Lesson 8.1 - Project Planning Summary + Brainstorming and Evaluating ## Learning Objectives Students will be able to... * Recall project planning basics from last semester * Identify factors to use when choosing between project ideas * Rank a group of proposed project ideas using the identified factors ## Materials/Preparation * [Project Ideas for Students](project_ideas.md.html) * [Editable Project Ideas] (https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/8_unit/01_lesson/project_ideas.docx) * [Printable Project Ideas] (https://github.com/TEALSK12/2nd-semester-introduction-to-computer-science/raw/master/units/8_unit/01_lesson/project_ideas.pdf) ## Pacing Guide | Duration | Description | | | --------- | --- | | | 5 minutes | Do Now | | 10 minutes | Introduce Project and Review Design| | 5 minutes | Brainstorming | | 10 minutes | Pitch writing | | 20 minutes | Peer review | | 5 minutes | Debrief and wrap-up| ## Instructor's Notes ### 1. Do Now * Project the Do Now on the board, circulate around the class to check that students are working and understand the instructions. ### 2. Introduce Project and Review Design #### Discussion * Talk about how far students have come this semester * Ask students to think back to the start of the semester and remember how little they knew about Python programming. * Show a lab or assignment from early on and remind them that, not that long ago, this was challenging, whereas it now seems nearly trivial (hopefully). #### Project Overview * Explain that, for their final project, the students will get to design and build a program of their own choosing. * Point out that this will involve more than just writing code * There will be planning, design, scheduling, and other project management tasks * Emphasize that students will be graded on not only the program they produce, but the process they used to design, plan, and implement it * Here are some [Project Ideas](projectideas.md.html) ##### Demonstration * Demonstrate a few example projects (with as much variety as possible). * Try to hit a bunch of different types of programs. * Many students will gravitate towards games, but other options include simulations, productivity tools, musical projects, animations, and more. ##### Rubric Discussion * Distribute the project rubric and point out key aspects of the requirements * Point out the steps in the process and that each one is equally important * Specifically mention the large number of points for things _not_ related to coding #### Review process and identify first steps * Display the Spec and Plan documents from last semester * Ask students to identify the steps in the design and planning process as discussed in [do_now](do_now.md.html). * Remind students that all steps are vital, and that thorough and thoughtful planning and design can make the coding phase much easier. * Inform students that today they will take the first steps in designing their final project. #### Brainstorming * Give students 3-4 minutes to brainstorm and write down as many project ideas as they can. This should be done mostly in silence. * At this point, there should be minimal detail, no evaluation or rejection of ideas, and no discussion. In particular, students should not think about the difficulty or "coolness" of the project yet. Just write down ideas. * If desired, have each student share one idea. Do not allow discussion, criticism, or explanation-- each idea should be summarized in only a few words or a single sentence. #### Pitch writing * Have students pick their **3** favorite ideas and write a "pitch" for the project. * A pitch should be no more than a short paragraph and should describe the basic, high-level features of the project. * The pitch should _not_ include any implementation details (scripts, sprites, etc.). * Pitches should include a moderate level of specificity-- enough for someone to imagine how the app will work, but not so much to get bogged down. * Enforce the "one short paragraph" restriction. * If a student is having difficulty developing a pitch for an idea, that might be a sign that the idea is not fully-formed enough to be a final project. * If a student is having trouble keeping the pitch short, the project may be too complex to complete in the available time. #### Peer Review * Pair students up and have students take turns reading one of their pitches to their partner and asking for feedback. * Partners should ask questions to help identify both the best and worst parts of each pitch. * Remind students to keep all feedback constructive, respectful, and professional. * Students should not criticize each other's ideas, but can point out potential concerns. * Students should take notes during their conversations and refine their pitches based on their partner's feedback and their own realizations. * If time allows (or over the course of multiple days), repeat this process with new partners. ### 3. Debrief * At this points, students should have between one and three pitches that are well-defined and reasonably well fleshed-out. * For Homework, students should consider their pitches and rank them in order or which they would most like to pursue as their final project. * Make sure students don't just pick the "coolest" sounding idea, but also consider the technical challenges, amount of time available, and their own interest in and willingness to see the project through to completion. ## Accommodation/Differentiation * If students are having difficulty coming up with project ideas, encourage them to think about existing software. While simply recreating an existing app should be a last resort, thinking about apps they already know can help students come up with functionality they might like to include. * If your class is fairly self-sufficient and mature,

you can consider allowing students to "borrow" an idea from a classmate if they find one they like better than any of their own. Make sure the person who had the idea is OK with it being borrowed, and emphasize that the students must each build their own version. * This can be a bit dangerous, as it puts the student somewhat behind in the process right from the start, since they won't have as much context having not written the pitch themselves. Consider carefully whether this is a good idea. * Encourage each student to pick a project that fits his or her level of technical skill. Make sure students are not overcommitting and choosing projects they do not have the skills to complete. Also try to discourage stronger students from choosing simpler projects in an effort to do less work. ## Forum discussion [Lesson 8.1 - Project Planning Summary + Brainstorming and Evaluating (TEALS Discourse Account Required)](https://forums.tealsk12.org/c/2nd-semester-unit-8-final-project/lesson-8-01-final-project-brainstorming-and-evalu)