The College of New Jersey
New Minor Approval or Change in Minor

Name of Minor: Computer Science Education Minor

Term Effective Date: Fall 2019

Home School: Science

Home Department: Computer Science

Type of Approval

X New Minor within a Department

___ New Interdisciplinary Minor

___Modification of Existing Minor

___Deactivation of Existing Minor (no replacement)

Briefly describe the minor and its requirements. For interdisciplinary minors, list other departments and schools with courses included in the minor.

Due to the increasing demands in the markets and continuous efforts from government as well as industry, the trend for computer science education in high school is clear and TCNJ is definitely willing to be one of the leaders in training qualified teachers for K-12 schools. In this minor, students will have intensive practices of computational thinking skills, will gain necessary programming skills and will also learn how to teach computer science courses in a high school environment. By taking the upper level optional course, the students will broaden their versions and keep themselves up-to-date. The CS Education Minor is open to all Education majors at TCNJ, including but not limited to: Math, Science, Technology, Humanities and Social Studies, and Special Education, at the elementary or secondary level. A candidate for the endorsement would have to hold a standard instructional certificate with at least one other endorsement. Curriculum includes three possible pathways: (1) CSC 102/250/270/305/300-400 Option; (2) CSC 215/250/270/305/300-400 Option; (3) CSC 220/230/270/305/300-400 Option. (See attached proposal for more information.)

Initial Approval

In the case of new minors, the signatures indicate that all affected units within a school have reviewed the attached proposal, and that the attached proposal includes a detailed outline of the curriculum and needed resources such as: a. faculty and support of instruction for all units involved; b. library resources; c. equipment, laboratory support, and computer support; and d. facilities for all units involved. If the minor does not involve courses outside the home department, the proposal is submitted to the Steering Committee after this step.

Department Chair — Date: 12/19/18

School Curriculum Committee — Date: 12/19/18

Dean — Date: 12/17/18

Additional Approvals for Interdisciplinary Minors

In the case of interdisciplinary minors, the new minor proposal must also include signatures from the department chairs of all affected units, indicating their review. On this form, include approvals from curriculum committees and deans of all involved schools. The role of the school curriculum committee(s) is to ensure that all procedures have been followed in the approval process, including review by all affected departments, and that the proposed minor is consistent with the mission of the College and can be reasonably supported with resources.

The minor proposal and new course were reviewed by faculty in the School of Education and Integrative STEM Education in School of Engineering. Since it is not an interdisplinary minor, we did not obtain signatures.

Curriculum Committee	Date
Dean	Date
Curriculum Committee	Date
Dean	Date
Curriculum Committee	Date
Dean	Date

College Governance

If recommended by the school committee(s), the proposal is submitted to the Steering Committee to be forwarded to the Committee on Academic Programs (CAP) for its review and recommendation.

Steering (indicating review by CAP)	Date

Final Approval

Provost	Date

Proposal: Minor in Computer Science Education

October 11, 2018

Rationale for minor:

In the former President Obama's "A Strategy for American Innovation", education and scientific research have been identified as two basic building blocks of American Innovation. To implement the strategy, Computer Science for All was presented as a bold initiative to empower all American students from kindergarten through high school to learn computer science and be equipped with the computational thinking skills. they need to be creators in the digital economy, not just consumers, and to be active citizens in our technology-driven world.

In the market, there were more than 500,000 high-paying tech jobs that were unfilled, and by the end of 2018, 51 percent of all STEM jobs are projected to be in computer science-related fields (according to the data from Code.org). Computer science and data science are not only important for the tech sector, but also for many other industries, including transportation, healthcare, education, and financial services.

Across high schools in the United States, increasing number of students have taken college computer science courses and the number of completed AP computer science exams increased from 46,344 in year 2015 to 111,262 in year 2017. However, by the estimates, just one quarter of all the K-12 schools in the United States offer high-quality computer science with programming and coding. The schools report a lack of qualified teachers as a key barrier to offering computer science courses.

Both government agencies and industrial companies, from different perspectives, are investing in the field intensively to bring computer science education to high schools. National Science Foundation's CS for ALL program aims to provide all U.S. students the opportunity to participate in computer science and computational thinking education in their schools at the preK-12 levels. Google's educator grants (formerly known as CS4HS) try to deliver professional development to educators in their local communities throughout the school year. Other New Jersey institutions, such as Kean University and Stockton University, are on their way to offer programs for computer science educators.

Due to the increasing demands in the markets and continuous efforts from government as well as industry, the trend for computer science education in high school is clear and TCNJ is definitely willing to be one of the leaders in training qualified teachers for K-12 schools. In this minor, students will have intensive practices of computational thinking skills, will gain necessary programming skills and will also learn how to teach computer science courses in a high school environment. By taking the upper

level optional course, the students will broaden their versions and keep themselves up-to-date.

It should be noted that, in January 2018, former Governor Christie signed a bill to the law that require all high schools in NJ to offer Computer Science classes. This made the state the 5th state of the nation to have such a requirement. Then on August 10, 2018, Governor Murphy signed the Computer Science teaching endorsement bill. Once NJDOE develops the specific regulations, colleges can begin offering a 15-credit add-on CS endorsement for 9-12 teachers who hold a certification in another subject area. The law has a grandfather clause for teachers who are already teaching CS. The endorsement will not be a requirement until NJDOE determines that there are enough CS teachers. All these recent changes in legislature give both opportunity and a sense of urgency to the College and the Computer Science Department to keep their leadership positions in the state's higher education system.

To this end, the CS for All Subcommittee of the Computer Science Department proposed the development of Minor in Computer Science Education program to the departmental curriculum committee on June 6, 2017, and the proposal was approved by the committee on June 12, 2017. The departmental curriculum committee approved the developed Minor in Computer Science Education program proposal on August 29, 2018. We are now requesting approval of the program from the School of Science Curriculum Committee.

Target:

The CS Education Minor is open to **all** Education majors at TCNJ, including but not limited to: Math, Science, Technology, Humanities and Social Studies, and Special Education, at the elementary or secondary level. A candidate for the endorsement would have to hold a standard instructional certificate with at least one other endorsement.

Curriculum:

The CS Education Minor requires five courses in Computer Science consisting of four core courses and one elective course. Students can choose one of the following three pathways to meet the minor program requirements:

(A)	(B)	(C)
1. CSC 102	1. CSC 215	1. CSC 220
2. CSC 250	2. CSC 250	2. CSC 230
3. CSC 270	3. CSC 270	3. CSC270
4. CSC 305 Methods	4. CSC 305 Methods	4. CSC 305 Methods
5. CSC Option	5. CSC Option	5. CSC Option

Students must obtain a grade of C or higher in each of the first three courses to take the CSC option course. Education Majors who want to minor in Computer Science Education must first complete the pedagogy course of their chosen major before taking the CSC 305 Methods in Teaching Computer Science (MTCS) course. Students who took and obtained a score of 4 or higher in AP CS Principles may be waived from the CSC 102 requirement. Similarly, students who took and obtained a score of 4 or higher in AP CS A may be waived from the CSC 220 requirement. In either case, the student needs to take an additional option course to meet the five-course requirement for the minor. For current TCNJ math majors, the CSC 270 requirement may be fulfilled with CSC 271. For the option courses, students can choose from the list of courses below.

List of CSC Option Courses:

1. CSC 307: Data Mining and Predictive Modeling
2. CSC 315: Database Systems
3. CSC 320: Information Retrieval
4. CSC 335: Analysis of Algorithms
5. CSC 350: Computer Graphics
6. CSC 355: Human Computer Interaction
7. CSC 360: Computer Networking
8. CSC 380: Artificial Intelligence
9. CSC 425: Compilers and Interpreters
10. CSC 435: Programming Languages
11. CSC 445: Theory of Computation
12. CSC 448: Algorithms in Computational Biology
13. CSC 450: Computer and Network Security
14. CSC 470: Topics in Computer Science
15. CSC 471: Genomics and Bioinformatics

Core Course Overview:

CSC 102: Introduction to Computational Thinking

This course introduces students to computational thinking and the foundational concepts of computer science and challenges them to explore how computing and technology can impact the world. Students will focus on creative problem solving and real-world applications. They will analyze problems, evaluate proposed solutions, and

create computational artifacts while learning about algorithms, data and information, and abstraction. Issues related to the Internet and the global impact of computing will also be discussed.

CSC 250: Accelerated CS I, II

A first, intensive course in computer science for non-majors with demonstrated programming experience. The basic introduction to programming (as covered in CSC 220) is considered a review, while the emphasis on problem solving and solution design is presented within the context of a thorough grounding in the classic data structures using the modern object-oriented framework (as covered in CSC 230).

CSC 270: Discrete Structures

This course introduces students to concepts and structures fundamental to computer science, while emphasizing algorithmic thinking, mathematical reasoning, and their applications. Topics include logic, proofs, sets, functions, relations, graphs, induction, recursion, languages and grammars, and finite-state machines.

For current TCNJ math majors, the CSC 270 requirement may be fulfilled with CSC 271. The course is an addition to MAT 200: Proof Writing Through Discrete Mathematics, and is meant to explore discrete mathematics concepts and structures that are fundamental to computer science, but are not covered in that class. Topics in CSC 271 will include applications of number theory, recursion, graphs, trees, languages and grammars, and finite state machines. MAT 200 should cover topics including sets, graphs, Euler and Hamilton circuits, connectivity, planar graphs, recursion and difference equations. Descriptions of the proof types such as direct, indirect, cases, contrapositive and induction should be worked into the material whenever possible. Some optional topics should be included such as fractals, algorithms, combinatorial proofs, discrete codes, logic, Platonic solids and combinatorial counting.

CSC 305: Methods in Teaching Computer Science

The student will become familiar with national and state standards relating to computer science instruction, know how to teach according to these standards, and become familiar with standards-based curricula. The student will understand learning theories as applied to the teaching and learning of computer science: learning in groups,

learning by inquiry, planning constructivist activities, creative and non-conventional use of the computer laboratory, use of metaphors, multimedia, and games. Students will also understand computer science problem-solving strategies.

Cover Sheet for Connecting
Methods of Teaching Computer Science
to
Computer Science Program

This cover sheet briefly summarizes the relationship between the proposed Methods of Teaching Computer Science (MTCS) course and the Computer Science (CS) program's learning goals. The cover sheet functions as an executive summary that provides the rationale for the course and information on the place of the course in the program's curriculum.

I. Learning goals

The proposed MTCS course will provide essential foundations to become effective teaching professionals in computing at K-12 schools. Thus, it meets the following CS program educational objectives: upon graduation students will be able to (1) become valued members in the computing profession and (2) contribute to the well-being of the community and the world at large. The student will understand learning theories as applied to the teaching, learning and assessing of CS through a range of approaches including but not limited to: learning in groups, learning by inquiry, planning constructivist activities, creative and non-conventional use of the computer laboratory, use of metaphors, multimedia and games.

II. Student assessment

Students will be assessed using variety of aspects of the content matter, including course and assessment design, research, case study analysis, practicum observation, and teaching demonstration. All student learning outcome targets closely map to the CS program outcome. For example, upon successful completion of the course, students must be able to demonstrate understanding of national and state standards. This will map to CS program objective for the graduates: an ability to recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles. Similar mapping can be applied to corresponding criteria.

III. Learning activities

Learning activities in the proposed MTCS course will consist of a combination of lectures, demonstrations, explorations, group work, participation in class discussions, reading, programming assignments, writing in class, writing lesson plans and a practicum of approximately 10 hours. Outside of class, the student is expected to do a significant amount of individual or group homework to achieve the learning goals. These will all contribute to accomplishing the CS program student outcome targets, including: (a) An ability to design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline, (b) an ability to communicate effectively in a variety of professional contexts, and (c) an ability to function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.

THE COLLEGE OF NEW JERSEY
COURSE APPROVAL/CHANGE FORM

☒ New Course
☐ Deactivate course
☐ Modify Course

Does this replace a previous course? ☒ No ☐ Yes, course number
(A previous course may be replaced by **one** new course.)

Course name: Methods of Teaching Computer Science

(30 characters, including spaces, maximum)

Course number (with prefix): CSC 305 number (If NEW, number may be suggested, but final assigned by Scheduling)

Term effective date: Spring 2019

School (select one)
☐ Arts and Communication ☐ Business ☐ Culture and Society ☐ Education
☐ Engineering ☐ Nursing/HES ☒ Science ☐ No school (explain in comment box)

Course level
(if undergraduate) ☐ 100 ☐ 200 ☒ 300 ☐ 400 ☐ 09x
(if graduate) ☐ 500 ☐ 600 ☐ 700

Proposed units:
(if undergraduate) ☒ 1 =4 SH ☐ .75 =3 SH ☐ .5 =2 SH ☐ .25 = 1 SH ☐ variable (explain in comment box)

(if graduate) ☐ 1 credit ☐ 2 credits ☐ 3 credits ☐ 4 credits ☐ 5 credits ☐ 6 credits ☐ variable (explain in comment box)

Is course repeatable for credit? ☐ Yes ☒ No If yes, how many times?

Are there pre-requisites? ☒ Yes ☐ No If yes, specify using "and", "or" to clarify:

CSC 102 (or CSC 215), CSC 230 (or CSC 250), CSC 270 each with a grade of C or higher.

Are there co-requisites? ☐ Yes ☒ No If yes, specify using "and", "or" to clarify:

Grade Type: ☒ Standard letter grade (GR) ☐ Pass/Unsat with credit (PU) ☐ Pass/Unsat no credit (DC) ☐ Other (explain in comment box

Offering schedule: ☐ fall only ☐ spring only ☐ fall and spring ☐ summer only ☐ all terms ☒ occasionally

Activity code for primary activity:
☒ Lecture /discussion (LDI) ☐ Lecture (LEC) ☐ Seminar (SEM) ☐ Lecture/ Studio (LSU)
☐ Interactive Proficiency (INA) ☐ Independent Research (INR) ☐ Independent Study (IND) ☐ Advance Tutorial/ Group Study (GPS)
☐ Group Research (RES) ☐ Large Ensemble (ENL) ☐ Moderate Sized Ensemble (ENS) ☐ Private Music Lessons (PVT)
☐ Internship (INT) ☐ Clinical (CNL) ☐ Practicum (PRA) ☐ Study Abroad Seminar
☐ Study Abroad Independent

Activity code for secondary activity:	☐No secondary activity	☐Laboratory (LAB)	☐Design (DES)	☐Recitation(RCT)
	☐Studio (STU)	☒Practicum (PRA)	☐Conversation Hour (CHR	☐Conference Hour (CON)

Is this course equivalent to any other? ☐ Yes ☒ No If yes, list course number(s) and name(s):

Is this a topics course? ? ☐ Yes ☒ No If yes, what are the anticipated topic names? (This list can be added to as more topics become available.)

Are there any known Liberal Learning course attributes? ☒ Yes ☐ No If yes, please list. (This list can be altered as needed.)

Quantitative Reasoning

Notes:

- Faculty weighted hours are determined by activity and weekly contact hours.
- Weekly contact hours are determined by the activity unless otherwise requested in the comment box below.
- Standard course cap is determined by the activity unless otherwise requested in the comment box below.

Comments:

This course will provide pedagogical knowledge for teaching computer science to K-12 students. Students will learn methods and tools for effective computer science teaching. A demonstration teaching session at the end of the course will be used as one of the key assessment measures of student performance.

Approvals:

Program Faculty — Date 9/13/2018

Appropriate Comm. (Optional) — Date

Dean — Date 12/17/18

Once completed and approved, please forward an electronic copy of this form from the office of the dean to schedule@tcnj.edu AND attach a Word document with a short course description (approximately 50-100 words) as it will appear in the Bulletin.

Revised September, 2011