MINISTRY OF HIGHER EDUCATION, SCIENCE, AND TECHNOLOGY
UNIVERSITAS NEGERI SURABAYA
FACULTY OF MATHEMATICS AND NATURAL SCIENCES
UNDERGRADUATE PROGRAM OF MATHEMATICS EDUCATION
Ketintang Campus, Jalan Ketintang, C8 C9 Building, Surabaya 60231
Phone: +62 895335466373, email: s1-pmat@unesa.ac.id
Website: https://pendidikan-matematika.fmipa.unesa.ac.id/

Module Name:	Computational Thinking Berpikir Komputasional
Module Level:	Sarjana (S-1) / Undergraduate
Abbreviation, if applicable:	8420202239
Sub-heading, if applicable:	-
Course included in the module, if applicable:	-
Semester/term:	8 / Fourth year
Module Coordinator(s):	Dr. Yurizka Melia Sari, M.Pd.
Lecturer(s):	Novita Vindri Harini, M.Pd.
Language:	Indonesian
Classification within the curriculum:	~~Compulsory course~~/elective studies
Teaching format/class hours per week during the semester	Teaching format: lectures, unplugged and hands-on computing activities, problem-based learning, project-based learning, and group discussions Class hours per week: 2 x 170 minutes = 340 minutes (5.67 hours), of which 2 x 50 minutes are lectures
Workload:	16 weeks per semester consisting of: • 2 hours lectures (2 x 50 minutes) per week, • 2 hours structured assignments (2 x 60 minutes) per week, • 2 hours independent study (2 x 60 minutes) per week, Total workload: 16 x 2 x 170 minutes = 5,440 minutes = 90.67 hours = 3.18 ECTS*
Credit Point:	2 CU (3.18 ECTS)
Requirements:	N/A
Learning Goals :	**Programme Learning Outcomes (PLO) addressed:** PLO 3: Develop logical, critical, systematic, and creative thinking in carrying out specific work in their area of expertise and in accordance with the work competency standards of the

	relevant field PLO 5: Apply the basic principles of mathematics to solve simple mathematical problems PLO 9: Demonstrate mathematical knowledge and insight **Course Learning Outcomes (CLO): after completing this module, students are able to:** CLO-1: Explain computational thinking and its components (decomposition, pattern recognition, abstraction, algorithmic thinking, and evaluation) and relate them to mathematical problem solving (PLO 3) CLO-2: Apply decomposition, pattern recognition, and abstraction to solve basic mathematical problems on number patterns, sequences, geometry, and probability, and generalise the patterns found (PLO 5) CLO-3: Design, represent (flowcharts and pseudocode), and verify algorithms for mathematical procedures, and analyse their correctness and efficiency (PLO 9) CLO-4: Implement algorithms and simulations for mathematical problems using spreadsheets, a programming environment, and GeoGebra (PLO 5) CLO-5: Test, debug, and evaluate computational solutions, and develop and present a computational solution to a mathematical problem together with a learning activity that uses computational thinking (PLO 3)
Content:	This course introduces computational thinking as a way of solving mathematical problems. Students learn its core components, namely decomposition, pattern recognition, abstraction, and algorithmic thinking, together with testing, debugging, and evaluation, and apply them to problems on number patterns, sequences, geometry, and probability. Students design and verify algorithms for mathematical procedures, represent them as flowcharts and pseudocode, and implement them with spreadsheets, a programming environment such as Scratch or Python, and GeoGebra. They also use simulations to explore

<table>
<tr><td></td><td>mathematical ideas, and look at how computational thinking can be integrated and assessed in school mathematics, for example through Bebras-style tasks. Learning is problem-based and project-based, ending in a final project in which students develop a computational solution to a mathematical problem and a short learning activity that uses it.</td></tr>
<tr><td>Study/exam achievements</td><td>

Students are considered competent and pass if the final score, calculated from weekly tasks and participation (45%), the midterm computational problem solving portfolio (20%), the final project presentation (5%), and the final examination (30%), is at least 55 or C.
Final score is calculated as follows:

<table>
<tr><th>Week</th><th>Course Learning Outcomes (CLO)</th><th>Programme Learning Outcomes (PLO)</th><th>Evaluation</th></tr>
<tr><td>1</td><td>CLO-1</td><td>PLO-3</td><td>3%</td></tr>
<tr><td>2</td><td>CLO-1</td><td>PLO-3</td><td>3%</td></tr>
<tr><td>3</td><td>CLO-2</td><td>PLO-5</td><td>3%</td></tr>
<tr><td>4</td><td>CLO-2</td><td>PLO-5</td><td>3%</td></tr>
<tr><td>5</td><td>CLO-2</td><td>PLO-5</td><td>3%</td></tr>
<tr><td>6</td><td>CLO-3</td><td>PLO-9</td><td>3%</td></tr>
<tr><td>7</td><td>CLO-3</td><td>PLO-9</td><td>3%</td></tr>
<tr><td>8</td><td>CLO-3</td><td>PLO-9</td><td>20%</td></tr>
<tr><td>9</td><td>CLO-4</td><td>PLO-5</td><td>4%</td></tr>
<tr><td>10</td><td>CLO-4</td><td>PLO-5</td><td>4%</td></tr>
<tr><td>11</td><td>CLO-4</td><td>PLO-5</td><td>4%</td></tr>
<tr><td>12</td><td>CLO-4</td><td>PLO-5</td><td>4%</td></tr>
<tr><td>13</td><td>CLO-5</td><td>PLO-3</td><td>4%</td></tr>
<tr><td>14</td><td>CLO-5</td><td>PLO-3</td><td>4%</td></tr>
<tr><td>15</td><td>CLO-5</td><td>PLO-3</td><td>5%</td></tr>
<tr><td>16</td><td>CLO-5</td><td>PLO-3</td><td>30%</td></tr>
</table>

Final index is defined as follow:

</td></tr>
</table>

	<table><tr><th>Index</th><th>Converted Score</th><th>Score Range</th></tr><tr><td>A</td><td>4.00</td><td>$85 \leq A \leq 100$</td></tr><tr><td>A-</td><td>3.75</td><td>$80 \leq A- < 85$</td></tr><tr><td>B+</td><td>3.50</td><td>$75 \leq B+ < 80$</td></tr><tr><td>B</td><td>3.00</td><td>$70 \leq B < 75$</td></tr><tr><td>B-</td><td>2.75</td><td>$65 \leq B- < 70$</td></tr><tr><td>C+</td><td>2.50</td><td>$60 \leq C+ < 65$</td></tr><tr><td>C</td><td>2.00</td><td>$55 \leq C < 60$</td></tr><tr><td>D</td><td>1.00</td><td>$40 \leq D < 55$</td></tr><tr><td>E</td><td>0.00</td><td>$0 \leq E < 40$</td></tr></table>
Forms of Media	Slides, LCD projector, whiteboard, UNESA Learning Management System (LMS), unplugged activity materials, spreadsheet software (Excel or Google Sheets), Scratch, Python (Google Colab), GeoGebra
Literature	1. Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35. 2. Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. Educational Researcher, 42(1), 38-43. 3. Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. Journal of Science Education and Technology, 25(1), 127-147. 4. Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. Educational Research Review, 22, 142-158. 5. Polya, G. (1945). How to solve it. Princeton University Press. 6. Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Paper presented at the Annual Meeting of the American Educational Research Association, Vancouver, Canada. 7. Román-González, M., Pérez-González, J.-C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. Computers in Human Behavior, 72, 678-691.

	8. Dagienė, V., & Stupuriene, G. (2016). Bebras: A sustainable community building model for the concept based learning of informatics and computational thinking. Informatics in Education, 15(1), 25-44.
Note	*Based on the Regulation of the Minister of Education, Culture, Research, and Technology of the Republic of Indonesia Number 53 of 2023 on Quality Assurance in Higher Education, 1 credit unit (CU) corresponds to 45 hours of learning per semester. UNESA implements this as 170 minutes per week over 16 weeks, including the midterm and final examinations: 1 CU = 170 x 16 = 2,720 minutes or 45.3 hours. The total workload of 144 CU is therefore 144 x 45.3 = 6,523.2 hours. UNESA sets this workload as equivalent to 229 ECTS, so that 1 ECTS = 6,523.2 / 229 = 28.48 hours and 1 CU = 1.59 ECTS.