| Overview: Integrated Curricular Unit with Differentiation | |
|---|---|
| Unit Title: Putting Algorithms On The Map | |
| Theoretical Framework: Constructionism | |
| Unit Theme: Algorithms | |
| Integration Pathway: Shared Model (Fogarty, 1992) | |
| Problem-Solving Task: Design an app that helps users quickly locate the nearest location of a franchise or service | |
| Unit Objectives:<br>1.Understand and use mathematical techniques commonly used in computer science applications<br>2.Design and implement Python code algorithms and user interfaces for mobile applications<br>3.Experience the process of funding and marketing a mobile application | |
| Standards/Guidelines/Expectations by Discipline | |
| Discipline 1: Mathematics<br><br>Iterations<br>Inequalities<br>Voronoi diagrams<br>Combinations<br>Information theory | Discipline 2: Digital Design<br><br>Flow charts<br>Python syntax<br>User Interface Design<br>Market research<br>User surveys |
| Desired Unit Results | |
| Six As (How will you incorporate each A in the unit? | |
| 1. Authenticity | • Students document and share their work using the same software tools as those in the workplace<br>• Students can choose to target their app towards a user group that frequents locations that are meaningful in their own lives<br>• The unit culminates in an app launch event |
| 2. Academic Rigor | • Lessons and project deliverables will be mapped to the appropriate discipline specific KUDs<br>• Adult experts check student content knowledge before allowing students to proceed with work beyond certain points |
| 3. Applied Learning | • Students complete all tasks in their project teams for the duration of the unit, so that newly acquired disciplinary knowledge and skills are immediately applied in the project context<br>• Students learn and apply project management techniques |

| | |
|---|---|
| 4. Active Exploration | • The results of spreadsheets, code, and graphs created as part of disciplinary lessons are used in the larger project context<br>• Students collect and analyze authentic data such as search traffic, downloads and user surveys<br>• Students create a “pitch deck” and present it to a panel of potential investors from the school community |
| 5. Adult Relationships | • Classroom visits by industry experts to consult with student project teams<br>• Guest speaker to attend the project launch<br>• Community experts have input into the project scope and assessment criteria |
| 6. Assessment | • A rubric will be used to judge the quality of student-developed project success criteria and use of said criteria to self-assess<br>• Assessment will include integrated communication forms such as app support documentation and promotional material, and project management documentation such as a project journal and lessons learned file |

Essential Questions (List at least 3-5 which connect the disciplines.)
These questions will help students discover the natural connections among the specific discipline fields:

- How can we design a product for an online audience?
- How do brands decide on new store locations?
- Should organizations always use mathematics to make decisions?

| Learners will know: refer to specific lesson objectives | Learners will be skilled at: refer to specific lesson objectives |
|---|---|

| Evidence of Learning | |
|---|---|
| Evaluative Criteria:<br><br>-Accuracy of mathematical calculations<br>-Efficiency of code and algorithms<br>-Persuasiveness of pitch deck slides | Assessment Evidence<br><br>Formative – see lesson details below<br>Summative – see lesson details below<br>Culminating - app launch event<br>Performance Task – pitch deck presentation<br>Other Evidence – project documentation |

| Lesson Plan 1<br>Summary of Key Learning Interactions and Instruction | | | |
|---|---|---|---|
| Lesson 1: Constructing Voronoi Diagrams | | | |
| Learning Objectives:<br>-determine the equation of a perpendicular bisector between two points on a map<br>-construct a Voronoi diagram by finding the intersection point of 3 perpendicular bisectors<br>-interpret the meaning of Voronoi regions on a map in a real-life context<br>-create shaded Voronoi diagrams on digital maps using Geogebra | | | |
| Formative Assessment: Desmos activity that introduces students to the concept of a Voronoi diagram as it relates to regions of closest proximity to certain locations on a map.<br><br>Summative Assessment: students create a “looks-like” prototype of their main app screen in Geogebra by overlaying a shaded Voronoi diagram, constructed from real-world data points for their chosen brand, over an actual map of their city. | | | |
| Interactions/Activities | Differentiation | Materials/Resources | Field Experiences/Adult Relationships |
| Description: Equations of perpendicular bisectors<br>Steps:<br>-short Desmos activity to explore concept<br>-teacher addresses misconceptions and formalizes the process | Intervention: perpendicular gradients and finding the equation of a straight line revision material<br><br>Extension: finding the shortest distance from a line to a point and applications | Closer to what?<br>(Desmos Activity) | Find two points on a map of their city that will be used in their app and find the equation of the perpendicular |

| | | | |
|---|---|---|---|
| -perpendicular bisectors quiz to check for understanding | | | bisector between them. |

| | | | |
|---|---|---|---|
| Description: Constructing and interpreting Voronoi Diagrams<br><br>Steps:<br>-project teams construct a 3-point Voronoi diagram on paper by picking a third location on their map and finding the equations of the perpendicular bisectors of each pair<br>-Formative Assessment (Desmos Activity) as assessment for learning | Intervention:<br><br>Further revision material on finding the equation of a perpendicular bisector<br><br>Extension:<br><br>Delaunay Triangulations activity (includes pseudocode and iterative loop) | "Voronoi Diagrams and Food Deserts" (Desmos Activity) | Students continue to find and process data that will be used in their app. The "Food Deserts" aspect of the Desmos activity is linked to social entrepreneurship |
| Description: Shading regions using inequalities and using the Geogebra Voronoi command (practicing the techniques needed to create a "looks like" prototype)<br><br>Steps:<br>-review of inequalities and their graphical meaning<br>-Geogebra tutorial on graphing inequalities<br>-students use the Geogebra Voronoi command to check their Voronoi diagram calculations from the previous activity | Intervention:<br><br>Review of representing one dimensional inequalities on a number line<br><br>Extension:<br><br>Defining regions on graphs using multiple inequalities | Graphing Inequalities (Geogebra Activity) | Students continue to find and process data that will be used in their app. |

| Lesson Plan 2<br>Summary of Key Learning Interactions and Instruction |
|---|
| Lesson 2: Implementing algorithms with Python code |
| Learning Objectives<br>-identify the common types of computational cost functions, and describe their importance to app design<br>-represent algorithms using pseudocode and flow charts<br>-convert algorithms into Python code using a library of commonly used code structures<br>-create Python code that can construct a Voronoi diagram when given an input of a set of map coordinates |
| Formative Assessment: Online Diagnostic Questions multiple-choice quiz on algorithms and Python. Appropriate multiple-choice questions from Diagnosticquestions.com that are directly related to the skills and knowledge they will need to apply to the project will be assigned to students. |

| Summative Assessment: Students create a “works-like” prototype of their app by implementing the algorithms that will construct their Voronoi diagram from the Lesson 1 Summative Assessment in Python code. | | | |
|---|---|---|---|
| Interactions/Activities | Differentiation | Materials/Resources | Field Experiences/Adult Relationships |
| Description: the mathematics of why different algorithms have different computational costs<br><br>Steps:<br>-the handshake problem and its relationship to finding all possible pairs of points on a Voronoi diagram<br>-online interactive lesson on the different types computational cost functions that result from different algorithms<br>-reflection on why this important to their app design project | Intervention:<br><br>Review of logarithms and exponents<br><br>Extension:<br><br>Formal treatment of information theory | “The Mathematics of Big O” | If possible, an industry partner (web developer) could visit during this lesson and open by discussing the importance of efficient code |

| | | | |
|---|---|---|---|
| Description: algorithms and Python code syntax<br><br>Steps:<br>-online interactive lesson on how to write pseudocode<br>-students represent their pseudocode using flowcharts<br>-students are given a set of Python code building blocks so that they can rearrange them to implement their pseudocode<br>-formative Python code quiz | Intervention:<br><br>Partially filled templates for students who are struggling to put the pieces together<br><br>Extension:<br><br>Open-ended design challenges for students who finish early, i.e. "construct an algorithm that does the following" | Pseudocode Tutorial<br><br>Python Diagnostic Questions Quiz | Continued consultation with industry expert |
| Description: constructing Voronoi diagrams from a set of coordinate points using Python code (practicing the techniques needed to create a "works like" prototype)<br><br>Steps:<br>-students run the code in the online tutorial "How to find the nearest hospital with a Voronoi diagram", then modify the data to what will be used in their app | Intervention:<br><br>Struggling students first try to implement with simpler case (the three-point Voronoi diagram from Lesson 1)<br><br>Extension:<br><br>Continue with Delaunay triangulations activity – implement the pseudocode as Python code | How to find the nearest hospital with a Voronoi diagram | Continued consultation with industry expert<br><br>Students continue to find and process data that will be used in their app. |

| Lesson Plan 3<br>Summary of Key Learning Interactions and Instruction |
|---|
| Lesson 3: Mobile app user interface design and culminating event prepraration |
| Learning Objectives<br>-create a simple mobile application user interface using Figma<br>-collect, analyze and present data for the purposes of convincing potential investors of the market potential for a mobile application<br>-create and present "pitch deck" slides to potential investors<br>-create a website with multimedia to promote a mobile application product<br>-manage team project documentation using an appropriate online platform |
| Formative Assessment: Online tutorial in Figma, an app development platform that is free for educational use<br><br>Summative Assessment: Students submit their app user interface design in Figma |

Performance Assessment: Simulated early round venture capital event in which students present a pitch deck to a panel of hypothetical investors. The slides would address each of the criteria in a generic social entrepreneurship project evaluation rubric and include their analysis of search traffic, download and user survey data.

Culminating Assessment: Students exhibit their app demo and website, including a promotional video, FAQ and user technical support documentation, at a trade show-style event. If time and facilities allow, student teams could also design a product logo and 3D print merchandise samples.

| Interactions/Activities | Differentiation | Materials/Resources | Field Experiences/Adult Relationships |
|---|---|---|---|
| Description: designing user interfaces for mobile apps<br><br>Steps:<br>-online interactive lesson on the use of interfaces as abstractions and how they collect user inputs<br>-learning the skills required to create an app user interface in Figma<br>-evaluating the user interfaces of commonly used apps and coming up with criteria for what makes a good user interface | Intervention:<br><br>Pre-made Figma templates<br><br>Extension:<br><br>The ethics of the Gale-Shapley algorithm and the stable matching problem | Introduction to interfaces in computer science<br><br>Intro to Figma tutorial | Students will use the criteria developed to evaluate their own user interfaces |
| Description: preparation for the performance assessment<br><br>Steps:<br>-discussion and analysis of performance assessment rubric<br>-overview of useful metrics and data presentation techniques to use | Intervention:<br><br>Review of basic data presentation techniques and how to create them (bar charts, pie charts, line graphs, etc)<br><br>Extension:<br><br>Explore the web traffic analysis free features at similarweb.com | Performance assessment rubric<br><br>Google Sheets<br><br>Canva or other infographic tool | Students consult with their assigned mentors while developing their pitch deck slides |
| Description: preparation for the culminating assessment<br><br>Steps: | Intervention:<br><br>Pre-made website templates | Google Sites Tutorial<br><br>HTML Tutorials | Students continue to consult with their mentors in the lead |

| | | | |
|---|---|---|---|
| -introduction to free website creation, video editing and logo design tools<br>-choosing an online project documentation platform (introduction to and comparison of Asana and Basecamp) | Extension:<br><br>Commonly used HTML code edits for website customization | | up to the culminating event |