GLOBALORIA™ INVENT▸BUILD▸SHARE™

The Globaloria Learning Process for Computational Thinking

Participate: Use the Globaloria Social Learning Network to work in teams, learn to solve programming problems and share computational knowledge publicly. Learn to collaborate onsite with classmates and educators, as well as virtually with students in other schools, and professional game makers and programmers.

Publish: Learn how to present and publish designs, code, and games online.

Play: Play to discover what makes a great educational game. Learn about game mechanics, simulations, genres, and design principles. Get inspired!

Program: Write the code for your game in Flash ActionScript. Learn to program, test, and get help from experts using tutorials and virtual network for communication.

Plan: Decide who the audience is and what your game is going to do. Research learning topics and learn your game content. Organize your ideas in a written plan. Keep adding to the plan as your research and design develops.

Prototype: Draw and videotape your game concept and test your prototype with users. Learn to use Flash to create an interactive demo that shows how the game will look.

Student-Made Health Games

Student-Made Civics Games

Student-Made Science Games

10 Design Principles for Teaching Computational Inventiveness the Globaloria Way:

1. Learn by designing digital artifacts, functional, representational and educational games.
2. Master complex subjects by constructing pedagogical games for others.
3. Work on open-ended computational design tasks that focus on topics of choice.
4. Learn in a transparent studio setting (online/onsite) where work is built and shared.
5. Spend significant time on task (100 hours), in year-long project-based learning.
6. Have ample opportunities for social expression and discussion about game projects.
7. Have ample time for reflection about games, wikis, blogs, and representations.
8. Use programming and computational design tools as primary constructs of learning.
9. Use multiple modalities in the learning process (text, imagery, video, simulation).
10. Learn alongside educators (co-learning) and from experts (just-in-time learning).

Key Constructionist Computational Thinking Abilities (Learning Outcomes):

1. The ability to do invention, progression, completion of an original project; capability to program an educational game, wiki or simulation.
2. The ability to manage project-based learning in Web 2.0 learning environments; capability to process complex project and team management (via programmable wiki systems).
3. The ability to produce original computational media; produce programming code, publish and distribute interactive, purposeful, digital media in social learning networks.

Free Trial: ***www.Globaloria.com/TestDrive***

Contact ***info@globaloria.com*** *to bring Globaloria to your school!*