**Upper School Computer Science Change Document**

The following proposed changes to the upper school computer science curriculum serve several purposes. This document provides the rationale for the changes, what we hope to change, the new upper school computer science objectives and the proposed list of courses.

**Purpose of Program Change**

The changes to the computer science program in the upper school are designed to:

1. Provide all Greenhill graduates the opportunity to be introduced to the rapidly growing and high demand field of computer science where most computer science majors in universities today enter the program with little to no prior experience in computer science. Greenhill graduates will all have an introduction to this field by the time they graduate.
2. Help increase gender diversity in a field that is largely represented by males.
3. Create a more challenging, creative and rewarding learning experience for our students helping to elevate computer science as a field of study across the entire community.

These changes are presented as part of a larger vision to see computer science and computational thinking as integral parts of a Greenhill student’s experience across all divisions and multiple disciplines. The following are proposed changes to the upper school program.

**Proposed Changes**

1. The graduation requirement for Greenhill students would change from one trimester of a computer course to one trimester of a qualified computer science course. This change reflects the increasing importance for individuals to understand the basics of how computers are programmed.
2. Students will be expected to complete this requirement by the end of their junior year. Opportunities to complete the requirement during the summer will be made available.
3. Fine arts classes will no longer count toward the computer science graduation credit but will count toward the fine arts credit and general credits for graduation.
4. Fashion Design will be dropped from the computer offerings.
5. All computer science classes in the upper school will follow a 5 day rotation. This will increase available instructional days and decrease the amount of time between the introduction of new concepts and follow-up class sessions where practice (homework) can be checked.
6. While we will not be teaching the AP computer science class, a pathway will be provided for those students who would like to take the AP computer science test. This pathway will require a specific combination of courses.

**Upper School Computer Science Objectives**

Students in the upper school will all have an opportunity to experience an introduction to computer science and pursue more advanced studies if they wish. Students will:

1. Analyze large problems and systematically break them into smaller sub-problems where they can develop original sequential and iterative procedures describing a solution progression.
2. Work in collaborative teams to develop and present solutions to complex problems where solutions to sub-problems can be assembled with solutions by other teams to solve the larger problem.
3. Understand the basics of translating solutions to novel problems into languages that can be understood and executed by computers.
4. Understand how computers work and how they as users are expected to interact with computers and their programs in specific predicted ways. This is the adage: “program or be programmed”.
5. Develop advanced application skills to create new works using existing computer applications.

**Proposed Courses**

The list of proposed courses represent a two year sequence with some courses being offered multiple times each year with others offered once each year or every other year. Five of these courses do not require any prerequisites while seven of them will require prior course work or programming experience.

The courses are divided into three tiers. Tier 1 is the entry level tier and is where most students will start with their computer science classes. Students can select from any of the 5 offerings to satisfy their computer science graduation requirement. Those who do not have prior experience but think they might want to take the AP computer exam should start with Beginning JAVA. Students who do not have a specific computer science interest are encouraged to take Introduction to Computer Science providing them with a varied experience across several different aspects of computer science.

Tier 2 courses require that students have completed one of the tier 1 courses or have equivalent experience as determined by examination and interview. These elective options provide an opportunity for students to explore in depth the creative and challenging aspects of computer science. Most of the options provided here incorporate a blend of disciplines and include elements of engineering and design. Additionally, students will have the opportunity to further develop their language skills through the learning of additional languages.

There is one course listed in tier 3. This course is Advanced Topics in Computer Science and is there to serve our most advanced computer science students. This is one course that can be taken multiple times with the student taking on a different focus each time. This is also where students who want to take the AP computer science test can focus on those elements not covered in the JAVA classes including the Grid World Case Studies.

Four of the classes that are proposed are classes we currently teach. These would receive a redesign to reflect the new class schedule and the changes needed to meet the new standards and outcomes for computer science courses. The other courses are new to the program and would need to be designed from the ground up. Since these course offerings represent a 2 year rotation, there would be a focus on one set of classes for fall 2014 and a second set to be rolled out fall of 2015.

A draft of the course descriptions is provided below as well as a map for how students would progress through the computer science program.

**TIER ONE CLASSES**

**9320 Computer Science 1: An Introduction**

*No Prerequisites*

This course serves as a foundation for entry to more advanced options and focuses on several big ideas related to computer science. Students will be introduced to simple web design skills, basic programming techniques, electronics and robotics as they work on creative solutions to real world problems. The students will often work in collaborative groups as they employ a team approach to creative problem solving using computers. Problems addressed in the course will be addressed with the programming language Python. Additionally, students will work with simple electric circuits and microprocessors to explore how these components work together with programming languages in our daily experiences.

**9040 Computer Science 1: Beginning JAVA Programming**

*No Prerequisites*

This course is designed for those who want to explore the world of computer programming using a high level object oriented language. Students are introduced to the programming environment, basic class structures, sequence, iteration and recursion. They will learn about basic software design practices and apply these principles to create simple programs that address problems related to science and mathematics. Students who take this course have the option of continuing with intermediate JAVA as a way of preparing for the AP Computer Science test.

**9105 Computer Science 1: Introduction to Web Design and Scripting Languages**

*No Prerequisites*

Students are introduced to basic skills and principles of web design using web editors such as DreamWeaver as well as web-based development apps. Students will develop a personal portfolio for their work and link it to several of the web-based applications. Students will also explore the basics of HTML and JAVA scripting languages to provide additional control and function to their sites.

**9205 Computer Science 1: 2D Animation and Game Design**

*No Prerequisites*

This course will introduce the basics of game design using object oriented languages. The basics of playability will be discussed as they explore what makes a game successful. Students will work with the Python programming language to develop their own interactive games. The course will allow students to explore ways they can model their physical world through game development. Through the course, students will have an opportunity to develop their own games as well as work collaboratively as programming teams as they modify existing game programs and develop their own.

**TIER TWO CLASSES**

**9050 Computer Science 2: Intermediate JAVA Programming**

*Prerequisites: Beginning JAVA or equivalent experience*

This course picks up where Beginning JAVA ended and allows students to further develop foundational skills in recursion and iteration as well as develop new skills using arrays and exploring data structures. This class will also introduce control structures and different forms of input. Students will expand on their use of specialized techniques for debugging programs. Students who take this course have the option of continuing with advanced topics as a way of preparing for the AP Computer Science Test.

**9650 Computer Science 2: Advanced Computational Design**

*Prerequisites: Any tier 1 class or prior basic programming knowledge similar to tier 1 class.*

Students will receive a box of electronic and mechanical parts, computer controllers, micro-processors, and a class budget with which to work with as they team up to solve a set of complex challenge problem. Students will be required to design products and develop programs that meet the class challenges. When possible, students will design and print parts needed for their solutions using 3D printers and design software. Students taking this class should have a strong interest in design engineering and experience developing programs in a high level language such as JAVA, PYTHON, or C++.

**Computer Based Solutions through Invention (Future Course)**

*Prerequisites: Any tier 1 class or prior basic programming knowledge similar to tier 1 class.*

Students will work with small computers known as Arduinos and Raspberry Pi to develop an invention that addresses a specific problem of their choice. Students start by configuring the operating system on their Pi then develop programming skills in Python as they explore possible inventions. Students will work through the design process as they brainstorm problems and ways they can solve them using the device they invent. Students will develop a strong foundation in design and computational practices as well as computer hardware configuration and basic electronics.

**Programmable Fabrics (Future Course or incorporated in existing course)**

*Prerequisites: Any tier 1 class or prior basic programming knowledge similar to tier 1 class.*

Students will approach programming from a dynamic and creative perspective as they develop and design with programmable LED embedded fabrics. Students work in teams to develop programs that that meet complex and specific standards related to timing, space, color and sequence. The projects are designed to build skills over the trimester culminating in a final class performance project involving music, choreography and programming. This class blends computer science with design and the arts.

**Programming Round Robin (Future Course)**

*Prerequisites: Intermediate JAVA or comparable knowledge in any other high level programming language.*

This class is designed to allow students who are already familiar with one high-level programming language to build on their existing knowledge and explore how computational solutions can be developed in multiple languages. Students will have the opportunity to explore the development of programs in a variety of languages such as JAVA, C++, Python, Ruby, and Objective C. This class will help computer science students understand the similarities, differences, advantages, and disadvantages of various computer languages.

**Systems Modeling (Future Course)**

*Prerequisites: Any tier 1 class or prior basic programming knowledge similar to tier 1 class.*

From social systems to biological and ecological systems, they can all be studied through interactive representative models. This class will introduce the concept of modeling systems as a way to study and learn about complex systems governing the world around us. Students will use a variety of tools including spreadsheets, STELLA, GIS, and high level programming languages to develop and explore system models.

**TIER 3 CLASSES**

**9070 Computer Science 3: Advanced Topics in Computer Science Tutorial**

*Prerequisites: Tier 2 elective or knowledge equivalent to completion of Intermediate JAVA*

This class allows students to explore a variety of advanced topics related to computer science. Students interested in taking the AP Computer Science test can explore the Grid World Case Study and prepare for the exam. Students can also explore independent designs and programs for developing APPs, designing and programming with Raspberry Pi and Arduinos, robotics, and software design and engineering. Students will be working individually and in small groups as they work within their selected focus. This class can be taken multiple times as each iteration can have a unique focus for the student.