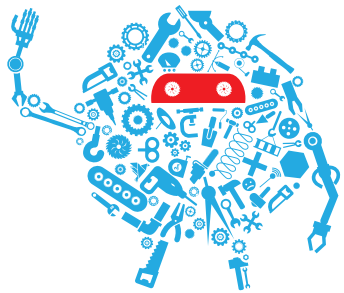


Tinker Education

ASK IMAGINE CREATE SHARE : CRITICAL THINKING

‘Tinker Education’ is a global STEM education brand of Korean top-tier EdTech company,

EMCAST’s STEM Education is driven by Computer Science

Tinker Education

Tinker Education teaches students Computer Science with the integration of Science, Mathematics, Engineering, and Art.
Tinker Education aims to teach students that the computer is a powerful tool for creativity and invention. We consider our students not as consumers but as future innovators and idea makers.
The students will be equipped to utilize technology confidently in the infinite world that is composed of 0 and 1. Tinker Education will enhance the student's natural talents by equipping them with critical thinking. We promise that Tinker Education students will become innovative thinkers who will surprise the world with dynamic solutions to real life needs.

CEO Kim Jungwoung

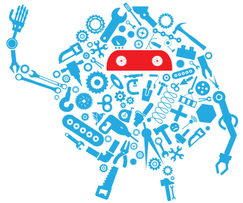
Research-based STEM Education

Curriculum, contents, and resources are delivered to students after in-depth research and assurance by the STEM R&D Center of HQ in Seoul, South Korea.

STEM R&D Center in Seoul, South Korea

Computer Science Framework of Tinker Education

COMPUTER SCIENCE *is powering approaches to many of our world's toughest challenges.*



Key Concepts

Learning Goals

To nurture a person who solves problems through a way that has never thought before and influences the world

Key Process

Educational content to achieve learning goals

A "key process" is something that a student of Computer Science should be able to do and know

Tinker Education's STEM Programs

Programming

Students gain competency in programming as well as mathematical and computational thinking for problem solving ability.

Physical Computing

Tinker's physical computing courses enable students to bring programming to life. They realize their ideas with various hands-on activities.

Life Science

By doing exciting experiments, students connect science to their everyday life and understand the principles of science that occur in their daily life.

Computer Science Key Concepts of Tinker Education

1. Logical Reasoning

Logical reasoning is the process of applying rules to problem solving. Logical reasoning selects an appropriate algorithm by predicting what happens at this stage. The predictions of each algorithm can be used to compare solutions and determine the best solution.

2. Algorithm

Step-by-step plans or guidelines for problem solving used for calculations, data processing, or automation

3. Decomposition

Separating the elements necessary for problem solving into small units

4. Pattern Recognition

Patterns can be said to be similarities or characteristics. It means the process of finding similarities and commonalities between small disaggregated problems that can help with problem-solving

5. Generalization

It is often used with patterns. It can be described as a generalization that produces a common problem-solving method that can solve various problems. Or, it is a generalization to use for solving a variety of new problems by applying a method that solved a similar problem in the past.

6. Abstraction

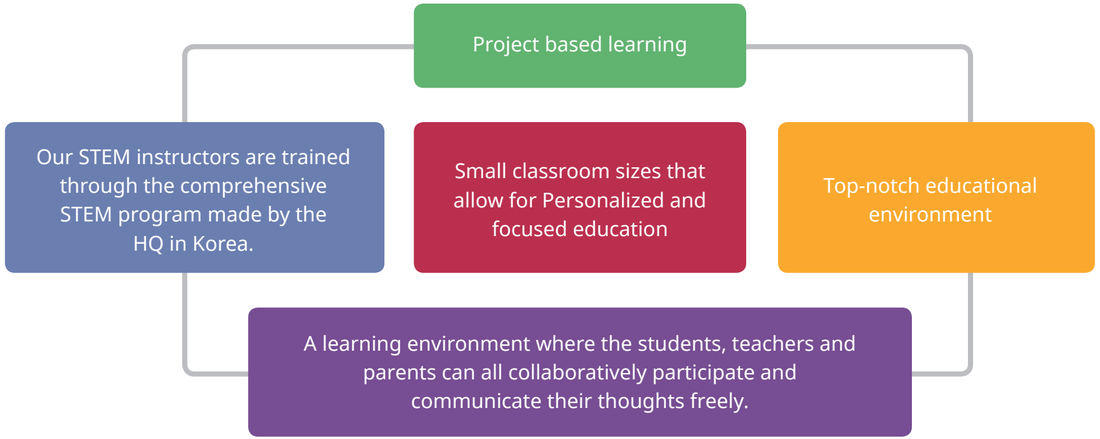
It is the process of concentrating on key ideas for the purpose of problem solving, gradually reducing the complexity of the problem, removing the irrelevant ones, and reducing the complexity by adding information for problem solving.

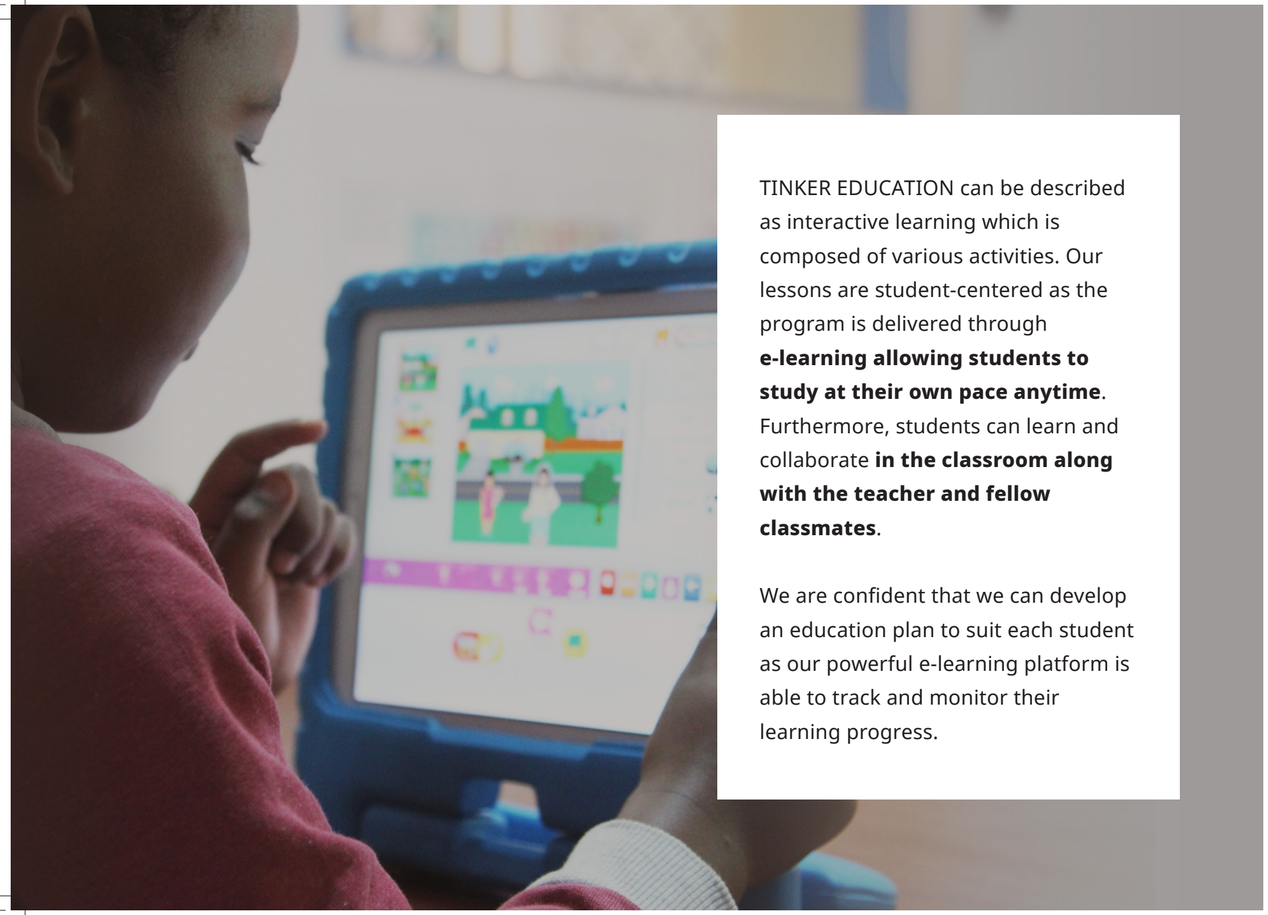
7. Evaluation

Decision making by judgment is about making judgments in an objective and systematic way as much as possible. Evaluation is done in daily situations that we face every day. We judge and decide what to do and how to think based on various factors.

Personalized Learning Path

TINKER EDUCATION is designed to realize student-centered learning that promotes learners to explore and reconstruct information to independently solve their problems, rather than the traditional model of students simply receiving knowledge from the instructor.





TINKER EDUCATION can be described as interactive learning which is composed of various activities. Our lessons are student-centered as the program is delivered through **e-learning allowing students to study at their own pace anytime**. Furthermore, students can learn and collaborate **in the classroom along with the teacher and fellow classmates**.

We are confident that we can develop an education plan to suit each student as our powerful e-learning platform is able to track and monitor their learning progress.

Learning Roadmap

EPL
Educational Programming Language

Physical computing and Hands on activity

3D Design and print

			CS + GAME		
Scratch Jr	Scratch Basic	Scratch intensive	Python Basic	Scratch intensive	
		CS + MATH		CS + MATH	
		CS + SCIENCE		CS + SCIENCE	
Beebot	Little bits Basic	Little bits STEM	Little bits cloud	Little bits STEM	
Ozobot		Little bits Invention		Little bits Invention	
	Microbits Basic	Microbits intensive		Microbits intensive	
		Microbits STEM		Microbits STEM	
Paper invention	Paper circuit	Lilypad Basic	e-Textile	Wearable Invention	
		Robot Basic	Robot intensive	Robot Invention	
	Sensor board	Arduino basic	Arduino intensive	I.O.T.	
	Tinker cad		Autocad 123	3D printer assemble	

Learning Path *Progress/Advancement by Stage*

Computer Science

STAGE 1
5-7 years old

i:Block 1 (Year 1)

1st Term (Level 1-1)	2nd Term (Level 1-2)	3rd Term (Level 1-3)

- Junior Coding with Scratch Junior
- STEM Unplugged Activities
- Hands-on Junior Course (Physical Computing: Bee-bot, Littlebits, etc.)

i:Block 2 (Year 2)

1st Term (Level 2-1)	2nd Term (Level 2-2)	3rd Term (Level 2-3)

- Beginner Coding with Scratch Basic
- STEM Unplugged Activities
- Hands-on Beginner Course (Physical Computing: Ozobot, Microbit, etc.)

i:Block 3 (Year 3)

1st Term (Level 3-1)	2nd Term (Level 3-2)	3rd Term (Level 3-3)

- Intermediate Coding
- STEM Unplugged Activities
- Hands-on Intermediate Course

STAGE 2
8-10 years old

i:Code 1 (Year 1)

1st Term (Level 1-1)	2nd Term (Level 1-2)	3rd Term (Level 1-3)

- Beginner Coding with Scratch Basic
- STEM Unplugged Activities
- Hands-on Beginner Course (Physical Computing: Ozobot, Mirobit etc.)

i:Code 2 (Year 2)

1st Term (Level 2-1)	2nd Term (Level 2-2)	3rd Term (Level 2-3)

- Intermediate Coding
- STEM Unplugged Activities
- Hands-on Intermediate Course (Physical computing: CODEino, Microbit etc.)

i:Code 3 (Year 3)

1st Term (Level 3-1)	2nd Term (Level 3-2)	3rd Term (Level 3-3)

- Advanced Coding
- STEM Unplugged Activities
- Hands-on Advanced Course

STAGE 3
11-15 years old

i:Imagine 1 (Year 1)

1st Term (Level 1-1)	2nd Term (Level 1-2)	3rd Term (Level 1-3)

- Advanced Coding with Python
- STEM Unplugged Activities
- Hands-on Advanced Course (Physical Computing: Microbit, Basic Arduino, etc.)

i: Imagine 2 (Year 2)

1st Term (Level 2-1)	2nd Term (Level 2-2)	3rd Term (Level 2-3)

- Professional Coding I : Web Development, App Inventor
- Hands-on Professional Course (Intensive Arduino, etc.)

i: Imagine 3 (Year 3)

1st Term (Level 3-1)	2nd Term (Level 3-2)	3rd Term (Level 3-3)

- Professional Coding II : Web Development, App Inventor
- Hands-on Professional Course

Science with Experiments

STAGE 1 5-7 Years Old → **STAGE 2** 8-11 Years Old → **STAGE 3** 12 & 14 Years Old → **STAGE 4** 15 & 16 Years Old

** Each stage is a year course. The science course of Tinker Education follows the national curriculum in England – science programs of study.*

Tinker Education

@TinkerEducation

www.tinkeredu.net | **T.** 0715-978-986 | **E.** support@tinkeredu.net | **A.** 2nd Floor, Edulink International College Ndemi Rd, Nairobi, Kenya