This is the Preprint Version.

The following publication H. C. B. Chan, "CARES Model for Computing Education," 2021 IEEE International Conference on Engineering, Technology & Education (TALE), 2021, pp. 01-05 is available at https://dx.doi.org/10.1109/TALE52509.2021.9678750.

CARES Model for Computing Education

Henry C. B. Chan
Department of Computing
The Hong Kong Polytechnic University
Hong Kong
cshchan@comp.polyu.edu.hk

***Abstract*—In this paper, we present a CARES model for computing education. It consists of three layers: core layer ("think"), course layer ("learn") and career/development layer ("develop"). The core layer is concerned with two fundamental thinking skills: computational thinking and creative thinking. The course layer is about the curriculum. The career/development layer is for development in four major areas: Application (applying knowledge), Research (advancing knowledge), Entrepreneurship (innovating/realizing knowledge) and Service (serving with knowledge). A freshman seminar subject has been designed based on the application, research and entrepreneurship elements of the CARES model, and the teaching experience for this subject is presented.**

Keywords—computing education, freshman seminar, student development

I. Introduction

As discussed in [1], with the advent of both technologies and pedagogies, there are fundamental changes in university education, as highlighted in Table I. In general, future university education will likely focus on student learning and development using blended learning and with more emphasis on high-level thinking capabilities.

TABLE I. CHANGES IN UNIVERSITY EDUCATION [1]

Past	Present and Future
Classroom	Learning space/communities
Physical	Hybrid/blended
Teacher-centered	Student-centered
Teach	Learn and develop
Local	Global
Knowledge	Thinking and learning capability
Textbooks	Multimedia resources and learning objects (Open Educational Resources (OERs))
Standard grading (e.g., exam)	Holistic assessment and learning analytics
Understand, Apply	Analyze, Design, Create

In this paper, our focus is on computing education. Computers play an important role in our society, so computing education is becoming increasingly important. On the other hand, the field of computing is changing very rapidly. Consider this: more than 10 years ago, we did not have smartphones (i.e., mini computers inside our pockets) and more than 20 years ago, we did not even have the World Wide Web. Looking forward, we do not know what will happen 10 and 20 years from now. We may be teaching students for an unknown future world or career, and what students learn today may become obsolete tomorrow. So, there are two interesting questions: "What and how should we teach our students (i.e., providing computing education in this dynamic environment)?" and "How should we prepare our students for their future career?"

As discussed in [2], *computing curriculum* (i.e., for computing education) has a long history, dating back to the 1960s. The first formal computing curriculum recommendation(s) (Curriculum'68) was published by ACM in 1968 [3]. Subsequently, other proposals have been published through the joint efforts of ACM and IEEE. In [2], it was also suggested to include the history of computing as part of the curriculum, as it would allow students to learn about the work of key people and professionals to learn their lessons (e.g., past mistakes). In the 2010s, CS2013 was the representative computer science curricular with the following key design principles [3]: providing required computing skills, defining the body of knowledge, supporting various computer science programs (e.g., by both large and small schools and covering professional practice (e.g., ethics and communication skills). More recently, the CC2020 project defines the latest computing curricular involving 36 professionals in 16 countries [4][5]. It is based on an IT competency model with three major components: knowledge (e.g., concepts); skills (e.g., capabilities and hands-on experience) and disposition (e.g., social skills and attitudes). Apart from the curricular, there are other development areas for student learning. For example, there has been growing interest in undergraduate research. In general, it enhances various student skills, such as formulating problems, analyzing data, interpreting results and enhancing self-confidence [6]. Another important development area is entrepreneurship, which involves three important elements: discovery, innovation and creativity [7]. Indeed, they are the core elements for the development of computing as well (e.g., see the computing history). In [8], two entrepreneurship models with both breadth and depth components were proposed to integrate entrepreneurship into the computing curriculum. Last but not least, service-learning is also becoming an important area to enhance student learning from another perspective. For example, as illustrated in [9], service-learning can be used to enhance student design and problem-solving skills.

Contributing to the development of computer science education and complementing the related work, this paper proposes a generic three-layer model called CARES (Computing for Application, Research, Entrepreneurship and Service). The focus is on four development areas, which require different skill sets. The remaining sections of this paper are organized as follows. Section II discusses the CARES model. Section III presents a related freshman seminar subject. Section IV gives the conclusion.

II. CARES MODEL

In this section, we present the CARES model. "Care" is the foundation of education (i.e., care for students) and the CARES model provides a framework for supporting computing education with the following *CARES* focus: *C*omputing for *A*pplication, *R*esearch, *E*ntrepreneurship, and *S*ervice. The model comprises three layers: core layer, course layer, and career/development layer. At the core layer, there are two basic thinking capabilities: computational thinking and creative thinking. The course layer is essentially the curriculum. There are four pillars in the career/development layer namely: Application, Research, Entrepreneurship, and Service. More explanations will be provided below.

The core layer seeks to address the fundamental thinking capabilities of students. Albert Einstein once said: "*Logic will get you from A to B. Imagination will take you everywhere*." Traditionally, computing education focuses on computational or logical thinking. In addition, it is now important to enhance students' creative thinking capability (e.g., for developing new computing applications). In summary, the CARES model aims to foster the development of both computational thinking and creative/innovative thinking (i.e., for critical thinking and problem solving).

The course layer addresses the curriculum. It covers knowledge and skills. A good reference, of course, is based on the ACM/IEEE Computer Science Curricula. Note that a curriculum needs to be revised regularly to address changes in the field of computing.

The career/development layer helps students to develop in a certain area. In general, after graduation, students may develop their career in one of the following main areas:

- Application – to apply computing in a certain area (e.g., to work as a programmer in a bank)
- Research – to continue graduate studies and engage in research work (e.g., to study for a PhD)
- Entrepreneurship – to launch a business and/or to join a startup company to develop a new business (e.g., to be a founder of a startup)
- Service – to serve society using computing (e.g., to serve in a charitable organization)

Each area has fundamental differences and requires different skills. It is important for students to understand these differences and skills so they can explore the best area for developing their career (see Table II). There may even be a major area and a minor area for development. In general, each area involves problem-solving with different focuses. With regard to the application area, the problem is relatively well-defined, and in many cases user requirements should be collected and analyzed. The solution usually involves choosing and using existing technologies. To enhance student development in the application area, work-integrated education cases and internships are some of the best teaching/learning methodologies. The area of research involves a more rigorous problem-solving process. The problem may not be well-defined, and in fact there may be new problems. To solve problems, new and/or more advanced technologies are often used. In recent years, there have been many undergraduate research experience programs at different universities. They provide extracurricular activities to enhance students' learning in research. The area of entrepreneurship involves exploring a new opportunity and managing available resources to accomplish the mission and to solve problems. Setting up and running virtual companies with students is an effective way to enhance students' entrepreneurship learning. Last, but not least, a different philosophy is emphasized in the area of service: "to give rather than to take." University education should not only be for the "brain," but also for the "heart." It is important for students to realize that it is important to "serve." In recent years, service learning has attracted considerable interest, and it provides an effective way to fulfill the goal of service education.

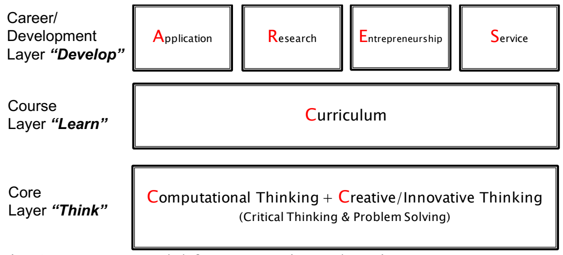

Fig. 1: CARES Model for Computing Education
(Computing for Application, Research, Entrepreneurship and Service)

TABLE II: CAREER/DEVELOPMENT LAYER – APPLICATION, RESEARCH, ENTREPRENEURSHIP AND SERVICE

	Application	Research	Entrepreneurship	Service
Focus	Applying knowledge	Advancing knowledge	Innovating and realizing knowledge	Serving with knowledge
Example	Develop an application to solve a problem	Design a new algorithm	Introduce an innovative service to satisfy a market need	Provide a service without reciprocity

III. FRESHMAN SEMINAR SUBJECT AND TEACHING EXPERIENCE

In this section, we present the design of a freshman seminar subject based on the application, research and entrepreneurship elements of the CARES model. The objectives of the subject are to:

- *educate and inspire students about different aspects of Information Technology and their applications;*
- *cultivate students' global outlook through the local and international social impact of Information Technology;*
- *cultivate and develop students' creative thinking, computational problem-solving and logical reasoning skills;*
- *educate students on different aspects of entrepreneurship and the process of creating new ventures in the information technology industry; and*
- *engage the students in desirable forms of learning in university, including self-regulation, autonomous learning and deep understanding.*

Based on the application, research and entrepreneurship components of the CARES model, students need to conduct three computing-related projects on application, research and entrepreneurship. The aim is to allow students to interact with the development layer through different types of projects (i.e., applying knowledge through an application project, advancing knowledge through a research project and innovating/realizing knowledge through an entrepreneurship project). Fundamentally, the projects seek to develop student computational thinking and creative/innovative thinking skills (i.e., related to the core layer). Project-based learning is used so that students can "learn by doing". The projects are designed to facilitate students to choose their later programs as well (i.e., for the course layer). There are two technical projects - one system/application-oriented project for the application component and one data-oriented project for the research component. In addition, there is one entrepreneurship project (i.e., for the entrepreneurship component), which is related to the application project. For the application project, students need to define user requirements and apply computing knowledge to satisfy the requirements. For the research project, students need to conduct research to study a stock price prediction method based on machine learning. For the entrepreneurship project, students are encouraged to learn from another perspective by forming a virtual startup and presenting a business idea/proposal. The application, research and entrepreneurship modules are further discussed in the following sub-sections.

<u>Application</u>

For teaching "Application", the aim is for students to learn how to define user requirements and develop an application to fulfill the requirements. First, a brief overview of software engineering was introduced, such as the basic steps in the Waterfall model. Furthermore, an example of a use case diagram was presented (e.g., see Fig. 1) based on a learning management system. Using a similar approach, students were encouraged to define the user requirements and functions of a mobile app for their application project.

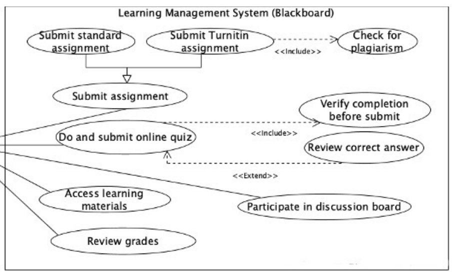

Fig. 1. User case diagram example

For the application projects, App Inventor [10] was used as the development tool. The advantage of App Inventor is that it is a user-friendly development tool based on visual programming for freshmen. It is well suited for teaching computational thinking (i.e., focusing on logic instead of syntax) for students/freshmen without a computing background. Before developing a mobile app, various programming logic topics (i.e., related to computational thinking) were presented. Fig. 2 shows an example of introducing a "while" loop with an exercise. Students were asked to provide the output of each iteration.

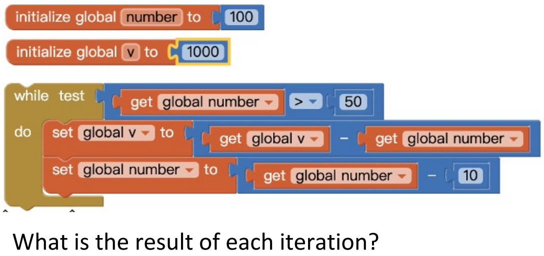

Fig. 2. Looping example

To facilitate students to develop their own apps, two basic apps were introduced with two different focuses. These apps are inspired by the tutorials in [10]. The first one (E-card app) focuses on user interfaces, which is inspired by the PaintPot tutorial in [10]. Fig. 3 shows an overview of the mobile app. Basically, it allows users to take a picture, draw on the picture, add text and share with others. The programming blocks are component-based, clearly showing the programming logic. For example, Fig. 4 shows how to take a picture and set the picture as the background image. This component-based approach greatly facilitates freshmen without any programming background to understand the basic concept and operation.

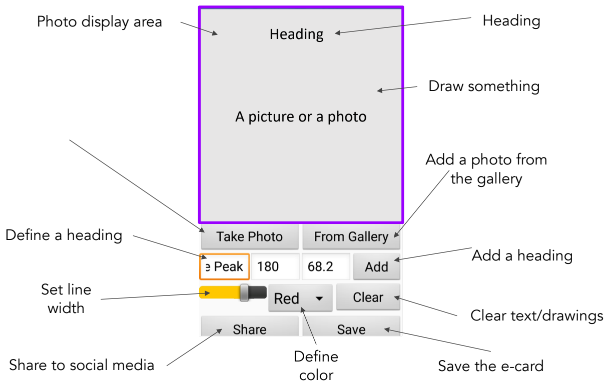

Fig. 3. Overview of the E-card app

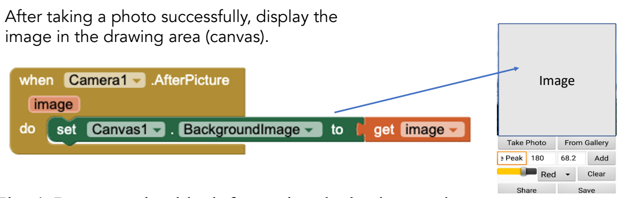

Fig. 4. Programming block for setting the background

The second app (a game) seeks to teach students more logical thinking, a different focus. Fig. 5 shows the game rules. Basically, a player needs to move the blue balls through the obstacles to meet the red ball. There are two major modules: detecting collision and controlling the ball movement. Fig. 6 shows the collision detection module, demonstrating the branching logic. Fig. 7 shows how the blue ball can be moved or controlled. Note that these two modules also show some basic object-oriented-like concepts for freshmen.

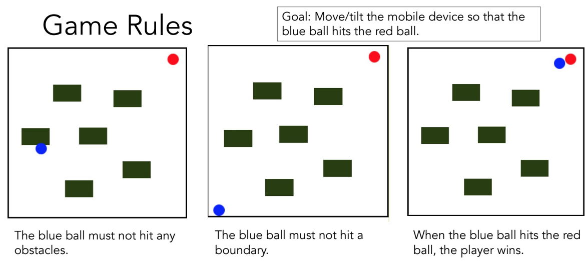

Fig. 5. Game rules

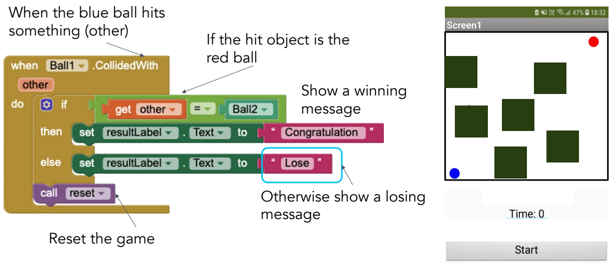

Fig. 6. Collision detection

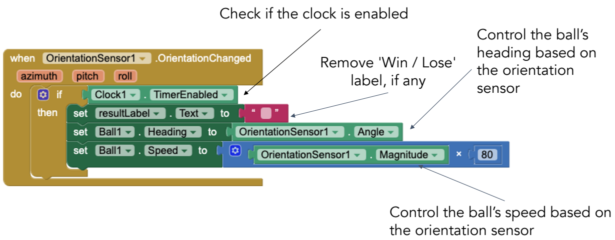

Fig. 7. Controlling ball movement

After learning App Inventor (e.g., through the aforementioned apps), each student group needed to conduct an application project (i.e., related to the development layer of the CARES model) by developing a mobile app. To support the application element of the development layer, there were two main parts in the application project: defining the user requirements through a use case diagram, and implementing the mobile app using App Inventor. The aforementioned apps provide the foundation, and students are encouraged to learn further through project-based learning.

Research

For teaching "Research", the aim is for students to learn how to define a problem, study related work, and investigate and evaluate a method. First, a brief overview of research methodology was introduced:

- Define objectives and formulate the problem statement – What is the research goal?
- Study related work (do a literature review) – What did others do?
- Formulate the research methodology – What methods can solve the problem?
- Prepare a proposal – What is the plan to solve the the problem?
- Conduct research – How to carry out research?
- Perform evaluation – Are the research results correct/reliable?
- Prepare reports/papers - What are the results/findings?

For studying and comparing related work, the following table-based approach (see Fig. 8) was introduced so that students could compare various attributes or related work.

Related work

Comparison

	Work 1	Work 2	Work 3
Attribute 1	Yes	No	N.A.
Attribute 2	High	Medium	Low
Attribute 3	X%	Y%	Z%
Attribute 4	Comment 1	Comment 2	Comment 3

Fig. 8. Comparing related work

As an example of project-based learning, a research project on stock price prediction was used. The objective is to investigate a few machine learning-based methods for predicting stock prices. First, the basic concept of machine learning was introduced to students. To facilitate the explanation, a result prediction example based on quiz result, test result etc. was used (see Fig. 9). This study-related example facilitates students to learn the basic concept. Students were encouraged to apply a similar approach for stock price prediction (e.g., by learning features such as the past five days of prices to predict the next day's price).

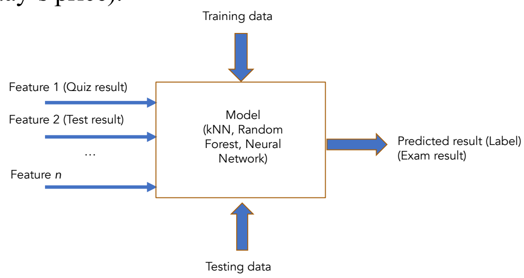

Fig. 9. Basic concept of machine learning

For the machine learning tool, Orange [11] was introduced, which is again a visual programming tool suitable for freshmen, as many freshmen might not have a programming background. The aim is to introduce machine learning for research purposes, focusing on the concept rather than on the programming syntax. Fig. 10 shows an example of the machine learning model. Students could import a spreadsheet file (i.e., stock data from, for example, Yahoo Finance, and define different columns for features training). Various machine learning models can be used for stock price prediction purposes. The predicted results could be saved in a file for further analysis. In summary, to support the research element of the development layer, the research project encouraged students to compare different models and methods for stock price prediction. The evaluation was based on mean absolute error (i.e., between the real price and predicted price).

Entrepreneurship

For teaching "Entrepreneurship", the aim is for students to learn how to develop a basic business model in particular, which can be linked with the application project. To facilitate the teaching, inspired by the Business Model Canvas [12], a concise C^3 Model-Map was used [13].

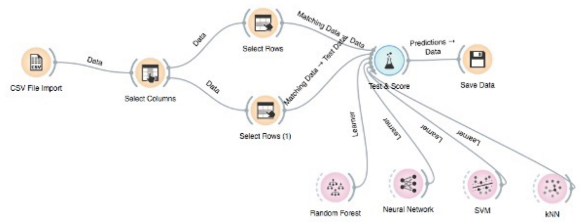

Fig. 10. Machine learning model using Orange [11]

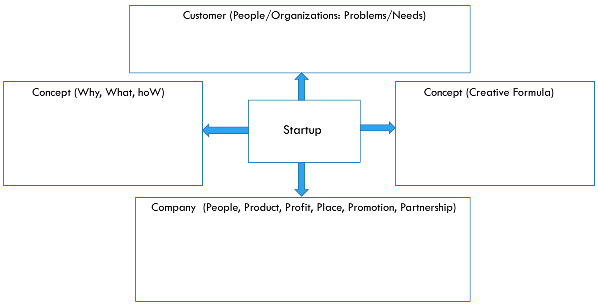

Fig. 11. C^3 Model-Map

As shown in Fig. 11, a C^3 Model-Map comprises three basic components: Customer, Concept and Company. In essence, these components seek to answer three basic questions [13]:

- Who are your customers and what are their needs/problems?
- What is the business concept to fulfill the customer's needs or solve the customer's problems?
- How to set up a company to implement the business concept?

Basically, students were encouraged to use a customer-oriented approach. First, they should identify the customer's needs/problems. Second, they should develop a business concept (using both analytical thinking and creative thinking) to solve the customer's problems or satisfy the customer's needs. Finally, they should design a company to implement the business concept using the 6P elements, an enhancement/extension of the traditional 4P marketing mix. In summary, to support the entrepreneurship element of the development layer, each student group needed to present a business proposal based on the C^3 Model-Map and other information.

IV. Conclusion

In conclusion, this paper has presented a CARES model for computing education with three layers and four major development elements: Application, Research, Entrepreneurship and Service. In essence, these elements seek to cover four areas: applying knowledge, advancing knowledge, innovating knowledge and serving with knowledge. A freshman seminar subject based on the Application, Research and Entrepreneurship components of the CARES model has also been presented. The subject is based on project-based learning – a mobile app project for learning application, a machine learning project for learning research and a business project for learning entrepreneurship. Fundamentally, the projects aim to develop students' computational thinking and creative thinking skills (i.e., the core layer of the CARES model).

Acknowledgments

This work was supported by the Department of Computing, The Hong Kong Polytechnic University. We thank Ho Yik Him for developing the E-card and Orientation Game apps, and Lam Wa Lap for developing some teaching materials.

References

[1] H. C. B. Chan and T. T. Fung, "Enhancing student learning through mobile learning groups", *Proc. 2020 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pp. 99-105, 2020.

[2] J. Impagliazzo, M. Campbell-Kelly, G. Davies and J. A. N. Lee, "History in the computing curriculum", *IEEE Annals of the History of Computing*, vol. 21, no. 1, pp. 4-16, Jan.-March 1999.

[3] S. Roach, M. Sahami and R. LeBlanc, "Special session: the CS2013 computer science curriculum guidelines project", *Proc. 2012 Frontiers in Education Conference*, pp. 1-2, 2012.

[4] J. Impagliazzo, A. Clear and H. Alrumaih, "Developing an overview of computing/engineering curricula via the CC2020 project", *Proc. 2018 IEEE World Engineering Education Conference (EDUNINE)*, pp. 1-4, 2018.

[5] J. Impagliazzo and A. N. Pears, "The CC2020 project — computing curricula guidelines for the 2020s", *Proc. 2018 IEEE Global Engineering Education Conference (EDUCON)*, pp. 2021-2024, 2018.

[6] V. P. Mishra, "A model to improve undergraduate research: the need of future", *Proc. 2018 Advances in Science and Engineering Technology International Conferences (ASET)*, pp. 1-2, 2018.

[7] A. Q. Gates, "Discovery, innovation, and creativity: the core of computing", *Computer*, vol. 43, no. 2, pp. 98-100, Feb. 2010.

[8] S. Doboli, G. L. Kamberova, J. Impagliazzo, X. Fu and E. H. Currie, "A model of entrepreneurship education for computer science and computer engineering students", *Proc. 2010 IEEE Frontiers in Education Conference (FIE)*, pp. T4D-1-T4D-6, 2010.

[9] L. Zhao and K. Vernaza, "Teaching first year engineering students engineering design process and problem solving through service learning projects", *Proc. 2016 IEEE Frontiers in Education Conference (FIE)*, pp. 1-7, 2016.

[10] https://appinventor.mit.edu

[11] https://orangedatamining.com

[12] https://www.strategyzer.com/books/business-model-generation

[13] H. C. B. Chan, "Teaching entrepreneurship using C^3 model-map", *Proc. 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, pp. 1114-1118, 2021.