e (9-12) Framework

District Name: KAM LOOPS/ THOMPSON

District Number: 73

Developed by: DigiPen Institute of Technology / R. J. Smith (rjsmith@digipen.edu)

Date Developed: 12/7/2010

School Name: @KOOL

Principal's Name: DEAN CODER

Board/Authority Approval Date:

Board/Authority Signature:

Course Name: DigiPen CS1 Online

Grade Level of Course: GRADE 10

Number of Course Credits: 4

Number of Hours of Instruction: 100

Prerequisite(s):

- Interest in Video Game Design and Programming
- Proficiency in Algebra 1
- Art skills suggested

Special Training, Facilities or Equipment Required:
Computer Science background, or experience in similar area along with training provided by DigiPen Institute of Technology. Facilities should include a computer lab with recent model computer, with reasonable video card and memory. Video projector, DigiPen training software (proprietary game development tools), graphics programs for 2D graphics (preferably PhotoShop CS3 or CS4, or Photoshop Elements). Printed support materials (workbooks, etc) are provided by DigiPen ï students retain their workbooks.

courses that make up the DigiPen Online Video Game
geted at high school students that show interest in video
. The course gives students an introduction to the art and science of interactive video game creation. Students learn the fundamentals of digital artwork and animation, the computer science concepts required to create game software, and the mathematics necessary to add basic movement, collision, and physics behaviors to a simulation. Students participate in synchronous instruction via online video and audio conferencing and use the course websites and workbooks to satisfy the asynchronous time commitment.

Rationale:
British Columbia is rapidly attracting a concentration of video game production companies. Employment and compensation opportunities provided in this industry are among the fastest growing in Canadaôs knowledge-based economy. This program will provide our students with an opportunity to participate in the curriculum developed by an internationally recognized video game university (DigiPen), while gaining an introduction to the field of video game creation and programming. Students will be encouraged to develop cross-curricular knowledge and skills in disciplines such as Mathematics, Science, and Art. The Online Video Game Programming Academy series of courses are designed to give students a serious introduction to the fundamentals of game software and related math topics.

In addition, the fourth course in this series, DigiPen's CS4 Online course, implements a College Board-approved AP Computer Science A curriculum and prepares students for the AP CS-A exam. Successful students may be awarded college credit based on this exam. The CS1 course is the first step in this sequence.

		Time
Unit 1	Module 1 (Game Elements) ï The Cage	40 hours (10 sync/20 async)
Unit 2	Module 2 (Game Elements) ï Base Maze	60 hours (20 sync/40 async)
	Total Hours	90 hours (30 sync/60 async)

Unit/Topic/Module Descriptions:

Unit 1: Module 1 (Game Elements) – The Cage

Module 1 introduces students to the basics of artistic elements such as lines, shapes, and form, and how those elements are used to create imagery in interactive games. A review of math concepts such as trigonometry and distance is included with an introduction to computer math topics. Fundamental computer science concepts such as hardware, software, logic, input/output, and digital signals are introduced. Students create a simple screensaver-like demonstration project in ProjectFUN Editor named ñThe Cage,ò which introduces in-game images, behaviors, and object collision and reflection.

Learning Resources

- Module workbook
- Synchronous live online course meetings
- Distance education website
- Pre-recorded video lessons
- Forum discussions

Learning Outcomes

It is expected the student will be able to:

A. Art
- Name, identify, and use the elements of art composition (line, shape, form, color, etc.) to create computer images
- Create two-dimensional artwork with a paint software package
- Add artwork to a ProjectFUN Editor game project

B. Math
- Demonstrate proficiency with basic trigonometric ratios (sine, cosine, tangent)
- Identify and use trigonometric identity formulas (double angle, half angle, angle summation)
- Use and convert among the decimal, binary, and hexadecimal number systems
- Create logical expressions using the primitive AND, OR, and NOT expressions
- Calculate distances and measures on the ñunit circleò with trigonometry

C. Computer Science
- Compare and contrast ñreal-time interactive simulationò software with motion picture technology
- Explain the difference between analog and digital information and identify examples of each
- Identify the main components of a computer system and explain their purposes
- Explain the concept of a ñgame loopò and recognize its presence in game software
- Create a C program with simple input/output operations
- Create simple math expression in a C program
- Understand how to add behaviors to objects in ProjectFUN Editor
- Implement and explain the role of collisions and reflections in game software

D. Game Industry / Production
- Understand the history of video games and digital interactive entertainment
- Identify trends in game software from the early 1980s
- List the common positions on a game development team
- Understand the purpose and structure of a game design document
- Analyze the ñfun factorò in popular games and identify major genres and categories of games

	%	Summative	%
	20	Problem Sets	20
	10	Programming projects	10
Lab exercises (programming / math)	20	ProjectFUN Editor game: The Cage	20
Total	**50%**	**Total**	**50%**

Unit 2: Module 2 (Game Elements) – Base Maze

In Module 2, students are introduced to the major art design principles such as symmetry, pattern, and unity, and animation design principles such as staging, arcs, timing, and squash and stretch. The math and computer science topics allow students to appreciate simulation as a quantitative process. The concept of the game loop is solidified into three discrete steps: input, handling (logic), and display. Students become more comfortable with C and C++ syntax and begin to write structured programs with functions and early object-oriented programming principles. In the culminating game project, Base Maze, students implement modulus arithmetic and create a text-based game program with a functional game loop.

Learning Resources

- Module workbook
- Synchronous live online course meetings
- Distance education website
- Pre-recorded video lessons
- Forum discussions

Learning Outcomes

It is expected the student will be able to:

A. Art
- Name, identify, and use the art design principles, including symmetry, pattern, and unity
- Name, and identify the principles of animation, including staging, arcs, timing, and squash and stretch, and other fundamentals.
- Understand basic color theory and create a color wheel
- Name primary, secondary, tertiary, and complementary colors
- Use additive and subtractive color schemes, and understand how computer programs use math to implement transparency and translucency

B. Math
- Demonstrate proficiency with fractions and ratios
- Factor a number into its prime factors
- Determine the least common multiple (LCM) and greatest common factor (GCF) of two or more numbers
- Understand the difference between integer and floating-point arithmetic and correctly implement examples of each in the C language
- Implement and use modulus division in the Base Maze game
- Solve and graph linear functions
- Solve and graph non-linear quadratic functions

C. Computer Science
- Start and compile a working C++ program in Microsoft Visual C++
- Identify and list the uses and limitations of different data types in computer software
- Understand that floating-point numbers are represented differently than integers in software and have limited precision
- Create C language expressions using operators and appropriate data types
- Use and display decimal, octal, and hexadecimal numbers in C programs
- Implement logical comparisons to determine collisions in Base Maze
- Use flow control statements in C++ such as:
  - conditional jumps: if, if-else, switch
  - loops: for, while, do-while
- Determine order of evaluation of an expression from the precedence and associativity of operators

e game loop with three discrete steps: input, handling, and display

compare the game to its game design document (GDD)
GDD for a text-based game
- Analyze and consider limitations of different computing platforms

Assessment Procedure

Formative	**%**	**Summative**	**%**
Reading questions	10	Problem Sets	20
Journal entries	10	Art composition	10
Lab exercises (programming / math)	20	Programming projects	10
		Base Maze C++ Text-Based game	20
Total	**40%**	**Total**	**60%**

Overall Course Requirements:

Instructional Components

- Direct instruction (live online)
- Interactive instruction (in-class labs / demos)
- Recorded video-based lessons
- Group-based forum discussions
- Modeling

Student Expectations

- Ability to work cooperatively
- High level of classroom maturity
- Leadership in classroom activities
- Dedication to developing math skills
- Basic knowledge of computer operation
- Independent learning and strong work ethic
- Detail-orientation and strong interest in logic and programming

Learning Resources

- Printed spiral-bound DigiPen Academy workbooks
- DigiPen Distance Education Website / course website
- DigiPen ProjectFUN website ï community resources and forums
- Computer Lab / Home Computer