Grade Level 11/12
1 Year

Teacher Tim Sampson
Email: tsampson@dishs.tp.edu.tw

Course Description

Welcome to Computer Science 9 at Dominican International School. Computer Science Discoveries 9 is the first half of an introductory computer science course focusing on Innovation and Impact. Students in this course will take the second half in grade 10. The two parts of this course take a wide lens on computer science by covering topics such as programming, physical computing, HTML/CSS, and data. Students are empowered to create authentic artifacts and engage with CS as a medium for creativity, communication, problem solving, and fun. This course uses Code.org's CS Discoveries Curriculum, for more details, please see the 2022-23 Curriculum Guide.

Content

The content covered in this year includes:
- Unit 2 - Web Development
- Unit 3 - Interactive Animations and Games

Objectives

Upon completion of unit 1, students should be able to:
- Practice using a problem solving process to address a series of puzzles, challenges, and real world scenarios.
- Learn how computers input, output, store, and process information to help humans solve problems.
- Design an application that helps solve a problem of your choosing.

Upon completion of unit 2, students should be able to:
- Create and share the content on your own web pages.
- Structure and style your pages using HTML and CSS.
- Practice valuable programming skills such as debugging and commenting.
- Create a personal website that you can publish to the Internet.

Upon completion of unit 3, students should be able to:
- Build on your coding experience as you program animations, interactive art, and games in Game Lab.
- Design simple shapes and builds up to more sophisticated sprite-based games, using the same programming concepts and the design process computer scientists use daily.
- Develop a personalized, interactive program.

Classroom Practices

The 6 Main Classroom Practices of CS Discoveries:
- Lead Learner
- Pair Programming
- Think-Pair-Share
- Authentic Choice
- Unplugged Activities
- Peer Feedback

Student Practices

Students in CS Discoveries work in a wide array of contexts, but these experiences are tied together by a core set of practices they develop throughout the course
- Problem Solving
- Persistence
- Creativity
- Collaboration
- Communication

ESLRs D'TORCH (Truthful, Organized, Reflective, Courageous and Helpful)

In CS classes the categories of the D'TORCH most practiced and assessed are:
- Organized - Students utilize Google Classroom to edit, submit and keep track of their assignments.
- Reflective - Students will regularly write activity reflections in their online journal.
- Helpful - Students are empowered to ask for and provide explanations and give examples to help classmates through particularly difficult problems.

Class Expectations

- Come to class on time and be prepared
- Have a positive attitude and be willing to learn.
- Respect yourself, others, and our school.
- Always complete your work and try your best.
- Actively participate, listen carefully, but don't speak out of turn.
- All assignments must be completed.

Homework and Quiz Rules

- All assignments must be turned in on the day they are due.
- 1 day late = Maximum of only 60%
- 2+ days late = Project-I & Only 60%
- If a student has been absent, it is his/her duty to find out what work is due, and hand it in a day later.
- All assignments must satisfactorily be completed.
- If you are absent on the day of the quiz, you will only be able to get a maximum of 60%.

Classroom Rules

- All students are expected to follow the rules. Consequences will follow if rules are broken.
- Read and follow the standard school rules.

- Be on time and neatly dressed, in full school uniform.
- Speak in ENGLISH ONLY.
- Respect your teachers, fellow students and their property.
- Keep your seating space and classroom clean and neat.
- No eating or drinking in the ICT Labs.
- Ask permission to leave the class.

Academic Dishonesty means employing a method or technique or engaging in conduct in an academic endeavor that contravenes the standards of ethical integrity expected at DIS. Academic dishonesty includes but is not limited to, the following:
1. Purposely incorporating the ideas, words of sentences, paragraphs, or parts thereof without appropriate acknowledgment and representing the product as one's own work; and
1. Representing another's intellectual work such as photographs, paintings, drawings, sculpture, or research or the like as one's own, including failure to attribute content to an AI.
2. Employing a tutor, making use of Artificial Intelligence without acknowledgement, getting a parent to write a paper or do an assignment, paying for an essay to be written by someone else and presented as the student's own work.
3. Committing any act that a reasonable person would conclude, when informed of the evidence, to be a dishonest means of obtaining or attempting to obtain credit for academic work.

Any act of academic dishonesty will result in an automatic zero on the entire assignment

Discipline
- Verbal warning
- Write-Up, entered into the discipline system and then referral to the Discipline Office.
- Parent-Teacher conference as required.

Links, tools and references:
- https://code.org/educate/csd
- https://developer.mozilla.org/en-US/docs/Learn
- https://www.w3schools.com/
- Web Lab — A browser-based tool for creating and publishing HTML and CSS web sites
- Game Lab — A browser-based JavaScript programming environment designed to create sprite-based drawings, animations and games, with the ability to freely switch between programming in blocks or text

Schedule for Computer Science Discoveries 9 Innovation and Impact

Week / Date	Topic / Projects / Assessments
(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week 1 **Aug 10th to 11th** **Only 2 School Days** *10 ~ First Day / Orientation Day*	**Unit 1 - Problem Solving and Computing** The Problem Solving and Computing unit is a highly interactive and collaborative introduction to the field of computer science, as framed within the broader pursuit of solving problems. Through a series of puzzles, challenges, and real world scenarios, students are introduced to a problem solving process that they will return to repeatedly throughout the course. Students then learn how computers input, output, store, and process information to help humans solve problems within the context of apps. The unit concludes with students designing an app that helps solve a problem of their choosing. **Lesson 2: The Problem Solving Process** This lesson introduces the formal problem solving process that the class will use

	over the course of the year: Define - Prepare - Try - Reflect. The class relates these steps to the problem from the previous lesson, then to a problem they are good at solving, then to a problem they want to improve at solving. At the end of the lesson, the class collects a list of generally useful strategies for each step of the process to put on posters that will be used throughout the unit and year.
Week 2 **Aug 14th to 18th** *15 ~ Opening Mass*	# Unit 2 - Web Development In the Web Development unit, students are empowered to create and share the content on their own web pages. They begin by thinking about the role of the web, and how it can be used as a medium for creative expression before creating their own personal web pages. As students develop their pages and begin to see themselves as programmers, they are encouraged to think critically about the impact of sharing information online and how to be more critical content consumers. They are also introduced to problem solving as it relates to programming, as they learn valuable skills such as debugging, using resources, and teamwork. At the conclusion of the unit, students work together to create a website to address a problem. **Lesson 1: Exploring Websites** This lesson covers the purposes that a web page might serve, both for the users and the creators. The class explores a handful of sample web pages and describes how each of those pages is useful for users and how they might also serve their creators. **Lesson 2: Intro to HTML** This lesson introduces HTML as a solution to the problem of how to communicate both the content and structure of a website to a computer. The lesson begins with a brief unplugged activity that demonstrates the challenges of effectively communicating the structure of a web page. Then, the class looks at an HTML page in Web Lab and discusses how HTML tags help solve this problem, before using HTML to write their first web pages of the unit.
Week 3 **Aug 21st to 25th**	**Lesson 3: Headings** This lesson continues the introduction to HTML tags, this time with headings. The class practices using heading tags to create page and section titles and learns how the different heading elements are displayed by default. **Lesson 4: Mini-Project: HTML Web Page** In this lesson, the class creates personal web pages on a topic of their choice. The lesson starts with a review of HTML tags. Next, the class designs web pages, first identifying the tags needed to implement them, and then creating the pages in Web Lab.
Week 4 **Aug 28th to Sep 1st**	**Lesson 5: Digital Footprint** This lesson takes a step back from creating the personal website to talk about the personal information that people choose to share digitally. The class begins by discussing what types of information they have shared on various websites, then they look at several sample social media pages to see what types of personal information could be shared intentionally or unintentionally. Finally, the class comes up with a set of guidelines to follow when putting information online.**Lesson 6: Styling Text with CSS** This lesson introduces CSS as a way to style elements on the page. The class learns the basic syntax for CSS rule-sets and then explores properties that impact HTML text elements. Finally, they discuss the differences between content, structure, and style when making a personal web page.
Week 5 **Sep 4th to 8th** *8 ~ Holy Mass & VIP Induction*	**Lesson 7: Mini-Project: Your Personal Style** In this lesson, students create their own styled web pages. The lesson starts with a review of the CSS. They then design the web page, identify which CSS properties they will need, and create their web pages in Web Lab.
Week 6 **Sep 11th to 15th** *12-14 ~ Pre-Exam Days*	**Lesson 8: Intellectual Property** Starting with a discussion of their personal opinions on how others should be allowed to use their work, the class explores the purpose and role of copyright for both creators and users of creative content. They then move on to an activity exploring the various Creative Commons licenses as a solution to the difficulties of dealing with copyright.

	Lesson 9: Using Images The class starts by considering the ethical implications of using images on websites, specifically in terms of intellectual property. They then learn how to add images to their web pages using the <img> tag and how to cite the image sources appropriately.
Week 7 **Sep 18th to 22nd**	**Lesson 10: Websites for Expression** This lesson introduces websites as a means of personal expression. Students first discuss the different ways that people express and share their interests and ideas, then they look at a few exemplar websites made by students from a previous course. Finally, everyone brainstorms and shares a list of topics and interests to include in a personal website, creating a resource for developing a personal website in the rest of the unit.**Lesson 11: Styling Elements with CSS** This lesson continues the introduction to CSS style properties, this time focusing more on non-text elements. The class begins by investigating and modifying the new CSS styles on a Desserts of the World page. Afterwards, everyone applies this new knowledge to their personal websites.
Week 8 **Sep 25th to 29th** **No Classes** *25-28 ~Teacher's Conference* *29 – Moon Festival Holiday*	**Q1 Final Exam**
Week 9 **Oct 2nd to 6th** **3 Days of Class** *5-6 ~Q1 Exams*	**Lesson 12: Your Web Page - Prepare** In this lesson, students engage in the "prepare" stage of the problem solving process by deciding what elements and style their web pages will have. They review the different HTML, CSS, and digital citizenship guidelines, then design and plan their pages, as well as download and document the images they will need. Afterwards, they reflect on how their plan will ensure that the website does what it is designed to do.

2nd QUARTER – TENTATIVE COURSE CONTENT

Week / Date	Topic / Projects / Assessments
(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week 1 (10) **Oct 9th to 13th** **3 Days of Class** *9-10 – Double 10 Holiday*	**Lesson 13: Project - Personal Web Page** After quickly reviewing the debugging process, the class goes online to create the pages that they have planned out in previous lessons, with the project guides as a reference. Afterwards, they engage in a structured reflection and feedback process before making any final updates.
Week 2 (11) **Oct 16th to 20th**	**Lesson 14: Websites for a Purpose** In this lesson, students explore the different reasons people make websites. After brainstorming various reasons that they visit websites, they investigate sample web sites that have been created to address a particular problem and decide what different purposes those websites might serve for the creators. The class then thinks of problems they might want to solve with their own websites.
Week 3 (12) **Oct 23rd to 27th**	**Lesson 15: Team Problem Solving** Teams work together to set group norms and brainstorm what features they would like their websites to have. The class starts by reflecting on what makes teams successful. Teams then make plans for how they will interact and achieve success in their own projects before brainstorming ideas for their website projects. **Lesson 16: Sources and Research** This lesson covers how to find relevant and trustworthy information online. After viewing and discussing a video about how search engines work, students search for information relevant to their sites, then analyze the sites for credibility to decide which are appropriate to use on their own website.

Week 4 (13) **Oct 30ᵗʰ to Nov 3ʳᵈ** *1 - All Saint's Day Mass*	**Lesson 17: CSS Classes** This lesson introduces CSS classes, which allow web developers to treat groups of elements they want styled differently than other elements of the same type. Students first investigate and modify classes on various pages, then create their own classes and use them to better control the appearance of their pages. Teams then reflect on how they could use this skill to improve their websites. **Lesson 18: Planning a Multi-Page Site** The class works in teams to plan out the final web sites, including a sketch of each page. They then download the media that they will need for their sites. At the end of the activity, they decide how the work will be distributed among them and report whether the entire team agreed to the plan.
Week 5 (14) **Nov 6ᵗʰ to 10ᵗʰ**	**Lesson 19: Linking Pages** The class begins this lesson by looking online for the internet's first web page and discussing how its use of links was what started the web. They then transition to Web Lab where they learn how to make their own links, as well as good conventions that make it easier for users to navigate on a page. Last, they reflect on their team project and what their personal goals are for the final stretch. **Lesson 20: Project - Website for a Purpose** In this lesson, teams are finally able to code the pages that they have been planning. Using the project guide, the team works together and individually to code all of their pages, then puts all of the work together into a single site.
Week 6 (15) **Nov 13ᵗʰ to 17ᵗʰ**	**Lesson 21: Peer Review and Final Touches** This lesson focuses on the value of peer feedback. The class first reflects on what they are proud of, and what they would like feedback on. They then give and get that feedback through a structured process that includes the project rubric criteria. Afterwards, everyone puts the finishing touches on their sites and reflects on the process before a final showcase.
Week 7 (16) **Nov 20ᵗʰ to 24ᵗʰ**	# Unit 3 - Interactive Animations and Games In the Animations and Games unit, students build on their coding experience as they create programmatic images, animations, interactive art, and games. Starting off with simple, primitive shapes and building up to more sophisticated sprite-based games, students become familiar with the programming concepts and the design process computer scientists use daily. They then learn how these simpler constructs can be combined to create more complex programs. In the final project, students develop a personalized, interactive program. Along the way, they practice design, testing, and iteration, as they come to see that failure and debugging are an expected and valuable part of the programming process. **Lesson 1: Programming for Entertainment** The class is asked to consider the "problems" of boredom and self expression, and to reflect on how they approach those problems in their own lives. From there, they will explore how Computer Science in general, and programming specifically, plays a role in either a specific form of entertainment or as a vehicle for self expression. **Lesson 2: Plotting Shapes** This lesson explores the challenges of communicating how to draw with shapes and use a tool that introduces how this problem is approached in Game Lab. The class uses a Game Lab tool to interactively place shapes on Game Lab's 400 by 400 grid. Partners then take turns instructing each other how to draw a hidden image using this tool, which accounts for many of the challenges of programming in Game Lab.
Week 8 (17) **Nov 27ᵗʰ to Dec 1ˢᵗ**	**Lesson 3: Drawing in Game Lab** The class is introduced to Game Lab, the programming environment for this unit, and begins to use it to position shapes on the screen. The lesson covers the basics of sequencing and debugging, as well as a few simple commands. At the end of the lesson, students will be able to program images like the ones they made with the drawing tool in the previous lesson.

	Lesson 4: Shapes and Parameters In this lesson, students continue to develop a familiarity with Game Lab by manipulating the width and height of the shapes they use to draw. The lesson kicks off with a discussion that connects expanded block functionality (e.g. different sized shapes) with the need for more block inputs, or "parameters." Finally, the class learns to draw with versions of ellipse() and rect() that include width and height parameters and to use the background() block.
Week 9 (18) **Dec 4th to 8th** 8 - Foundation Day Celebrations	**Q2 Final Exam**
Week 10 (19) **Dec 11th to 15th** **3 Days of Class** *14-15 ~ Q2 Exams*	**Lesson 5: Variables** This lesson introduces variables as a way to label a number in a program or save a randomly generated value. The class begins the lesson with a very basic description of the purpose of a variable and practices using the new blocks, then completes a level progression that reinforces the model of a variable as a way to label or name a number.
Dec 19th to Jan 2nd	**Christmas Break**

3rd QUARTER – TENTATIVE COURSE CONTENT

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 (20) **Jan 3rd to 5th** **3 Days of Class** *4 ~ New Year Mass*	**Lesson 6: Random Numbers** Students are introduced to the randomNumber() block and how it can be used to create new behaviors in their programs. They then learn how to update variables during a program and use those skills to draw randomized images.
Week 2 (21) **Jan 8th to 12th**	**Lesson 7: Sprites** In order to create more interesting and detailed images, the class is introduced to the sprite object. The lesson starts with a discussion of the various information that programs must keep track of, then presents sprites as a way to keep track of that information. Students then learn how to assign each sprite an image, which greatly increases the complexity of what can be drawn on the screen.**Lesson 8: Sprite Properties** Students extend their understanding of sprites by interacting with sprite properties. The lesson starts with a review of what a sprite is, then moves on to Game Lab for more practice with sprites, using their properties to change their appearance. The class then reflects on the connections between properties and variables.
Week 3 (22) **Jan 15th to 19th**	**Lesson 9: Text** This lesson introduces Game Lab's text commands, giving students more practice using the coordinate plane and parameters. At the beginning of the lesson, they are asked to caption a cartoon created in Game Lab. They then move onto Code Studio where they practice placing text on the screen and controlling other text properties, such as size. **Lesson 10: Mini-Project - Captioned Scenes** After a quick review of the code learned so far, the class is introduced to the first creative project of the unit. Using the problem solving process as a model, students define the scene that they want to create, prepare by thinking of the different code they will need, try their plan in Game Lab, then reflect on what they have created. In the end, they also have a chance to share their creations with their peers.
Week 4 (23) **Jan 22nd to 26th**	**Lesson 11: The Draw Loop** This lesson introduces the draw loop, one of the core programming paradigms in Game Lab. Students learn how to combine the draw loop with random numbers to manipulate some simple animations first with dots and then with sprites. **Lesson 12: Sprite Movement** In this lesson, the class learns how to control sprite movement using a construct called the counter pattern, which incrementally changes a sprite's properties. After brainstorming different ways that they could animate sprites by controlling their properties, students explore the

	counter pattern in Code Studio, using the counter pattern to create various types of sprite movements.
Week 5 (24) **Jan 29ᵗʰ to Feb 2ⁿᵈ**	**Lesson 13: Mini-Project - Animation** In this lesson, the class is asked to combine different methods from previous lessons to create an animated scene. Students first review the types of movement and animation that they have learned, and brainstorm what types of scenes might need that movement. They then begin to plan out their own animated scenes, which they create in Game Lab. **Lesson 14: Conditionals** This lesson introduces students to booleans and conditionals, which allow a program to run differently depending on whether a condition is true. The class starts by playing a game in which they respond according to whether particular conditions are met. They then move to Code Studio where they learn how the computer evaluates boolean expressions, and how they can be used to structure a program.
Week 6 (25) **Feb 5ᵗʰ to 9ᵗʰ** **3 Days of Class** *8-9 ~ CNY*	**Lesson 15: Keyboard Input** Following the introduction to booleans and if statements in the previous lesson, students are introduced to a new block called keyDown(), which returns a boolean and can be used in conditional statements to move sprites around the screen. By the end of this lesson they will have written programs that take keyboard input from the user to control sprites on the screen. **Lesson 16: Mouse Input** The class continues to explore ways to use conditional statements to take user input. In addition to the keyboard commands learned yesterday, they learn about several ways to take mouse input. They also expand their understanding of conditional to include else, which allows for the computer to run a certain section of code when a condition is true, and a different section of code when it is not.
Feb 8ᵗʰ to 16ᵗʰ	**CNY Holiday**
Week 7 (26) **Feb 19ᵗʰ to 23ʳᵈ** *19 ~ Lenten Mass* *21-23 ~ Pre-Exam Days*	**Lesson 17: Project - Interactive Card** In this cumulative project for Chapter 1, students plan for and develop an interactive greeting card using all of the programming techniques they've learned to this point.
Week 8 (27) **Feb 26ᵗʰ to March 1ˢᵗ** **4 Days of Class** *28 ~ 228 Memorial Day Holiday*	**Q3 Final Exam**
Week 9 (28) **March 4ᵗʰ to 8ᵗʰ** **4 Days of Class** *8 ~ Q3 Exams*	**Lesson 18: Velocity** After a brief review of how the counter pattern is used to move sprites, the class is introduced to the idea of hiding those patterns in a single block, in order to help manage the complexity of programs. They then head to Code Studio to try out new blocks that set a sprite's velocity directly, and look at the various ways that they are able to code more complex behaviors in their sprites.

4th QUARTER – TENTATIVE COURSE CONTENT

(NB: Depending on time and interest, the teacher may delete and/or add other selections.)	
Week / Date	Topic / Projects / Assessments
Week 1 (29) **March 11ᵗʰ to 15ᵗʰ** **4 Days of Class** *11 ~ Q3 Exams* *12 ~ Q4 Begins*	Lesson 19: Collision Detection In this lesson, the class learns about collision detection on the computer. Working in pairs, they explore how a computer could use math, along with the sprite location and size properties, to detect whether two sprites are touching. They then use the isTouching() block to create different effects when sprites collide and practice using the block to model various interactions. **Lesson 20: Mini-Project - Side Scroller** Students use what they have learned about collision detection and setting velocity to create simple side scroller games. After looking at a sample side scroller game, they brainstorm what

	sort of side scroller they would like, then use a structured process to program the game in Code Studio.
Week 2 (30) **March 18th to 22nd** *18-21 ~ Fire Drill*	**Lesson 21: Complex Sprite Movement** The class learns to combine the velocity properties of sprites with the counter pattern to create more complex sprite movement. After reviewing the two concepts, they explore various scenarios in which velocity is used in the counter pattern, and observe the different types of movement that result. In particular, students learn how to simulate gravity. They then reflect on how they were able to get new behaviors by combining blocks and patterns that they already knew. **Lesson 22: Collisions** In this lesson, the class programs their sprites to interact in new ways. After a brief review of how they used the isTouching block, students brainstorm other ways that two sprites could interact. They then use isTouching to make one sprite push another across the screen before practicing with the four collision blocks (collide, displace, bounce, and bounceOff).
March 25th to Apr 5th	**Easter Holiday**
Week 3 (31) **Apr 8th to 12th** *10 ~ Easter Mass*	**Lesson 23: Mini-Project - Flyer Game** Students use what they have learned about simulating gravity and the different types of collisions to create simple flyer games. After looking at a sample flyer game, they brainstorm what sort of flyer they would like, then use a structured process to program the game in Code Studio. **Lesson 24: Functions** This lesson covers functions as a way for students to organize their code, make it more readable, and remove repeated blocks of code. The class learns that higher level or more abstract steps make it easier to understand and reason about steps, then begins to create functions in Game Lab.
Week 4 (33) **Apr 15th to 19th**	**Lesson 25: The Game Design Process** This lesson introduces the process that students will use to design games for the remainder of the unit. This process is centered around a project guide that asks students to define their sprites, variables, and functions before they begin programming their game. They walk through this process in a series of levels. At the end of the lesson, students have an opportunity to make improvements to the game to make it their own. **Lesson 26: Using the Game Design Process** In this multi-day lesson, the class uses the problem solving process from Unit 1 to create a platform jumper game. After looking at a sample game, they define what their games will look like and use a structured process to build them. Finally, the class reflects on how the games could be improved, and implements those changes.
Week 5 (34) **Apr 22th to 26th** *22-26 ~ AP Mock Exams*	**Lesson 27: Project - Design a Game** Students plan and build original games using the project guide from the previous two lessons. Working individually or in pairs, they plan, develop, and give feedback on the games. After incorporating the peer feedback, students share their completed games.
Week 6 (35) **Apr 29th to May 3rd** *1-2 ~ Pre-Exam* *1-10~ Final Exams (K, 5, 8, 12 only)* *4/29 – 5/10 ~ AP Exams*	**Lesson 27: Project - Design a Game - Continued**
Week 7 (36) **May 6th to 10th** *1-10~ Final Exams (K, 5, 8, 12 only)* *4/29 – 5/10 ~ AP Exams*	**Lesson 27: Project - Design a Game - Continued**
Week 8 (37) **May 13th to 17th** **2 Days of Class** *15-16 ~ Q4 Exams* *17 ~ Record Day*	Q4 Exam
Week 9 (38)	
Week 10 (39)	CS Discoveries 9 Web Development, Interactive Animations and Games

May 20ᵗʰ to 24ᵗʰ **ACTIVITIES:** *Double check the school calendar and emails from the administration.*	-- *20-24 ~ Student Clearance Days* *21 ~ Baccalaureate Mass for Graduating classes* *22 & 23 ~ Middle & High School Sports Day* *23 ~ Pre-Kindergarten & Gr. 1 - 4 Recognition/Kindergarten Graduation/Gr. 5 Promotion* *24 ~ Gr. 6 – 7 Recognition and Gr. 8 Graduation* *24 ~ Lower School Sports Day*
May 27ᵗʰ to 31ˢᵗ **ACTIVITIES:** *Double check the school calendar and emails from the administration.*	*27 ~ House Culminating Activity* *28 ~ Gr. 9-11 Recognition and Gr. 12 Graduation* *29 ~ Class Party* *30 ~ Last Day of School & Report Card Distribution (half day)* *31 ~ Teachers/Staff Meeting*

The end ~ Have a great summer 😊