016.

[4] D. Matson, "Automatic license plate recognition (ALPR) scanning systems," in Experienced Criminal Lawyers, Experienced Criminal Lawyers, 2010. [Online]. Available: http://www.experiencedcriminallawyers.com/articles/automatic-license-plate-recognition-alpr-scanning-systems/. Accessed: Sep. 8, 2016.

***Based on your experience at the AP® Reading with student responses, what advice would you offer to teachers to help them improve the student performance on the exam?***

- Using Development Process and Tools – Submission Requirement 1 and 2a
    - Students must use a computing tool to create a computational artifact. Some examples of computing tools used in the creation of a computational artifact include, but were not limited to, Screencast-O-Matic, Movie Maker, Word, Photoshop, cell phone video recording, cell phone audio recordings. A digital picture of a non-computational artifact is not considered computational.

- Using Development Process and Tools – Submission Requirement 2b
    - Students should explicitly state what computing tool they are using to create their computational artifact.

- Finding and Evaluating Information – Submission Requirement 2e
    - Students must provide attributions for the sources of information in their written responses. Either by saying, "According to…," including the name of a source and/or link, or quoting and putting a numeral to the citation.

***What resources would you recommend to teachers to better prepare their students for the content and skill(s) required on this question?***

Suggested Resources Include:

- Reproducible for Students section of the Course and Exam Description must be provided to students. The submission template and scoring guidelines are helpful tools to use alongside the official performance task directions and guidelines, but are not an adequate substitute.
- The Assessment Overview section of the Course and Exam Description provides a section called "Preparing for the Task". This section provides teachers with a suggested series of steps students can be used to develop their program.
- The Instructional Approaches section of the Course and Exam Description provides a section on Supporting Claims with Evidence which is useful when providing instruction on citing resources used in computational artifact and written responses.