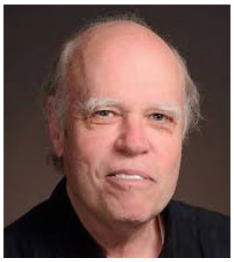
Full STEAM Ahead

Science, Technology, Engineering, Art, and Math from NCLab More Than Coding

Issue 7, June 2017

Video Games Are Good For Learning

by James P. Gee, (Regents Professor and Fulton Presidential Chair of Literature Studies at the Arizona State University. Member of the National Academy of Education)



What makes video games good for learning has little to do directly with the fact that they are games. Furthermore, video games that are most interesting for learning are not just any video games. Different types of games can have different effects. Puzzle games like Tetris and Bejeweled may very well exercise pattern recognition capacities; Trivial Pursuit games may well make learning facts fun. But these are not, in my view, the sorts of video games which are most interesting in regard to learning.

The video games that I think are most interesting for learning, are digital simulations of worlds that are "played" in the sense that a player has a surrogate or surrogates through which the player can act within and on the simulation and that have "win states" (reachable goals that the player has discovered or formed through his or her surrogate). By the way, in augmented reality games, a person can be playing a virtual role (e.g., urban planner, toxic spill specialist, detective) in a rule system that is designed to play out partly in a virtual world and partly in the real world.

Games are simulations that are focused on problem solving and which players can enter and not just inspect from the outside. Since humans learn best from experiences that are focused on well-mentored problem solving, games are an ideal setting for learning. And, by the way, games have teachers: the game designer and the mentoring and teaching built right into the game design itself.

Video games don't just carry the potential to replicate a sophisticated scientific way of thinking. They actually externalize the way in which the human mind works and thinks in a better fashion than any other technology we have. We humans think by simulating and testing possible courses of action in our minds before we act. Video games are just an external simulation and testing platform, directly supplementing our minds with the potential to better prepare us to act in the world (Imagine that Presidential candidates had to win at least one game of Civilization before they could run for office).

Writing, digital computers, and networks each allow us to externalize some functions of the mind. Though they are not commonly thought of in these terms, video games are a new technology in this same line. They are a new tool with which to think about the mind and through which we can externalize some of its functions. Better yet, they allow for collaboration and collective intelligence as forms of intelligence than can far transcend our individual knowledge.

Video games are good for learning because, among other reasons, they have the following features:

- They can create an embodied empathy for a complex system built as a world focused on particular types of problems.
- They are action-and-goal-directed preparations for, and simulations of, choices and actions.
- They involve distributed intelligence via the creation of smart tools like the "mods" in World of Warcraft that tell players how to coordinate their play with others.
- They create opportunities for cross-functional affiliation where smart teams combine people with different specific deep skills but where each team member can coordinate with each other's skill set effectively.
- They allow meaning to be situated in (associated with) images, actions, goals, and experiences, not just in verbal definitions cut off from the world.
- They can be open-ended, allowing for goals and projects that meld one's personal goals, social goals as collaborators, and the game designer's goals (the teacher's goals).

None of this is to say that video games do these good things all by themselves. It all depends on how they are used and what sorts of wider learning systems (activities and relationships) they are made a part of. None of these reasons why video games are good for learning stems primarily from a game's great 3-D graphics or the mere fact that it is a game in the general sense of "game". The cutting edge of games and learning is not in video game technology—although great graphics are wonderful and technical improvements are important. The cutting edge is realizing the potential of games for learning by building good games into good learning systems in and out of classrooms and by building the good learning principles in good games into learning in and out of school whether or not a video game is present.

Reference to full article: "Why Are Video Games Good For Learning?" by J.P. Gee, http://www.academiccolab.org/resources/documents/MacArthur.pdf

NClab in Public Libraries in Nevada

by Tammy Westergard, Assistant Administrator of Library / Library Development Services, Nevada State Library, Archives and Public Records

Every day changes and advancements in technology, accelerates what it takes to live day to day. Everyday learning in the 21st century means level up or be left behind. This is the stark reality. Fortunately, within Nevada's Everyman's University System (also known as our public libraries) NCLab is there to explain, demonstrate and inspire. Through their exceptionally designed online learning system, anchored with sound teaching methods and learning design, Nevadans are developing coding skills and abilities essential in today's workplace.

Nevadans are learning at the library how to combine building and problem-solving. This talent will allow them to work, earn a living wage and find purpose in creating tomorrow's digital tools and solutions necessary to address challenges we can't even see today.

J.D. Salinger said, "You can't stop a teacher when they want to do something. They just do it." This fundamentally describes the NCLab teaching team. And their success in Nevada libraries is the prime example. People of many, many different ages and circumstances are learning a vital 21st century literacy: coding. And putting the skill to work for them. The Nevada State Library is honored to have engaged NCLab on behalf of every Nevadan who seeks opportunity through education and needs the library to help them get there. Nevada will continue to deploy the NCLab tool in the coming years and provide exciting, challenging hands-on computer science education that is the NCLab experience. This model is only the beginning for all Americans. Our libraries are the great equalizer to make us job-ready on day one.

When Computers Wore Skirts

Today, the word "computer" usually refers to a powerful electronic machine. But this was not always the case. Before the use of electronic computers, and even before the use of mechanical computers, the computers were human. Among the earliest records of organized human computing was the computation of the trajectory of the Halley's Comet by Alexis Claude Clairaut and his two colleagues in France in the 18th century. In the late 19th century, thirty five human computers were employed to create the British Nautical Almanac of data that was used for navigation at sea. Many human computers were employed to compile Western European mathematical tables, mainly those for trigonometry and logarithms. The computers were often educated middle class women. In many cases, they stayed at home and received and send back large packets of calculations by post.

Human computers played an integral role in the World War II war effort in the United States, and in particular in the Manhattan Project. Most of them were women, since male workforce was depleted due to the draft. Naturally, all six people responsible for setting up problems on the ENIAC, who were drafted from a corps of human computers, were women: Kay McNulty, Betty Snyder, Marlyn Wescoff, Ruth Lichterman, Betty Jean Jennings, and Fran Bilas. Following World War II, the National Advisory Committee for Aeronautics (NACA) used human computers in flight research. The role of human computers at NASA was to calculate anything from how many rockets were needed to make a plane airborne to what kind of rocket propellants were needed to propel a spacecraft. A remarkable group of African American women worked at what would become NASA's Langley Research Center in Virginia. The legacy of these women, and that of the other early human computers, is literally written in the stars.

Tank, Octopus, And Submarine: 3D Models From Texas

NCLab is delighted to share some great 3D models from McKinney High School in Texas. Computer Science teacher Theresa Horvath writes:

"After attending the NCLab workshop at the 2016 CSTA Annual Conference last year, I returned to school with a goal of having students use the NCLab programs before the end of the school year. My 1st year computer science students learn to code using Scratch, Jeroo and Java. I was eager for them to also use the NCLab courses to learn Python and create tangible products produced by their own code. I submitted a grant request for a 3D printer and our local McKinney Education Foundation awarded me $2500.

The 3D printer arrived in January. Students have been working on projects during this 1st-year computer science course. They began by completing Karel and Turtle Tina lessons, including numerous Turtle Tina 3D designs. Then they used the 3D Modeling lessons and online example projects as a guide in order to write code for their own 3D objects. The students created various designs including a football field, hockey rink, basketball hoop, toilet paper roll and key ring.

After the AP exam in early May, my AP Computer Science A students began using the NCLab software also. They explored a little with Turtle Tina lessons, but most jumped right into the Python 3D Modeling. Numerous students wrote code for amazing projects such as an Octopus Planter, Military Tank and Submarine."

How did the students build such complex models? Each one begins by defining objects as simple shapes. These shapes are then combined, subtracted, rotated, moved, and tweaked in many ways. Can you see how the planter started as a sphere-shaped shell? Or how the tank includes several repeated shapes? The fuselage of the submarine began as a set of points used to create a Bezier curve.

Here they are: Andrew Cope's "Military Tank", Udani Satarasinghe's "Octopus Planter", and Michael Hamilton's "Submarine". You can run the code yourself! Visit NCLab's 3D Gallery, and follow the link below each image to open the files. Many thanks to Mrs. Horvath's class for sharing these 3D models with NCLab.

Michael Hamilton's submarine looks simple, but includes complex curves and precise details.

Udani's planter in action!

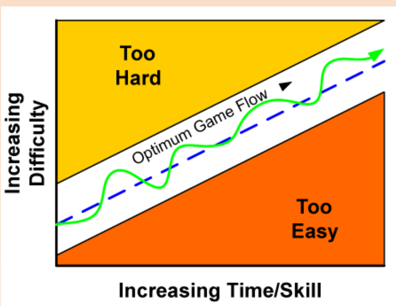
A close up of Andrew's tank.

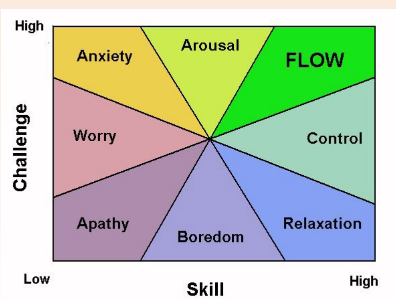
Video Games and the Art Of Failing Creatively

by Kevin Fredericks, NCLab Game Director

At NCLab, I have been helping develop software that gives learners interesting choices and precise feedback. I think this software improves the way people learn, and a lot of this has to do with the way people fail.

In the institutions of education, one often hears about the importance of "critical thinking". Critical thinking is the science of failure. How did this machine fail? How did this argument fail? How did this computer program fail? We have accepted the base assumption that learners should think critically, but what do we understand about failure? The truth is, we can only understand failure by experiencing it, and to fail we have to try to learn something new. We need to take a risk.





If you want to improve your ability to learn, you need to improve your ability to fail. Successes come naturally, but a useful failure takes work. I might sound like a motivational poster, but I urge you to take chance on an important challenge. There is no magic bullet for learning how to code or use computers. There is no four-week bootcamp that will ever teach you how to think critically if you haven't already learned to fail creatively. With the right help, learners everywhere will look forward to their next failure as an opportunity, and then we might each learn what it means to succeed

Nevada Ready 21 Digital Summits – Wrapping Up Year 1

Digital classrooms, 1:1, student-centered learning, a rich array of apps, extensions, and tools, and of course, NCLab … it's time to reflect on Year One of Nevada Ready 21. This Nevada Department of Education program put CTL Chromebooks in the hands of 20,000 students and their teachers, along with great software and training opportunities. In the 1:1 classroom, every student in a classroom is connected, not just during computer lab time, but any time. Their work is saved to the cloud, and there are many opportunities to collaborate.

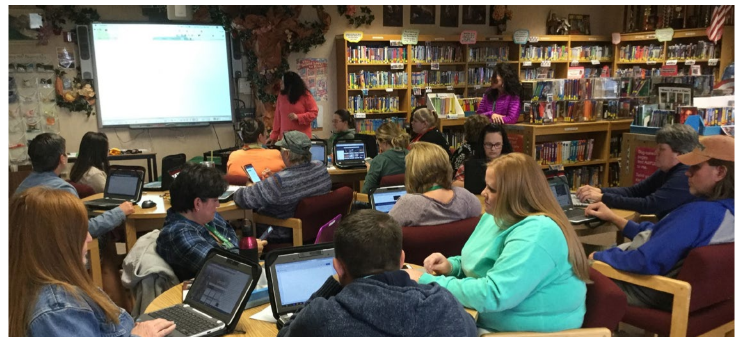
At the April 2017 Digital Summits in Elko, Carson City, and Henderson, NR21 educators converged to share their experiences and learn something new. The air hummed with "aha" moments and the sense of being part of a breakthrough in education. The winner for teachers? Real-time formative assessment.

Formative assessment is assessing progress while students are learning. With the 1:1 classroom, teachers can see how their students are doing in real time, which is possible when everyone is working on their own Internet-connected device. Teachers can then help those who need it, when they need it. Students become active learners, self-correcting and revising as they go. This is exactly how NCLab works too!

Plus, the digital classroom is fun.

Many thanks to CTL and the Nevada DOE for the Nevada Ready 21 Digital Summit events. Thanks also to our hosts at Spring Creek, Carson, and Bob Miller Middle Schools. Last but not least, the program is a success due to all the educators involved – teachers, coaches, professional development instructors, administrators, and edtech vendors.

On to Year Two!



Teachers from Elko, Battle Mountain, and Ely took a different approach to their NE Nevada Ready 21 Digital Summit. With Blanca Duarte's guidance, they created an "unconference" to share expertise and solve problems. Even the smallest, most isolated schools are rocketing forward with the power of the Internet and 1:1.

Sheila Bunch and Josh Billings strike a pose at the Nevada Ready 21 NW Digital Summit. Josh includes NCLab courses in his STEM curriculum at Carson Middle School. STEM is all about problem solving. Right now, Josh's class is using technology to examine the impact of this winter's record snowfall and pending snow melt.

From left to right: Amy Al-Khalisi (CTL), Sheila Bunch (NCLab), Clifton Roozeboom (Pocketlab), and Wayne Lawson (Bob Miller Middle School) at the Southern Nevada Ready 21 Digital Summit. Discussions ranged from using virtual reality as a learning experience, to working with data from wireless sensors. This is 21st century education!

Building Karel the Robot: Invaluable Learning Experience For Students

by Dave Riske, Networking Technologies Instructor, Western Nevada College (WNC), Carson City

NCLab's Karel the Robot is a perfect tool to teach students the fundamental ideas of computational logic and structured design, uncluttered by specific programming language requirements. The premise behind the Karel language is to use a simple syntax structure with simple logic as a starting point and then slowly building the complexity of both syntax and logic. Though students were grasping the concepts presented in the activities in Karel, there was still a tangible component missing. WNC already owned several versions of the Lego Mindstorm NXT and EV3 robotic kits.

Therefore I decided to try to externalize Karel the Robot from the screen to the tabletop. Initial requirements were simple:

- The robot must look like Karel the Robot.
- The robot must respond in a manner consistent with the virtual robot in NCLab.
- Students should be able to use a command set similar to the commands available in NCLab to control the robot to solve mazes.
- Building mazes for the robot must be easy and relatively inexpensive.

Physical Robot and Playing Field

The robot and the playing field must mimic the environment available in NCLab. The initial plan for the playing field was to use a 4×8 sheet of plywood as a base. This was gridded out in 8×8 squares yielding a playing field of 6 by 12 squares, which could expand easily with the addition of another board to 12×12. This would require careful consideration of which puzzles and mazes to externalize as many of the mazes are larger than the available physical space. Ultimately, the playing field is a maze. We are currently investigating using foam core, cut in an interlocking method, that will allow us to build an interior wall that can be 'seen' by the robot and provide the means to have the robot respond to the environment of the maze. The foam core will also be used to provide a border wall around the playing field.

Initial searches yielded a robot design that looked similar to Karel the Robot and was modifiable with respect to available components within the Lego kits already owned by WNC. This initial build was also modifiable dependent on needs that became evident as the various programs were designed to control the robot. Further work and development will be done on the physical aspects of the robot to simplify the build to one type of standard kit and its associated expansion kit.

During the rebuild of the physical robot to include a single Lego Kit the robot base size grew by approximately 1.5 inches when the EV3 treads system was used. This made the initial design of the field inadequate to hold the robot. Resizing the squares to 12 inches will further shrink the effective playing field.

Command Structure

The commands available to control Karel the Robot are Go, Left, Right, Get and Put. These are initially used in sequences and are later combined within decision structures (if-elif-else statement) and loops (repeat, while, for) to enhance both the logic and the syntax of the program as well as provide discussion for the three primary components of programming structure. Our initial intent is to make Karel the Robot as independent as possible with respect to movement, not relying on cues from the maze like line following. Get and Put commands would rely on control of the robot's screen to display output with respect to number of objects.

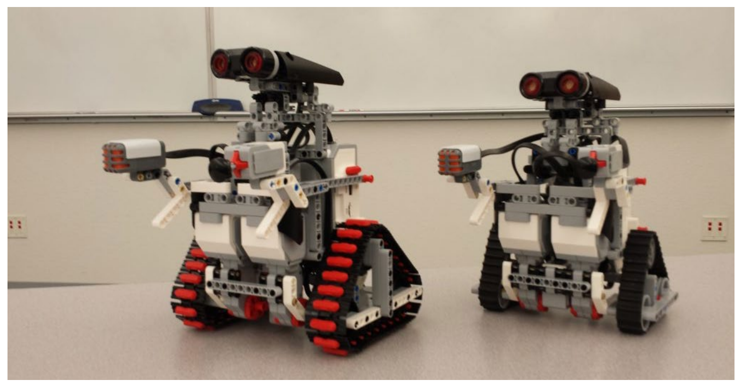
Challenges

With the initial trials with respect to turning the robot, we discovered very quickly that there is an inherent lack of accuracy in the Lego parts. Variations in the motors, power applied to the motors, the treads, the maze and even the battery power all made for variations on the turn. Installing the gyroscope sensor on the robot allowed for measurement of the error, +/- 4 degrees on a 90-degree turn. It was evident that a self-correcting program was needed that would measure and seek the desired turn. Searches revealed programs for selecting the direction of the turn (left or right) and for controlling (and measuring) the amount of turn. Combining these two programs yielded a program that allows for precise turns with self-correction. The correction, however did allow the discovery of the inherent error in the gyroscope measurement. Measurement of the turn, when the gyroscope reads 90 degrees is currently at 90.8 degrees. The hope is that this error is consistent and, therefore, able to be programmed out of the turn.

Future Work

Currently students are working on refinement of the turning function, a true 12 inch Go Function and providing the Karel the Robot with the ability to recognize different colored objects which will be used to distinguish object within the mazes. They are also working on the physical aspect of the maze as additional materials are found which can provide for building.

Students are embracing the challenges involved in turning the robot 'toy' into a functioning tool. Even if the challenges prove to be insurmountable due to the nature of the Lego system as a toy, the experience for the students in confronting those challenges and discovering the limitations is proving to be invaluable as a learning process.



From left to right: Karel the Robot model 2.0 (single Lego set) and model 1.0 (two Lego sets).

Congratulations to our Latest Karel Black Belt

Anyone who has completed the Karel course will tell you that it takes perseverance to go all the way to the end. Matthew Manwill, a 7th grade student at Pau Wa Lu Middle School in Gardnerville, Nevada, recently earned the highest course honor: a Black Belt of Second Degree. His computer programming teacher, Claudia Bertolone-Smith, thinks he has amazing skills – and so do we! Let's hear from the master coder himself:

"My name is Matthew Manwill, and 4 months ago, I started working on NCLab. Since then, I've completed Karel by passing challenges and helping others do the same along the way. Currently, I am working on Turtle Tina, which teaches Python by creating designs that can be 3D modeled or printed. NCLab is a educational program, and I have learned much from it. The code templates were really helpful as I worked in NCLab. The biggest challenge was programming to make random decisions. Hopefully, this achievement follows me into my future, because, so far, I'm set on becoming a programmer. I am very interested in working in the field of robotics. Developing increasingly sophisticated robots will be beneficial to mankind in the future. I would like to recommend NCLab to others as well."

Thanks, Matthew! We are honored to be part of your journey.