**Why Games at School?**

Game programming is educational.

Game programming was advanced by Seymour Papert of MIT the originator of Logo which later was commercialised as Microworlds. MIT was also involved in the programming of Lego Mindstorms. The justification for teaching Logo to young children was that the programming skills would transfer to other areas such as mathematics and logical thinking. The main argument against Logo was by Roy Pea who held that a child's "attention is typically so riveted to simply getting a program to work that any appreciation for more general cognitive strategies is lost.".

Programming tools have advanced considerably since Logo, which was conceived in the 70's. Their simple drag and drop interface liberates children from programming syntax and allows them to concentrate on higher order and more transferable tasks, the criticisms of Logo are much less valid for these new programming tools. Games programming can be justified on three grounds, transferable cognitive skills, metacogitive skills and affective benefits:

The idea behind transferable cognitive skills is that students are learning skills in areas such as mathematics and literacy while programming games and that these skills will transfer to the more traditional areas with measurable outcomes.

Metacognitive skills are the self management skills we employ when we are learning.

Affective benefits refer to our attitudes to school, teachers and classrooms. If students enjoy going to school, they will learn better.

**What cognitive skills are likely to transfer?**

- Cartesian coordinates
- negative number
- position, speed, acceleration
- algebraic variables
- relative & absolute value
- estimation
- chance
- a programming language similar to Visual Basic
- metacognitive skills
- structured thinking
- logical thinking inc. Boolean operations
- planning and top-down design
- team planning and development
- ability to represent operations as systems of sequence/selection/iteration.
- re-usability of code and parameterisation

- representational structures and metalanguages
- program proving
- simple physics of gravity, collisions, kinematics
- outcome matrices
- file handling
- efficient data structures and code and possibly network issues, security, database design, graphics, etc
- New as yet unidentified skills for a digital age?

The development process of gaming is laden with maths and science concepts, offering many possibilities for technology integration. A teacher may choose to teach game development as a process requiring collaboration of a team and coordination of multiple resources. The unit could focus primarily on the mathematical concepts or the laws and practice or science or both!

As with any computer programming activity, the lessons could emphasize the importance of reading and writing clearly defined instructions and code.

Game Maker is an ideal tool for teaching programming since the instructions can be defined with the drag-and-drop interface or handwritten in the scripting window. As a game is developed in Game Maker, the software is writing instructions to the computer in the form of conditional statements with interchangeable variables.

For younger students, you may choose to provide the statements and variables to them, or you may extend the lessons to define the concept so they are able to generate the statements and variables independently.

**How we can use Game Maker to support learning in other subjects**

***Science:*** Laws of physics, testing trial and error, and the Scientific Method can all be learned as students apply various Events and Actions to the objects in their game. As you review the list of Game Maker actions and events, consider the possibilities for *Move, Collision, Solid, Friction, Speed, Create, Gravity, Random, Bounce, Transform, Blend, etc.*

***Maths:*** Game rooms in any gaming software are laid out on a grid. The grid size and orientation can be adjusted by the developer, but all objects are placed on the grid at a given x, y coordinate (x, y, z in a 3D game). This offers a great opportunity to integrate maths and geometry concepts as well. In addition to the x, y axis of the game room, each individual object has its own x, y axis to consider.

**Through Game Design:** The design process focuses on the content of the game. Will the game be arcade-style, an adventure, or a competition? Will it be educational or strictly entertaining? After students learn the tools and concepts of game development, they will easily be able to creatively design games with more purpose and strategy. You may choose to define a content area for them to use, or give them more broad criteria and focus on the development skills. My personal approach to teaching gaming is in three phases, from teacher-guided, to teacher-facilitated, to student-guided. These lessons begins with a tutorial, introducing students to the game software, the main concepts, and relevant terminology and gradually leads to a self guided approach.

**Games as a Framework for Learning**
The power of games programming is its power to motivate. Game programming requires mathematical and logical skills. Good games also have a storyline, graphics and music. Good games are created by a team of creators working co-operatively.

Game programming could be a framework within which team skills, music, art, drama, maths, history, geography or almost anything could be learned. The important feature of the game is its power to motivate. Motivation leads to learning.

“The computer is a medium of human expression and if it has not yet had its Shakespeares, its Michelangelos or its Einsteins, it will. …. We have scarcely begun to grasp its human and social implications.”

*Computer Criticism vs. Technocentric Thinking By Seymour Papert*

**Games and Enterprise**
We can think about games in relation to the 5 key concepts

**Languages**: what are the unique features of games and the ways in which they communicate with and engage players? How do we need to reconceptualise some of our previous assumptions about media communication to include concepts such as immersion, feedback, rewarding the player and so on?

**Representations**: how are people, places, ideas and emotions portrayed in games and what sorts of social and cultural assumptions underlie those portrayals? Eg. gender representations - why are males and females portrayed in certain ways? What are the consequences of this? What should be our response? (note that this is very much up for debate and discussion - assumptions should not be made).

**Technologies**: what should students understand about the technology related to games production, distribution, access and play? How are social and economic factors related to the development of games technologies? How can we think critically about these technologies?

**Audiences**: How should we think about games players? What is the relationship between games and players - where does the power lie? To what extent should games be censored? Theories of audiences suggest that players are not vulnerable dupes, but active and intelligent users - how can we test the validity of this?

**Institutions**: how can we get students thinking critically about the institutions that enable and constrain the production and distribution and regulation of video games - production companies (roles in games production, profit motive, the role of corporations etc), regulation organisations and distribution points (the aims and objectives of retailers, arcades etc - and the tactics they use for marketing and branding).