2019

Fostering computational thinking in technology and engineering education: an unplugged hands-on engineering design approach

Euisuk Sung
CUNY New York City College of Technology

feature article

Figure 5. An example of student prototypes.

fostering computational thinking in technology and engineering education:

an unplugged hands-on engineering design approach

This article, through the C-Boat lesson model, shows how an engineering design task can promote computational thinking in an engineering design context.

by Euisuk Sung

Introduction

Computational thinking has been popularized in the last decade, particularly with the emphasis on coding education in K-12 schools. The core idea of computational thinking has a close relationship with technology and engineering education (TEE). Jeannette Wing (2008) introduced the term *computational thinking* as, "taking an approach to solving problems, designing systems, and understanding human behavior that draws on concepts fundamental to computing" (p. 3717). Wing holds the view that computational thinking is a universal attitude and skill set that facilitates a human thought process similar to the approaches taken by a computer scientist. The concept of computational thinking is not a completely new idea. Similar concepts, such as computer literacy and information computer technology (ICT) education, have already been discussed and practiced in K-12 education (Papert, 1980).

It is obvious that the increasing interest in computational thinking brings great opportunities to technology and engineering educators. Hacker (2017) argued about the relationship between computer science and technological literacy and concluded that technology and engineering teachers can contribute to enhancing students' computational thinking skills and knowledge without major modifications of the TEE content. In fact, TEE has emphasized the use of computing skills to solve problems, and integrative STEM education encourages the adoption of math and science to solve engineering problems. Therefore, this article will examine the relationship between computational thinking and TEE and address a way to teach computational thinking using an engineering-design instruction model.

Computational Thinking in TEE

Technology and engineering education (TEE) has a long connection with the concept of computational thinking. K-12 TEE inherently associates with various fields of industry, including computer science. ITEA/ITEEA (2000/2002/2007) defined the term technology as, "the act of making or crafting, but more generally it refers to the diverse collection of processes and knowledge that people use to extend human abilities and to satisfy human needs and wants" (p. 2). Using the broader meaning of technology, computing is an application of technological activities, and computational thinking is a process and skill set that people have developed and accumulated to solve real-life problems.

Many researchers argued that computational thinking is a mental process, and therefore not necessarily obtained through learning computer programming or computer science (Lu & Fletcher, 2009; Lye & Koh, 2014; Wing, 2008). Voogt, et al. (2015) suggested that teaching computational thinking could be implemented through several forms in K-12 education, such as: an entirely separate subject, within cross-curricular practices, or as an afterschool program.

Computational Thinking in STEM Education

Standards for Technological Literacy (*STL*) did not directly mention the term computational thinking (ITEA/ITEEA, 2000/2002/2007). Instead *STL* used the term mathematical thinking as a concept similar to computational thinking. For example, *STL* Standard 2-W endorsed systems thinking, explained how technology education can use computational thinking to improve technological systems, and noted: "Students should have opportunities to use simulation or mathematical modeling, both of which are critical to the success of developing an optimum design" (p. 41). In addition, Standard 3-J described the relationship between computational thinking and mathematical thinking: "The mathematical and scientific ideas applied in the development of these digital devices promoted further developments that resulted in new tools, such as computer modeling" (p. 52). Although these standards did not mention directly the term computational thinking, they showed that the nature of TEE encourages students to develop and use computational thinking approaches.

Key Elements of Computational Thinking

1. Abstraction and automation.
2. Systematic process of information.
3. Symbol systems and representations.
4. Algorithmic notations of flow control.
5. Structured problem decomposition.
6. Iterative, recursive, and parallel thinking.
7. Conditional logic.
8. Efficiency and performance constraints.
9. Debugging and systematic error detection.

(Grover & Pea, 2013)

Table 1. Key Elements of Computational Thinking.

Next Generation Science Standards (*NGSS*) also noted computational thinking as an integral building block of science learning (NGSS Lead States, 2013). The *NGSS* framework stated that students should use the computational thinking approach to identify relationships of physical variables and to predict the behavior of the possible consequences. In addition, *NGSS* placed computational thinking as a core method of learning science, assigning the Scientific and Engineering Practices section as one of the core science teaching strategies. For example, *NGSS* MS-PS4-1 states, "Mathematical and computational thinking at the 6–8 level builds on K–5 and progresses to identifying patterns in large data sets and using mathematical concepts to support explanations and arguments" (p. 47).

What are the Core Concepts of Computational Thinking?

A variety of research on computational thinking has been conducted in K-12 education. Research on computational thinking can be broadly grouped into two fundamental questions: (1) What is the core component of computational thinking? and (2) How can young students develop computational thinking abilities? Glover and Pea (2013) reviewed the recent computational thinking research studies and summarized nine key elements of computational thinking (Table 1).

Although most of the elements in Table 1 were written in computer science terms, Grover and Pea (2013) noted that the imple-

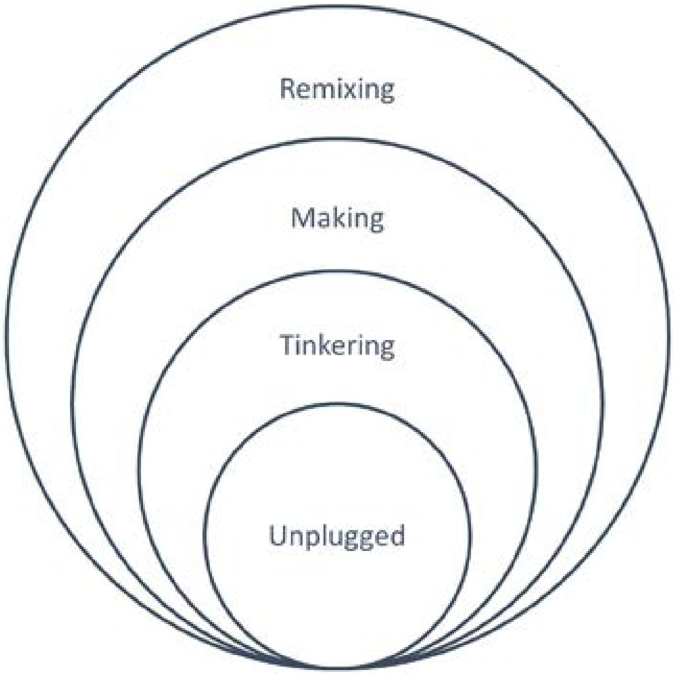


Figure 1. Four practical experiences of computational thinking. (Kotsopoulos et al, 2017, p. 6).

mentation of computational thinking practices can be achieved through multiple approaches and does not necessarily require the use of computers. For example, *abstraction* and *automation* are essences of computational thinking, and an *algorithm* is a key procedure to build an abstraction. In computer science, *abstraction* represents the process of generalizing that simplifies complex phenomena or problem-solving procedures (Lee, et al., 2011). A computer system is made up of thousands of subsystems, and building one subsystem is called an abstraction. The abstraction benefits people and enables them to use the complex computer system without knowing how the internal system works. Wing (2008) positioned abstraction as a core component of computational thinking and asserted that students should learn the concept of abstraction. Wing believed that learning the concept of abstraction helps students develop higher-order thinking abilities to think like a computer scientist.

Fostering Computational Thinking Through an Unplugged Design Activity

Kotsopoulos, et al. (2017) conducted research on the computational thinking framework used in K-12 education and presented four pedagogical experiences to teach computational thinking (Figure 1). Kotsopoulos pointed out that learning a programming language can be a barrier when students learn computational thinking. Therefore, when students first learn concepts of computational thinking, teachers might need to start providing them with a variety of problem-solving experiences using algorithms, rather than learning a computer language first. In fact, many educators often focus on teaching technical skills to operate electronic devices such as *Vex Robots*, *Arduino*, *Raspberry Pi*, or *LEGO Mindstorms* without teaching fundamental elements of computer science. Although college level students majoring in computer science need to use programming languages to learn advanced computational thinking, young students need to start learning the basic thought processes and core elements of computational thinking like a computer scientist.

Building Computational Thinking Through an Unplugged Approach

This article will present a practical approach to teach computational thinking strategies to TEE students using an unplugged learning experience. The unplugged approach has been addressed to teach core concepts of computer science through engaging games, puzzles, or solving real-life problems, using simple computer algorithms (CS Unplugged, n.d.). The lesson, titled *C-Boat*, was designed to incorporate a hands-on engineering design activity with computational thinking practices. In the *C-Boat* lesson, students will use the computational thinking approach to accurately predict the outcome of their solution in the problem-solving process. Key to the engineering design process is the ability to predict results prior to testing prototypes (ITEA/ITEEA, 2000/2002/2007). The *C-Boat* lesson consists of two design parts: (1) Design a boat model with a trial-and-error approach; (2) Redesign the boat model through a computational thinking practice. Below is the design brief for the *C-Boat* lesson.

In the first-round design phase, students will design a boat model that needs to hold 20 golf balls. The design task provides

C-Boat Problem Statement

A boat manufacturer is seeking to create a new line of instant emergency boats that are portable for hikers. The company would like to hire you to design a boat for this purpose. To test your design ability, a senior designer will provide you with a sheet of aluminum foil (40 X 30cm) and ask you to build a boat model that could hold at least 20 golf balls afloat on water. The boat with the 20 golf balls should not become submerged or sunk. In addition, you have to identify the maximum capacity of your boat model. You will have 10 minutes to design your model that will demonstrate your design ability.

Your Task

Before you start designing the boat model, identify the following items suggested in the problem statement:

1. Who is the client?
2. What is the problem?
3. What is the criteria?
4. What are the constraints?

Figure 2. *C-Boat* Design Brief.

students with ten minutes for designing and the material needed—a sheet of aluminum foil, size 40 X 30 cm. The first round of the design activity does not provide any scientific knowledge or computational thinking practice. The teacher simply provides the design brief to students, so students use intuitive skills to solve the problem. When finished building a model, students should test how many golf balls the model will hold and record the results.

Intervention: Computational Thinking and Making

After the completion of the first-round design, the *C-Boat* design instruction provides students with computational thinking practice. The science concept embedded in this design problem is buoyancy. The concept of buoyancy can be explained through Archimedes' principle. Archimedes found an upward force when the body of an object is immersed in a fluid and explained that the buoyance force of the fluid is equal to the amount of water displaced by the object. So, when we put an object into the water, the height of the water level increases because the water is displaced by the volume of the object. In addition, when an object is denser than water, it will sink in water. Conversely, an object will float if its density is less than water, as shown in Figure 3.

In normal conditions, the density of water is 1 ml/g (military/gram), and the formula for the density of water can be represented through the density formula triangle in Figure 4.

When computer scientists solve problems, they often create or use algorithms that consist of various optimal logics of problem solving. In this design task, students could build an algorithm that illustrates several steps of procedural logics to accurately predict the maximum capacity of their boat design. Technology and Engineering teachers encourage students to build an algorithm using guiding questions such as those presented in Table 2 (page 12). For example, the guiding questions (1, 2, and 3) allow students to simplify the design requirements and set the variables presented in the design brief. Guiding questions 4 and 5 help students create formulas for problem solving and lead them to illustrate the formula in a visual chart.

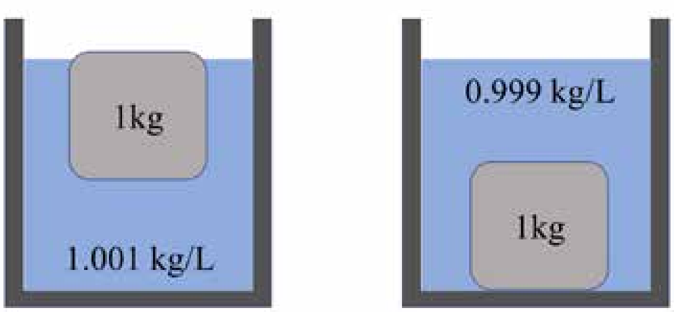


Figure 3. Water displacement by the weight of an object.

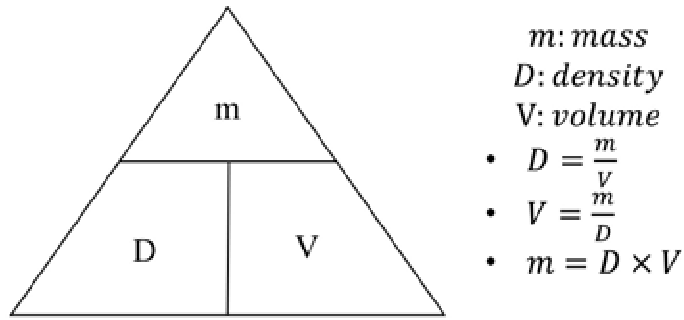


Figure 4. Density formula triangle.

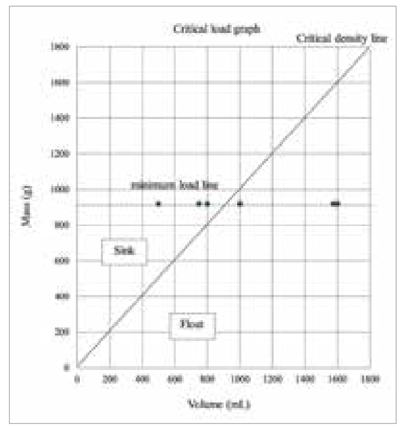
Building a clear algorithm helps students not only simplify a complex set of logics in an *abstraction*, but also helps them focus on the next level of problem solving. In computer science, simplifying a big problem into smaller pieces is often called *problem decomposition* or *divide and conquer*. Additionally, a well-defined algorithm is equivalent to a flowchart of problem solving. Usually, programming starts with building a flowchart that illustrates specified logic from the initial stage to the final stage.

Using the guiding question strategy, the *C-Boat* instruction could provide students with a strategy to more accurately predict the maximum capacity of the boat model and develop their computational thinking as well. This practice provides an accurate mathematical prediction of the maximum capacity and also guides the thinking processes that can create a mathematical algorithm to solve the given problem. In this design task, the boat must have a capacity of at least 920 cm^3 of volume (assume the mass of one golf ball is 46 g).

Conclusion: Incorporating Unplugged Design Activity with Computational Thinking

The computational thinking movement is emerging along with a rapidly developing intellectual revolution. Bundy (2007) wrote an article, "Computational Thinking is Pervasive," and claimed that our computer-dependent society not only calls for students to have the ability to use computer technologies in an effective way, but also requires them to change the way they think in order to cope with new types of problems. This article, through the *C-Boat* lesson model, shows how an engineering design task can promote computational thinking in an engineering design context. A number of ways have been addressed to develop computational thinking. The unplugged approach shows that TEE can promote computational thinking abilities by modifying existing engineering design lessons. In addition, this unplugged approach can serve as an important stepping stone to the "plugged in" approaches such as *Vex Robots*, *Arduino*, *Raspberry Pi*, or *LEGO Mindstorms*.

Table 2. An Example of Computational Thinking Practices.

Guiding Questions	Use of Computational Thinking
1) What is the weight of a golf ball?	Weight of a golf ball using a scale. e.g., $m_{golf\ ball} = 46g$
2) What is the total weight of 20 golf balls?	Build a formula to get the total weight. $m_{golf\ ball} \times 20 = 46g \times 20 = 920g$
3) What is the formula to get the volume of a boat that holds 920g?	Using the density triangle, complete the volume formula that meets the critical load. • $V_{critical\ load} = \frac{920g}{D_{water}}$, $D_{water} = \frac{1g}{1ml}$ • $V_{critical\ load} = 920cm^3$, $1ml = 1cm^3$
4) What is the minimum length, depth, and height for the critical load?	Compute the volume of your design model. • $920cm^3 < L_{cm} \times D_{cm} \times H_{cm}$
5) How would the critical load be illustrated in a graph using the formula? The graph should include two lines that represent critical density numbers and minimum loads.	Illustrate a graph that represents the critical density of water and minimum loads. 
6) What is the theoretical number of golf balls that your boat model loads?	• $m_{max\ load} = D_{water} \times V_{raft\ model}$ • e.g.) L = 15cm, D = 15cm, H = 5cm, $V_{LXDXH} = 1125cm^3$ • $m_{max\ load} = 1125g$ • $Predicted\ number\ of\ golf\ balls = \frac{1125g}{46g} = 24.46$

Furthermore, the lesson can be used to teach the relationship between technology and other fields of study such as mathematics and science. Many engineering students believe using mathematics or science simply provides a correct answer. In the *C-Boat* design instruction, the following formulas enable a mathematical solution that can be seen as a true solution.

$$46g \times 20 = 920g,\ V = \frac{960g}{D_{water}},\ D_{water} = 1g/cm^3$$

$$V_{boat\ model} > 90cm^3$$

However, the mathematical formulas only provide the minimum requirements for solving the engineering problem. These calculations do not guarantee the success of problem solving. In fact, the boat model could fail if it has any structural shortage, like a hole in the bottom or a leak in a corner of the folded aluminum foil. If the boat model loses balance while loading golf balls, it may sink. Similarly, the *C-Boat* design activity also shows many of the challenging aspects of engineering design, such as failure, uncertainty, constrants, and optimization (Koen, 2003; Petrosky, 2006). This lesson shows how students apply mathematics and scientific knowledge and explains the relationship between scientific/mathematical knowledge and engineering design.

Acknowledgement

The lesson used in this article was developed as part of an NSF-funded project titled TRAILS (DRL – 1513248), but any opinions and findings expressed in this material are the author's and do not necessarily reflect the views of NSF. The author wishes to acknowledge Dr. Kelley, Dr. Bartholomew, and Dr. Strimel at Purdue University for helping develop and refine the lesson idea.

References

CS Unplugged. (n. d.). Computer science unplugged trailer. Retrieved from http://csunplugged.com

Grover, S. & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field, *Educational Research, 42*(38), 38-43, doi: 10.3102/0013189X12463051.

Hacker, M. (2017). Is computer science compatible with technological literacy? *Technology and Engineering Teacher, 77*(4), 8-14.

International Technology Education Association (ITEA/ITEEA). (2000/2002/2007). *Standards for technological literacy: Content for the study of technology.* Reston, VA: Author.

Koen, B. V. (2003). *Discussion of the method: Conducting the engineer's approach to problem solving.* New York: Oxford University Press.

Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A pedagogical framework for computational thinking. *Digital Experiences in Mathematics Education, 3*(2), 154-171. DOI 10.1007/s40751-017-0031-2.

Lee, I., Martin, F., Denner, J., Coulter, B., Allen, W., Ericson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads, 2*(1), 32-37.

Lu, J. J. & Fletcher, G. H. (2009). Thinking about computational thinking. *ACM SIGCSE Bulletin, 41*(1), 260-264.

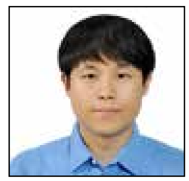
NGSS Lead States. (2013). *Next generation science standards: For states, by states.* Washington, DC: The National Academies Press.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas.* Basic Books, Inc.

Petrosky, H. (2006). *Success through failure: The paradox of design.* Princeton University Press.

Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies, 20*(4), 715-728.

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 366*(1881), 3717-3725. doi:10.1098/rsta.2008.0118.

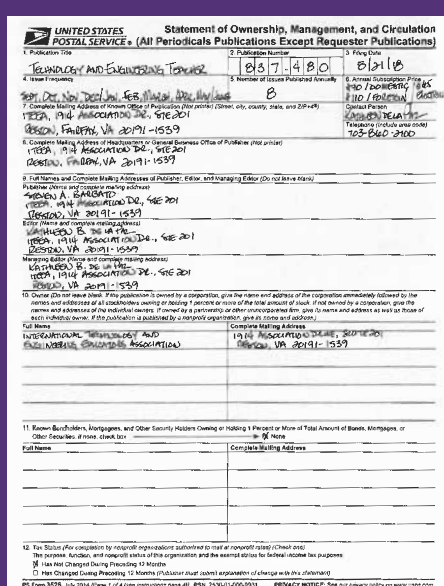


Euisuk Sung, Ph.D. *earned his doctoral degree in Engineering and Technology Teacher Education at Purdue University. He is a postdoctoral researcher in Science Education at Indiana University. He can be reached at* sunge@iu.edu.

This is a refereed article.

Ad Index

Technology and Engineering Teacher Statement of Ownership



UNITED STATES POSTAL SERVICE® Statement of Ownership, Management, and Circulation (All Periodicals Publications Except Requester Publications)

1. Publication Title: Technology and Engineering Teacher
2. Publication Number: 837-480
3. Filing Date: 8/21/18
4. Issue Frequency: Sept, Oct, Nov, Dec/Jan, Feb, March, Apr, May/June
5. Number of Issues Published Annually: 8
6. Annual Subscription Price: $90 / Domestic; $110 / Foreign
7. Complete Mailing Address of Known Office of Publication (Not printer) (Street, city, county, state, and ZIP+4®): ITEEA, 1914 Association Dr., STE 201, Reston, Fairfax, VA 20191-1539
Contact Person: Kathleen De La Paz
Telephone (Include area code): 703-860-2100
8. Complete Mailing Address of Headquarters or General Business Office of Publisher (Not printer): ITEEA, 1914 Association Dr., STE 201, Reston, Fairfax, VA 20191-1539
9. Full Names and Complete Mailing Addresses of Publisher, Editor, and Managing Editor (Do not leave blank)
Publisher (Name and complete mailing address): Steven A. Barbato, ITEEA, 1914 Association Dr., STE 201, Reston, VA 20191-1539
Editor (Name and complete mailing address): Kathleen B. de la Paz, ITEEA, 1914 Association Dr., STE 201, Reston, VA 20191-1539
Managing Editor (Name and complete mailing address): Kathleen B. de la Paz, ITEEA, 1914 Association Dr., STE 201, Reston, VA 20191-1539
10. Owner (Do not leave blank. If the publication is owned by a corporation, give the name and address of the corporation immediately followed by the names and addresses of all stockholders owning or holding 1 percent or more of the total amount of stock. If not owned by a corporation, give the names and addresses of the individual owners. If owned by a partnership or other unincorporated firm, give its name and address as well as those of each individual owner. If the publication is published by a nonprofit organization, give its name and address.)

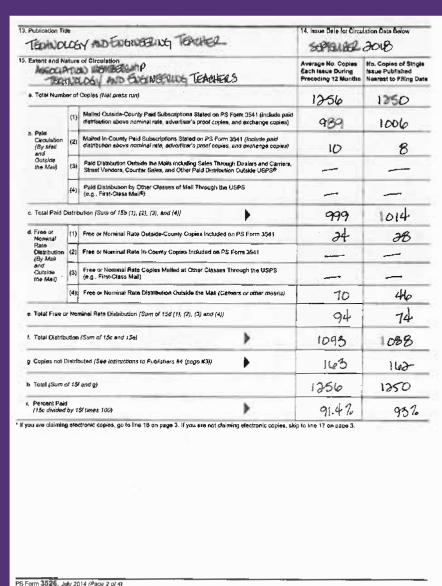
Full Name	Complete Mailing Address
International Technology and Engineering Educators Association	1914 Association Drive, Suite 201, Reston, VA 20191-1539

11. Known Bondholders, Mortgagees, and Other Security Holders Owning or Holding 1 Percent or More of Total Amount of Bonds, Mortgages, or Other Securities. If none, check box ☒ None

Full Name	Complete Mailing Address

12. Tax Status (For completion by nonprofit organizations authorized to mail at nonprofit rates) (Check one)
The purpose, function, and nonprofit status of this organization and the exempt status for federal income tax purposes:
☒ Has Not Changed During Preceding 12 Months
☐ Has Changed During Preceding 12 Months (Publisher must submit explanation of change with this statement)

PS Form 3526, July 2014 (Page 1 of 4 (see instructions page 4)) PSN 7530-01-000-9931 PRIVACY NOTICE: See our privacy policy on www.usps.com

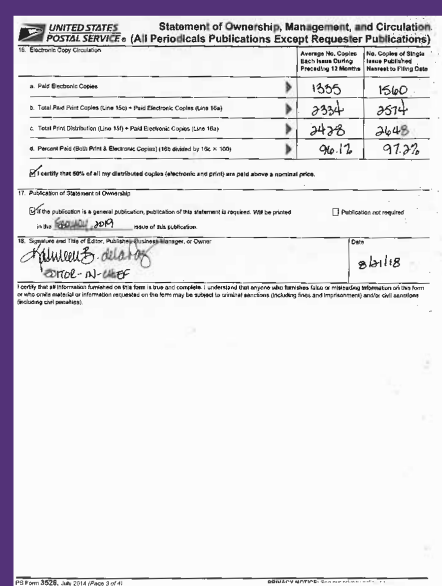


13. Publication Title: Technology and Engineering Teacher
14. Issue Date for Circulation Data Below: September 2018

15. Extent and Nature of Circulation: Association Membership, Technology and Engineering Teachers	Average No. Copies Each Issue During Preceding 12 Months	No. Copies of Single Issue Published Nearest to Filing Date
a. Total Number of Copies (Net press run)	1256	1250
b. Paid Circulation (By Mail and Outside the Mail) (1) Mailed Outside-County Paid Subscriptions Stated on PS Form 3541 (Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies)	989	1006
(2) Mailed In-County Paid Subscriptions Stated on PS Form 3541 (Include paid distribution above nominal rate, advertiser's proof copies, and exchange copies)	10	8
(3) Paid Distribution Outside the Mails Including Sales Through Dealers and Carriers, Street Vendors, Counter Sales, and Other Paid Distribution Outside USPS®	—	—
(4) Paid Distribution by Other Classes of Mail Through the USPS (e.g., First-Class Mail®)	—	—
c. Total Paid Distribution (Sum of 15b (1), (2), (3), and (4))	999	1014
d. Free or Nominal Rate Distribution (By Mail and Outside the Mail) (1) Free or Nominal Rate Outside-County Copies Included on PS Form 3541	24	28
(2) Free or Nominal Rate In-County Copies Included on PS Form 3541	—	—
(3) Free or Nominal Rate Copies Mailed at Other Classes Through the USPS (e.g., First-Class Mail)	—	—
(4) Free or Nominal Rate Distribution Outside the Mail (Carriers or other means)	70	46
e. Total Free or Nominal Rate Distribution (Sum of 15d (1), (2), (3) and (4))	94	74
f. Total Distribution (Sum of 15c and 15e)	1093	1088
g. Copies not Distributed (See Instructions to Publishers #4 (page #3))	163	162
h. Total (Sum of 15f and g)	1256	1250
i. Percent Paid (15c divided by 15f times 100)	91.4%	93%

* If you are claiming electronic copies, go to line 16 on page 3. If you are not claiming electronic copies, skip to line 17 on page 3.

PS Form 3526, July 2014 (Page 2 of 4)



UNITED STATES POSTAL SERVICE® Statement of Ownership, Management, and Circulation (All Periodicals Publications Except Requester Publications)

16. Electronic Copy Circulation	Average No. Copies Each Issue During Preceding 12 Months	No. Copies of Single Issue Published Nearest to Filing Date
a. Paid Electronic Copies	1335	1560
b. Total Paid Print Copies (Line 15c) + Paid Electronic Copies (Line 16a)	2334	2574
c. Total Print Distribution (Line 15f) + Paid Electronic Copies (Line 16a)	2428	2648
d. Percent Paid (Both Print & Electronic Copies) (16b divided by 16c × 100)	96.1%	97.2%

☑ I certify that 50% of all my distributed copies (electronic and print) are paid above a nominal price.

17. Publication of Statement of Ownership
☑ If the publication is a general publication, publication of this statement is required. Will be printed in the February 2019 issue of this publication.
☐ Publication not required

18. Signature and Title of Editor, Publisher, Business Manager, or Owner: Kathleen B. de la Paz, Editor-in-Chief
Date: 8/21/18

I certify that all information furnished on this form is true and complete. I understand that anyone who furnishes false or misleading information on this form or who omits material or information requested on the form may be subject to criminal sanctions (including fines and imprisonment) and/or civil sanctions (including civil penalties).

PS Form 3526, July 2014 (Page 3 of 4)