TEAP Journal

www.teap-online.org | Winter, 2009

PA Standards
for Science, Technology and Engineering

Richland Wins
National Championship

Designing for the Real World

Thinking
Like a Programmer

Making Stuff in America

Value Added of
Technology Education

Volume 57 Number 4

PR SRT STD
U.S. POSTAGE
PAID
Permit No. 138
Washington, PA

3750 Woodcrest Court, Erie, PA 16506-5506
Please keep your TEAP membership current.
Change Service Requested

Thinking Like A Programmer

Mr. Brian Herrig, Canonsburg Middle School

As the subjects covered in Technology Education become increasingly complicated, teachers are compelled to provide students with the foundation needed to transition from one level to another. We must strive to shorten the learning curve by giving our students the intermediate steps they need to climb the ladder of knowledge without having to extend beyond the skills they possess. In building the foundation, it is important to remember that "as human beings, we are the kinds of creatures who can learn in many ways" (Gardner, 2000).

The central theme of Gardner's *Theory of Multiple Intelligences,* it is certainly applicable to the concept of progressive content in Technology Education. Our students have grown up digitally. They interact with various technologies multiple times per day and may have trouble imagining a world without the technological artifacts that have become such an integral part of their daily lives (Prensky, 2001). If we do not embrace our students' desire to interact with what they are learning, we will be selling their potential short and setting the stage for an uphill battle as we force them to be passive rather than active recipients of knowledge.

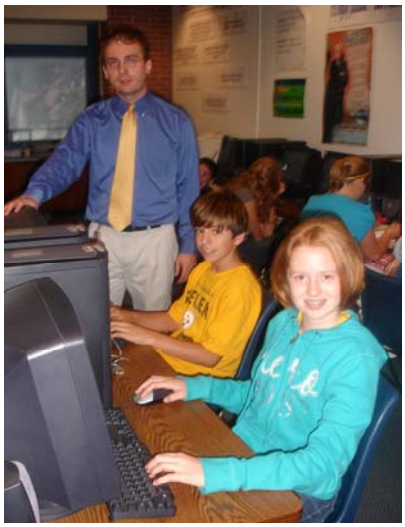
Video Game Design

Recently, in an effort to incorporate more 21st century skills and keep the classroom focus firmly on students of the digital age, a new component was added to the seventh grade curriculum at Canonsburg Middle School. With strong support from administration, a video game design unit was added to the seventh grade Technology Applications class to prepare students for material they will encounter in eighth grade Automation Technologies.

A common first impression of teaching seventh graders about video games is that it must be a fun unit for the students as well as the teacher. While it is true that many students enjoy video game design, it is far from being (pardon the pun) "all fun and games." The unit is less about video games than the name implies. The goal does not make the students video game experts, but rather helps students to understand the basic mindset of computer programming: any action that needs to occur needs to be programmed. Each student learns to "think like a programmer."

Reverse Engineering

After being introduced to terminology, the students get a playable version of a basic video game and told to "play the game". Reverse engineering is done with pages of direct questions about the game. To complete the "play-by-play", students must carefully observe game actions. They must resist the urge to simply play but must answer questions related to what they see and hear. In addition to the encouraging careful observation, the reverse engineering serves as a guide they will use to verify the functionality of their own game as they work to recreate it in the next steps.



Mr. Herrig and his game programmers

Game Maker

As the students progress, they use Game Maker; a video game design software, to recreate a basic game. All students start by creating the same basic game so they all know what the final product should look like and function. This ensures that all students acquire foundational skills. First, they create the basic game elements; sprites, images, and sounds. Next is the fun part, telling the elements what to do. The elements must be told how to move, react to game objects, and play sounds. Every action in the game must be programmed. Finally, a "room" is created to give the game elements a place in which to exist and the game to be played.

Extending Knowledge

As they program their games, the students are told what needs to be done and specific steps to make it happen. As they advance, students are given fewer "how to" details. This transition helps students to strengthen their foundational base and extend their knowledge to complete the project. Throughout the teacher-created instructions, the Pennsylvania Academic Standards for Science and Technology are reinforced. Students' resource files are saved to a common network server drive. All students can access network files, but only the teacher can modify or delete.

Questioning

An unexpected benefit of having students become critical observers is that they become skilled at asking, and answering, questions. Initially, many questions are broad: "why doesn't my game work?". Teacher replies help them understand that only specific questions are addressed and the form of the inquiries changes dramatically. Questions become: "why doesn't my score increase when I click on this object?" The answer: "because you didn't tell it to." By guiding students to ask specific questions, they are guided to specific answers. Game Maker's interface makes troubleshooting quite easy if you have identified the specific action that needs corrected. By recognizing that game points should increase when an object is clicked, the student can check mouse click events and determine what they have programmed it to do and make corrections. The concept that all actions must be programmed is reinforced repeatedly.

Troubleshooting

A game follows programmed directions as given; right or wrong. A student grasped this concept when his game did not operate as expected. Without re-programming, he re-tested and got the same undesirable results. After he tried to troubleshoot the problem independently, I asked why the game was functioning the way that it was. His accurate answer: "because I told it to, now I need to figure out where." It was clear that he understood that all the game actions were under his control. He was thinking like a programmer.

Summary

Having students think like programmers helps prepare them for challenges that require logical, sequential thought. Game design challenges them to achieve at all levels of Bloom's Taxonomy of Cognitive Objectives to develop foundational knowledge: terminology, reverse engineering, analyzing actions and reactions, questioning techniques, and troubleshooting.

The scope of the content possible in Technology Education can vary far more than concepts covered in other subjects. In the same way that we help students to grasp algebra concepts before we expect them to master calculus, it is essential that we pro-

See Thinking, page 17

PLTW & TSA Partner

Project Lead the Way (PLTW) announced an Engineering Alliance partnership with TSA and SkillsUSA to provide PLTW students with the opportunity to participate in innovative activities in STEM education outside of the classroom. The alliance will offer STEM-related co-curricular activities for PLTW students to participate in, including online competitions and leadership development activities designed to support teachers and excite students about technology, innovation and engineering. Anticipated contests include electronic gaming, solving a design problem with STEM applications, CAD design, and building an engineering model. Leadership activities will feature teamwork, communication and time management.

"We feel our partnership is a natural one," said Rosanne White, executive director of TSA. SkillsUSA and PLTW share our goal of preparing students for the high tech workforce. The Engineering Alliance is an exciting step for America's future workforce."

To learn more, visit :
www.pltw.org/ or *www.tsaweb.org/*

Thinking Like a Programmer

Continued from page 9

vide students with a solid foundation if we expect them to have mastery level understanding of the high level content. Helping students to be active learners increases their desire to learn and content retention. The high level critical thinking skills Technology Education develops will serve our students well as they progress through our classes and beyond.

Game Maker Versions

Game Maker is an excellent resource. Currently, a free version is available with limited functionality. An upgraded fee version has no restrictions. Site licenses are available, making it possible for schools to equip a lab for less than a few seats of other software. Game Maker's free version can be downloaded at home so students can continue and enrich their studies. To download Game Maker or for more information, visit *www.yoyogames.com*.

References

Gardner, H. (2000). Can technology exploit our many ways of knowing? In D.T. Gordon (Ed.), *The digital classroom: How technology is changing the way we teach and learn* (pp. 32-35). Cambridge, MA: Harvard Education Letter.

Prensky, M. (2001, Oct.) Digital natives, digital immigrants. *On the Horizon, 9*(5), 1-6.