

Improving Educational Game Design  
Methods: A Rubric to Assess the Engagement  
and Educational Value of Educational Games

Ken Hoff  
University of Colorado at Boulder  
`kendall.hoff@colorado.edu`

November 18, 2013

## **Abstract**

Existing educational games lack the combination of education and engagement. The objective is to research, synthesize, and test a more effective educational design method. The procedure is to research current literature and existing effective educational games, synthesize an educational game design rubric based off of this research, then apply this method to known educational games. Amazon's Mechanical Turk will be used to gather large amounts of data about the consistency of this rubric, after which the data will be collated and analyzed.

# Contents

0.1	Problem . . . . .	3
0.2	Thesis Statement/Solution . . . . .	4
0.3	Approach . . . . .	4
0.4	Research . . . . .	4
0.4.1	Edutainment vs Educational Games . . . . .	4
0.4.2	Tangential Learning . . . . .	6
0.4.3	Flow . . . . .	7
0.4.4	Play . . . . .	8
0.5	Rubric . . . . .	9
0.5.1	Location of Game Encyclopedia . . . . .	10
0.5.2	Content of Game Encyclopedia . . . . .	10
0.5.3	Amount of Referential Material . . . . .	11
0.5.4	Popularity of Referential Material . . . . .	11
0.5.5	Rewards for knowing Referential Material . . . . .	12
0.5.6	Adaptive Difficulty . . . . .	12
0.5.7	Contextual Tutorials . . . . .	13
0.5.8	Low game resource penalty for failure . . . . .	14
0.5.9	Low "reset time" for failure (time to return to failure point) . . . . .	14
0.5.10	High checkpoint frequency . . . . .	15
0.5.11	Freedom of exploration . . . . .	15
0.5.12	Iterative feedback . . . . .	16
0.5.13	Unorthodox problem solving . . . . .	16
0.6	Games . . . . .	17
0.6.1	Darfur is Dying . . . . .	18
0.6.2	Light Bot . . . . .	19
0.6.3	Pandemic 2 . . . . .	21
0.6.4	Oregon Trail . . . . .	23

0.6.5	Number Munchers . . . . .	25
0.6.6	BotLogic . . . . .	27
0.6.7	Math Baseball . . . . .	29
0.6.8	Notpron . . . . .	30
0.6.9	Lemmings . . . . .	32
0.6.10	The Incredible Machine . . . . .	33
0.7	Survey . . . . .	35
0.7.1	Basic Survey . . . . .	35
0.7.2	Rubric . . . . .	37
0.7.3	darfur Quiz . . . . .	43
0.7.4	oregon Quiz . . . . .	46
0.7.5	munchers Quiz . . . . .	48
0.7.6	Lightbot Quiz . . . . .	51
0.8	Conclusion . . . . .	54

# List of Figures

1	Darfur is Dying's title screen . . . . .	19
2	A screenshot from Darfur Is Dying's village management section	19
3	Light Bot's main menu . . . . .	20
4	A screenshot from one of Light Bot's levels . . . . .	21
5	The title screen of Pandemic 2 . . . . .	22
6	A screenshot from Pandemic 2's evolution view . . . . .	23
7	A screenshot from the opening sequence of Oregon Trail . . .	24
8	A screenshot from the gameplay of Oregon Trail . . . . .	25
9	The Number Munchers opening screen . . . . .	26
10	A screenshot from the gameplay of Number Munchers . . . . .	27
11	BotLogic's logo . . . . .	28
12	One of the levels from BotLogic . . . . .	28
13	Math Baseball's title screen . . . . .	29
14	A screenshot from the gameplay of Math Baseball . . . . .	30
15	The Notpron opening screen . . . . .	31
16	The first level of Notpron . . . . .	31
17	The Lemmings box art . . . . .	32
18	One of the levels from Lemmings . . . . .	33
19	The title screen from The Incredible Machine . . . . .	34
20	One of the levels from The Incredible Machine . . . . .	34

## 0.1 Problem

Educational games lack the combination of education and engagement. The standard is cheesy games, where the content being taught is either uninteresting or unrelated to the type of play. A good example is Math Blaster; players answer simple arithmetic questions to fill up their energy, which they then use to shoot space debris. These games are usually just homework problems,

either included as an unrelated component inside the game, or interspersed between gameplay segments in a quiz format.

## **0.2 Thesis Statement/Solution**

I'm going to synthesize an education game design method by creating a list of attributes that educational games should have a high degree of in order to be effective and engaging. The list is not a set of standards, but a set of guidelines; it is possible (though unlikely) for a game to contain all of the attributes and still be ineffective, or for a game to have none of the attributes and be very effective. It is simply meant to infer that game designs that contain these elements are more likely to be effective and engaging educational games.

## **0.3 Approach**

I'll begin by examining current educational game design methods, through research of published literature. I'll then synthesize my own educational game design method, a list of attributes found in effective and engaging educational games, influenced heavily by my research and findings. Afterwards, I'll attempt to apply my method to existing educational and semi-educational games, and see if the most effective educational games correlate to my design method. In contrast, ineffective educational games won't correlate at all to my design method.

## **0.4 Research**

### **0.4.1 Edutainment vs Educational Games**

#### **Bloom's Taxonomy**

Bloom's Taxonomy is one of the most widely cited classifications of learning. It provides milestones for three different domains of learning: Psychomotor, Affective, and Cognitive.

The first domain, Psychomotor, deals with the ability to perform intricate physical tasks, like hammering a nail or operating a complex machine or instrument. Games are currently very good at teaching concepts within the

Psychomotor domain. The best examples are ones where the game provides a near-simulation to the task being performed in real life. Music games like Guitar Hero and Rock Band improve coordination and motor control while playing an instrument. Simulation driving games like Forza and Gran Turismo provide players with continuous feedback to help them learn the correct response to stimulus given the car they are driving, to help them improve their driving skills. Other examples include any game that improves motor control; Kinect's Dance Central improves player's bodily movements, and even first person shooter games like Halo or Call Of Duty improve player's hand-eye coordination.

The Affective domain is one that is far harder for games to address. It deals with learning empathy, emotion, and includes things that are commonly learned through interaction with people and not necessarily through formal education. Games commonly expose players to this in two ways. The first is through game narrative, where players are exposed to characters, situations, and choices where they observe and learn about the human condition. Games with extreme character development (Mass Effect, The Walking Dead) typically explore human choices in great detail. There are also extremely minimalistic games that communicate powerful empathic ideas without a large amount of gameplay or graphics, or conversation (Thomas was alone, Journey), as well as completely narrative-based, choose-your-own-adventure Twine games that address extremely empathic concepts (Howling dogs). The second is through putting players in situations with other players where they will need empathy and skills from the Affective domain in order to solve their problems. All multiplayer games facilitate this to some degree, but puzzle games (Portal, LittleBigPlanet), and MMO's (WoW, EVE Online) generally encourage players to communicate, negotiate, and work together to achieve a common goal.

When we think of educational games, we commonly think of the Cognitive domain. The Cognitive domain deals with knowledge of facts, associations, and mechanics, and typically consists of the material that educators attempt to teach in schools. This is also the domain that games attempt to address the most; being able to communicate these concepts effectively would reduce the amount of material educators would need to teach, allowing them to focus on more complex concepts. Because games in the Psychomotor domain are well traversed, and games in the Affective domain are far more nebulous as to measuring their educational value, we will be focusing exclusively on the Cognitive domain.

image

## **Educational Games, Not Edutainment**

It's also important for us to differentiate between edutainment and educational games. For this paper, we're going to define edutainment as the simple gamification of a task. The purpose of edutainment is to increase the player's skill at something by getting them to play a game; the player mimics or actually does the skills within the game, and the game rewards them and keeps them engaged in order to continue increasing their skill. These kinds of games, while effective, are ineffective at teaching complex concepts and systems. For that purpose, we use educational games; they have the more rewarding task of teaching complicated content, but it is far more difficult to do so, as they can't utilize the skill and drill method of education. We're going to focus primarily on educational games in this paper.

### **0.4.2 Tangential Learning**

In their paper on Serious Games and Learning, Breuer and Bente (2010) outline a method called Tangential Learning. Originally presented as part of the Extra Credits video series online (Floyd and Portnow, 2008), the presenter gives us a method for which we can encourage significant learning using games. The theory is that, given the proper incentive, players will engage in a form of self-education separate from the game environment.

They base this on the principle that we're able to recall and understand material that we're more interested or passionate about, rather than material that is uninteresting or boring to us. The same principle applies to video games; if we are interested in the context of an educational game, we're far more likely to remember that instead of the content being taught as part of it. The example Extra Credits uses is geography; it's easy for some players to draw the entire map of Azeroth from memory, but they're unable to properly identify US states.

The most prominent field that this could be applied to is history; as most games are set in the past, such as Ancient Rome or World War II, it would be easy for players to learn more about history on their own simply because they were playing the game in the appropriate context. There are other contexts that players can be placed in, in order to educate about other topics. Players in music games can self-educate on music theory and different musical



genres, and players in space simulations might self-educate on spaceflight and astronomy.

However, there are numerous other ways players can be educated apart from being placed in the context of the educational goal. An easy-to-implement way would be to include facts as part of the loading screens in the game; instead of having game tips or strategies, designers could include interesting bits of external information to help players retain information. Also, designers can include subtle references to real-life objects within their games. Extra Credits gives the example of Sephiroth from Final Fantasy, or the Excalibur and the Masamune. By naming characters or objects after real-life characters or objects, designers can encourage players to go out and seek the origin of the character or object; it also helps to have once object in a group be something that's easily recognizable, so that players immediately know that other objects in the group are referential to the real world.

Another excellent way to integrate tangential learning into games is by including an in-game encyclopedia, where players don't even have to leave the game to explore more about the topic they're interested in. Usually, these encyclopedias include information related to playing the game; the stats of an object, or the rules of the game. However, some include information that isn't related to playing the game, like the history or origin of an object or character. A notable example of this is the game Civilization, which includes the Civlopedia, a reference to every unit, building, and wonder in the game, as well as other information related to it. An easy way for games to integrate this would be just including Wikipedia links to every object in the game that has real world roots.

### 0.4.3 Flow

Dondlinger (2007) explains why it's extremely important to give players the perception of free will. In educational games, we don't want to give the players no direction as to where to go within the game; we risk players missing the educational goals, and worse, becoming frustrated with the lack of progress. Conversely, we don't want to overly guide the players; too much hand-holding results in the game stifling creativity, as well as not allowing players to really learn by doing. It's important for us to give them the illusion of having an open world, where almost any actions are possible, in order to allow players to experiment and learn by doing.

In parallel to the illusion of free will is the concept of flow.' In games, we

want to the player to experience just the right difficulty; games that are too hard will be frustrating, and games that are too easy will be boring. The same is true of education and educational games; material that is far beyond the comprehension of the learner won't be retained, but material that the learner has already covered sufficiently will be boring. We need to maintain a level of difficulty to our game that Csikszentmihalyi (1997) calls flow.' When players are in a state of flow,' the game provides to them a clear set of inputs and outputs; material that they are somewhat familiar with, but also obscure enough to encourage to players to try several options in order to solve the problem.

Following the concept of flow,' we can extrapolate the concept of adaptive difficulty.' It's logical to assume that games that continually adapt their difficulty level to the players will be more effective in retaining the player's interest and educating the player than games that have a fixed difficulty curve. However, not every game can implement adaptive difficulty effectively; games need the players to continuously attempt problems and give feedback on how well they're doing to ascertain how well the difficulty matches up to the player.

#### **0.4.4 Play**

Similar to the illusion that players have a seemingly limitless game world to interact with is the concept of play.' Paras (2005) defines a world where play can happen simply as a world with a series of constraining rules. We don't want the rules to be too constraining, because then we inhibit creativity and discourage alternative solutions to problems, but at the same time, we want the rules to be constraining enough to guide the player towards the solution and mimic the rules of the real world.

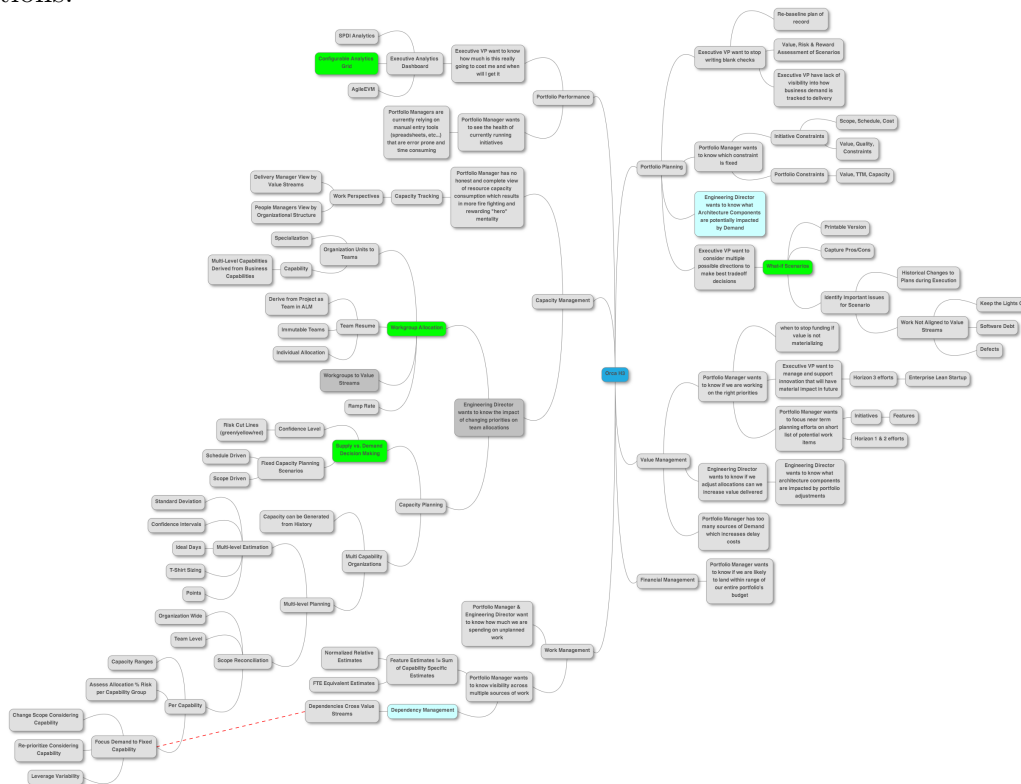
The way the game guides players around these rules is extremely important. If the game gives the player copious amounts of tutorials and guidance as to what they can or can't do in the game, the player won't learn them as well. This isn't to say that the player should be encouraged to break the game rules, but instead to encourage the player to explore the world's rules; they won't understand what will and won't work unless they encounter it on their own and are enabled to learn from it.

The game also shouldn't penalize the player for exploring the boundaries of the game world, as it would discourage them from trying new things and discovering additional boundaries. However, we still need to encour-

age players to solve problems within the game world without breaking any boundaries, just to ensure that they understand the world that they’re playing in; that’s why rewarding the players for not breaking any rules works much better than penalizing the player for breaking rules.

## 0.5 Rubric

What follows is a list of properties that make up the educational game design, created by researching and synthesizing properties of existing educational games. Games are not required to have all or any of these properties, but the hope is that games with more of these properties will be considered more effective and engaging educational games. Certain sections of the rubric are left blank, but properties of the section can be extrapolated from adjacent sections.



### **0.5.1 Location of Game Encyclopedia**

Some educational games include a game encyclopedia as part of the game, either internally or externally as a game manual or wiki. Having such an encyclopedia available within the game greatly increases the chance of players using it for self-directed and self-motivated learning, as part of tangential learning.

- 1 Game contains no encyclopedia of game content
- 2 Game information is located online, in a non-central location
- 3 Game has an outside manual (or wiki) of game content
- 4 Game doesn't have an in-game encyclopedia, but points to a central location elsewhere
- 5 Game has an in-game encyclopedia of game content

### **0.5.2 Content of Game Encyclopedia**

Having a game encyclopedia as part of the game encourages self-motivated players to seek out additional help. If the encyclopedia includes more than just game mechanics as part of its content, players may be more encouraged to self-educate themselves about those topics.

- 1 Game contains no encyclopedia or Encyclopedia contains no content
- 2 Encyclopedia contains content only related to game mechanics
- 3 Encyclopedia contains information about game mechanics, and also limited historical/factual information
- 4 Encyclopedia contains content related to game mechanics and historical/factual information

**5** Encyclopedia contains content related to game mechanics and historical/factual information and outside links or references

### **0.5.3 Amount of Referential Material**

By having many objects or events in the game that are references to real-life objects or events, players may recognize certain objects or events and self-motivate themselves to learn more about those objects or events. More objects like that in the game means more chances for a player to recognize a real-life object or event.

- 1** No game objects or events are references to real-life objects or events
- 2** At least one event or object is a reference to a real-world event or object
- 3** Several non-connected events or objects are references to a real-world event or object
- 4** At least one group of objects or events are references to real-life objects or events
- 5** Numerous groups of objects or events are references to real-life objects or events

### **0.5.4 Popularity of Referential Material**

Having real-life objects or events in the game means that some players will recognize them and attempt to self-educate themselves about those objects or events. In addition, having a large amount of those references be popular means that more players will recognize the more popular objects/events, and by association, education about the lesser-known objects/events will be the result of self-motivated education about the popular items.

- 1** Existing references are extremely obscure
- 2** Some of the references are popular

- 3 About half of the references are popular
- 4 Many of the references are popular
- 5 Most or all of the references are popular

### **0.5.5 Rewards for knowing Referential Material**

If games allow players to use their knowledge of the referential material in a positive manner within the game, it reinforces the desire for the player to tangentially learn about all the referential material in the game. This is in contrast to traditional methods, where the content that the player is trying to be educated about serves as a barrier of entry to the later levels of the game, or the game penalizes players that don't know the content; with this, players can still play the entirety of the game, but are incentivized to do better by learning the referential material.

- 1 Knowing the referential material is purely irrelevant; doesn't affect the gameplay at all
- 2 Knowing the referential material is a little useful; only affects the gameplay a small amount
- 3 Knowing the referential material is somewhat useful; moderately affects the player's choices during gameplay
- 4 Knowing the referential material is very useful; usually affects the player's choices
- 5 Knowing the referential material always significantly affects gameplay

### **0.5.6 Adaptive Difficulty**

In order to incorporate flow as much as possible, the game must use a form of Adaptive Difficulty; the game tries to match the difficulty of the obstacles to the skill level of the player. This can be done in a rudimentary way by allowing the player to select the difficulty themselves, but is most effective

when it recognizes how well a player is doing on a certain puzzle and adjusting the difficulty automatically. This ensures that the player isn't trivially challenged, but at the same time isn't pushed outside the bounds of their ability.

- 1 Game only has one difficulty
- 2 Game has several difficulties, but players can only select difficulty at beginning of game
- 3 Game has several difficulties, and players can change difficulty mid-game
- 4 Game has several difficulties, and prompts the player to increase or decrease the difficulty as needed
- 5 Game has several difficulties, and automatically adjusts the difficulty of the game as needed

### **0.5.7 Contextual Tutorials**

In keeping with flow, players need to be challenged at their level of skill within every game. This means that players shouldn't be hindered by challenges that are too easy for them, but likewise shouldn't feel completely lost on difficult challenges. The method of introducing these challenges is equally important; players that are forced to go through numerous, easy introductory levels will quickly become bored, but players that skip through tutorials at later stages or forget about valuable information learned early in the game quickly become frustrated. It's important to give the player only information that they need; similar to the concept of adaptive difficulty, the game gives the player helpful hints or suggestions when they are stuck, or when the player doesn't intuitively grasp a new mechanic the first time. Similarly, there needs to be tutorials for just about every concept in the game, to cater to players that might have learning 'gaps' in their history.

- 1 Tutorials aren't given in the game
- 2 Tutorials are given at the beginning of the game

- 3 Tutorials are given every few levels/sections of the game
- 4 Tutorials are given at the beginning of every level/section
- 5 Tutorials are offered continuously

### **0.5.8 Low game resource penalty for failure**

In order for the game to encourage exploration and play, the player needs to feel free to fail. This means that if the player does fail, there is a minimal amount of game resources (e.g. lives, tries, health points) consumed each time the player fails. With this, the player is comfortable trying different solutions to problems within the game without much penalty.

- 1 Game resource penalty is extreme (e.g. restart the game)
- 2 Game resource penalty is large (e.g. 1/3 lives, 10 health points / 100)
- 3 Game resource penalty is moderate
- 4 Game resource penalty is small
- 5 Game resource penalty is nonexistent (e.g. unlimited lives)

### **0.5.9 Low "reset time" for failure (time to return to failure point)**

In order for the game to encourage exploration and play, the player needs to feel free to fail. This means that if the player does fail, there is a minimal amount of external resources (e.g. time, frustration) consumed each time the player fails. With this, the player is comfortable trying different solutions to problems within the game without much penalty.

- 1 Reset time is very long (e.g. reloading level takes a long time)
- 2 Reset time is long (e.g. greater than 10 seconds)



- 3 Reset time is moderate (e.g. up to 10 seconds)
- 4 Reset time is short (e.g. a few seconds)
- 5 Reset time is very short (e.g. near instant)

### **0.5.10 High checkpoint frequency**

It's important to have players feel free to fail within games. Having a high frequency of checkpoints reduces both the amount of time/frustration that it takes for the player to return to the point of a failure, as well as in-game resources.

- 1 Zero checkpoints
- 2 Checkpoints are few and far between (e.g. levels are the only places to restart)
- 3 Checkpoints are moderate
- 4 Checkpoints are numerous (e.g. players can restart at the beginning of each puzzle)
- 5 Checkpoints are frequent (e.g. players can restart part of the way through puzzles)

### **0.5.11 Freedom of exploration**

If players are to be encouraged to explore, the nature of the game must be nonlinear and allow for players to progress in varying sections of the game at their own pace. That isn't saying that they shouldn't have a finite goal to work towards, but the methods or paths with which they work towards that goal should be optional and variable.

- 1 Players are placed in a strictly linear world or lesson progression
- 2 Players are allowed just a few large-scale choices in their game world

- 3 Players have the option to make choices about the direction of their progression in the world, but it is largely linear
- 4 Players can choose from many choices within the game world to explore, including lessons
- 5 Players are free to choose the direction they want, both educationally and within the game world; allowed to jump between parallel lessons

### 0.5.12 Iterative feedback

In the case of games with a high degree of free-form exploration, it's easy to see that some players might get lost and become frustrated. In order to maintain flow, these players must receive some kind of feedback that they're on the right track, and not just wandering aimlessly through the game world or puzzles. It's extremely helpful for players to receive feedback at every step they take, even during the completion of puzzles; that way, they can be sure that they're progressing, and easily see the steps ahead of them, enabling flow.

- 1 Game gives no feedback other than high-level progression through the game
- 2 Game gives feedback after each level
- 3 Game gives feedback at various points through a level, after a series of puzzles
- 4 Game gives feedback after each puzzle
- 5 Game constantly gives feedback (e.g. during a puzzle)

### 0.5.13 Unorthodox problem solving

Another important element of allowing players to explore and play is to give them the option to solve problems using unorthodox solutions; solutions that the game didn't anticipate, but still validate the puzzle's end condition.

Sometimes, the player may be able to figure out how to circumvent certain sections of a puzzle; this should still be considered valid, but it's hoped that the circumvention wasn't because of faulty game logic. This allows players to try unusual solutions to problems, and help them understand the system far better than if they had just been taught the correct solution.

- 1** There is only one way to solve any given problem, with one given progression that is valid as a solution
- 2** Some problems within the game may be solved more than one way
- 3** Multiple solutions are available for each problem, but players are limited to using one of those solutions
- 4** Players can solve each problem via any solution, so long as they do not circumvent it
- 5** Players can solve a problem any way they like, or even circumvent the problem, and be given full (or bonus) points

## **0.6 Games**

### 0.6.1 Darfur is Dying

**URL** <http://www.darfurisdying.com/>

**Description** Darfur is Dying, developed by mtvU, is a flash-based game released in April of 2006. The game consists of two main sections. In the first section, players must select a member of a Darfuri refugee; the family consists of a male, female, and several children. Once the player has chosen, the player must guide the refugee to a well using only compass and distance directions, while attempting to avoid Janjaweed militia patrols in trucks. If the player is caught, the game describes the fate of the refugee, and the player is prompted to select another refugee. Once the player has made it to the well, the player must return through the same section to their encampment, but lose water during their journey. Once they have returned to the encampment, the player enters a top-down strategy-like game simulation; they can use the water they have retrieved to grow crops and keep the encampment in good condition. However, if they run out of water, they will need to return to the first section and make the well run again. The goal of the game is to keep the encampment alive for 7 days. In addition, the community is constantly under threat from attacks from the militia; if the militia attacks, the encampment is lost, and the player must start again. The player can prevent attacks from the militia by participating in various viral and advocacy campaign tactics, such as inviting their friends to play, posting on social media, or writing to government officials.

**Educational Content** The educational content of Darfur is Dying is centered almost entirely around awareness. Through playing the game, players learn about the nation of Darfur, including its history, wars, environment, climate, and people. The players are forced to be aware of the troubles that plight Darfuri residents and families, such as militia attacks. Players are encouraged to read the backstories of every man, woman, and child, as well as stories associated with locations within the village that expose various events that have happened, such as being unable to fend off sickness without medical aid, or when a militia recently stormed the village and murdered numerous people.



Figure 1: Darfur is Dying's title screen



Figure 2: A screenshot from Darfur Is Dying's village management section

## 0.6.2 Light Bot

URL <http://armorgames.com/play/2205/light-bot>

**Description** Light Bot, made by Danny Yaroslavski, is a programming and robotics puzzle game. It was originally a flash game, but has since been ported to iOS and Android. Players assume control of a robot on a grid of

varying sizes and orientations. Each grid square can also have a height. The robot has the ability to move forward, turn left or right, jump up one level or down one level, and turn a square "on." The robot also has the ability to call "functions," where the robot can execute sequences of events and repeat functions several times or indefinitely. The goal of the robot is to navigate to all of the blue squares and turn them "on" to a yellow state. The robot can do this in any order and using any sequence they like, so long as it fits within the provided instruction spaces. There are 40 levels to the game, ranging from the simple to extremely difficult.

**Educational Content** Light Bot's educational purpose focuses on teaching programming at a very simple level. Players learn that the bot will follow sets of instructions. Initially, these instructions will be very simple (e.g. forward, turn, blink), but the player will realize quickly that the bot will follow the instructions explicitly, even if they do not solve the puzzle. This teaches players that computers are very powerful but very simple machines, and will do exactly what they are directed to do, even if it's not what the programmer intends to do. The game also teaches the concepts of functions and loops; players can "call" predefined functions numerous times, as well as have a function call itself to loop the function indefinitely. These programming concepts, while simple, are a wonderful introduction to programming for students.

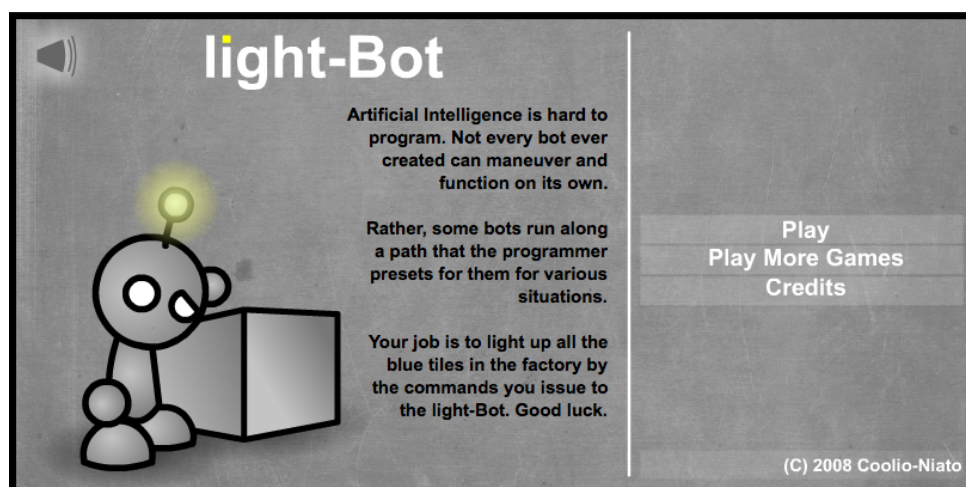


Figure 3: Light Bot's main menu

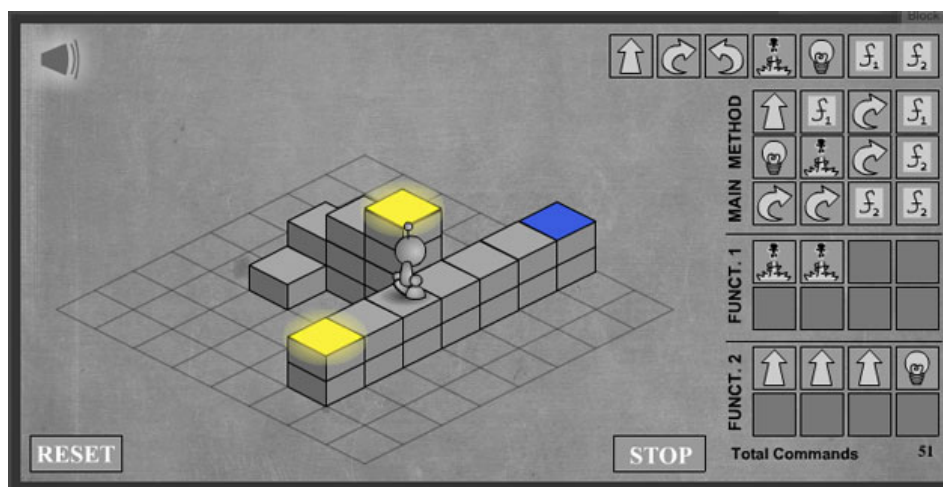


Figure 4: A screenshot from one of Light Bot's levels

### 0.6.3 Pandemic 2

**URL** <http://www.crazymonkeygames.com/Pandemic-2.html#game>

**Description** Pandemic 2 is a flash-based strategy game involving infectious diseases, viruses, and bacteria. The player is in charge of designing and mutating an infectious organism, which infects the world population. The objective is to have the infection spread to and kill every human being on the planet, rendering the human race extinct. The virus starts out as being only mildly visible, lethal, and infectious, and can be mutated to more effective versions through "upgrades," received as more humans are infected and die. To combat the spread of the infection, world nations begin to close their borders, set up quarantines, and close off trade routes, cutting off the transmission of the infection to their nation and making it more difficult or nearly impossible for the disease to spread. The player typically alters between the disease upgrade screen and the world monitoring screen, which includes notable headlines and the statuses of the nations. Global high scores are given to players that successfully eliminate the human race in the shortest amount of time.

**Educational Content** Pandemic has two limited educational aspects to it. The first is the notion of learning about infectious diseases and organ-

isms. While there isn't much science within the game behind mutating an organism to be more deadly, there are plenty of terms and game mechanics that the player can familiarize themselves with, such as organism's resistance to humidity, or how airborne diseases differ from waterborne. There's also an element of strategizing, risk-taking, and planning ahead associated with playing the game; if players run into a problem with a certain method, they may be able to solve the problem a different way, or circumvent the problem on another playthrough.



Figure 5: The title screen of Pandemic 2



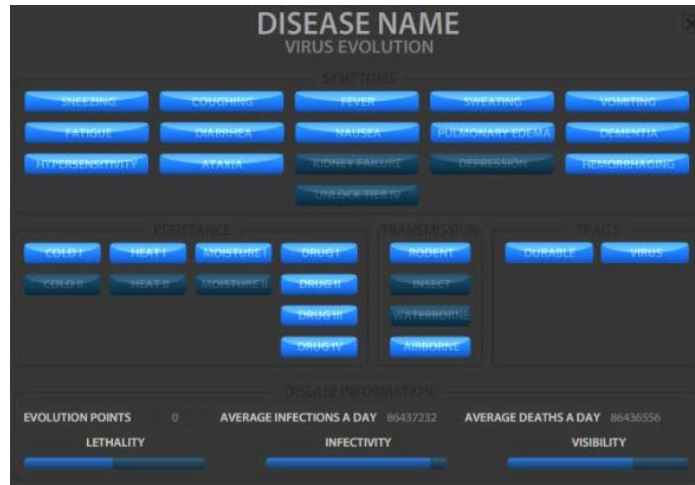


Figure 6: A screenshot from Pandemic 2’s evolution view

#### 0.6.4 Oregon Trail

**URL** <http://www.virtualapple.org/oregontraildisk.html>

**Description** The Oregon Trail is a series of games detailing the experience of a pioneers traveling the Oregon Trail, a wagon route from the Missouri river to Oregon in 1848. The original game, developed for the Apple II in December of 1971, turned out to be a extremely well received by middle and high school students. In the game, the player first selects an identity which determines how much starting money they have. They then select supplies to buy for their journey; wagons, spare parts, oxen, food, and other supplies. Once they embark on their journey, time passes and food is automatically consumed. Occasionally, players have the chance to hunt, where they can play a simple top-down 2D game to shoot animals that provide food to their party. In addition, numerous events will happen that the player needs to make descisions for, such as a party member falling ill, a wagon part breaking or oxen becoming injured, or needing to cross a river. The player’s objective is to travel the entire Oregon Trail with using the minimum amount of resources; the player receives more points at the end for having living party members, items in inventory, and number of dollars, as well as receiving a multiplier if they started the game with less cash.

**Educational Content** Oregon Trail's educational value comes in two forms. The first is the most apparent one; though not explicitly sitting players down and teaching them, the game educates students on life in the 19th century, as well as the hardships and trials endured by explorers of the early Oregon Trail. It teaches them about the kinds of materials that were used in everyday life, like wagons and oxen, as well as the diseases that commonly plagued explorers (diarrhea, dysentery), and what day-to-day activities were like on the trails, such as maintaining the wagons, crossing rivers, and hunting for food. The other educational aspect that Oregon Trail focuses on is planning and risk management; though not explicitly teaching players how to assess the risk of various actions, players who properly evaluate their initial inventory options as well as the options they have during events on the trail will end up doing better than players who don't.



Figure 7: A screenshot from the opening sequence of Oregon Trail



Figure 8: A screenshot from the gameplay of Oregon Trail

### 0.6.5 Number Munchers

**URL** [http://wallofgame.com/free-online-games/arcade/988/Number\\_Munchers.html](http://wallofgame.com/free-online-games/arcade/988/Number_Munchers.html)

**Description** Number Munchers is part of a "Munchers" series that began as a strictly educational game around teaching children basic arithmetic and other numerical concepts. The series then expanded to teach words and other knowledge. The player first selects the educational content that they'd like to play, including grade level. Administrative controls are available for parents and teachers to restrict what kind of content is available; for example, restricting higher-grade students to their level, instead of allowing them to play at trivial levels. Once in the game, the player assumes control of a green 'Muncher,' which can move up, down, left or right. The game board is a grid, with each space containing a number that is a solution to the level they're playing (e.g. equal to 12, multiples of 3, divisible by 5). The player moves onto the appropriate space, and presses the space key to 'eat' the number. If the number matches the level criteria, the player gets points; if it does not, the player loses one of three lives. The player also has to deal with 'troggles,' which cross the grid slowly at random intervals. If the player occupies the same space as a troggles, the player loses a life and is reset to a different grid space. The level ends once all the appropriate numbers are

eaten, and the player receives a bonus for time.

**Educational Content** The educational content of Number Munchers is extremely straightforward. Players will learn arithmetic, primes, and other numerical concepts while playing this game. Players who learn the concepts will be able to identify the correct solutions faster, and consequently achieve a higher score while playing this game.



Figure 9: The Number Munchers opening screen

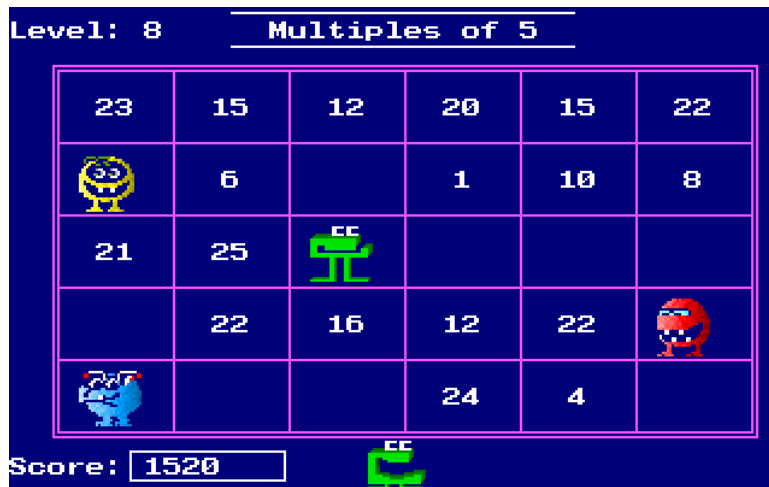


Figure 10: A screenshot from the gameplay of Number Munchers

### 0.6.6 BotLogic

**URL** <http://botlogic.us/>

**Description** BotLogic is a HTML5/Javascript developed by Dolphin Micro, a web development consulting firm located in Colorado. In the game, the players assume control of a robot whose task is to return home. The game takes place on a grid, with the robot located on one space, the home on another space some distance away, and a number of obstacles in between. The player can direct the robot up, down, left, or right, and does so by queueing up commands before running the program. The player can wait for the program to run, then add more commands before running the rest of the program, which allows players to incrementally build their program. However, the robot has a limited amount of energy, which limits the number of moves the player can take. Later in the game, more obstacles and powerups are introduced, such as electric fences, buttons, and recharging stations. The game contains 20 levels.

**Educational Content** Botlogic teaches players about simple programming concepts, namely passing a sequence of instructions to a robot and watching them run. However, the game doesn't include any functional or object-oriented programming concepts, and is only slightly abstracted away

from the player having direct directional control of the robot.



Figure 11: BotLogic's logo

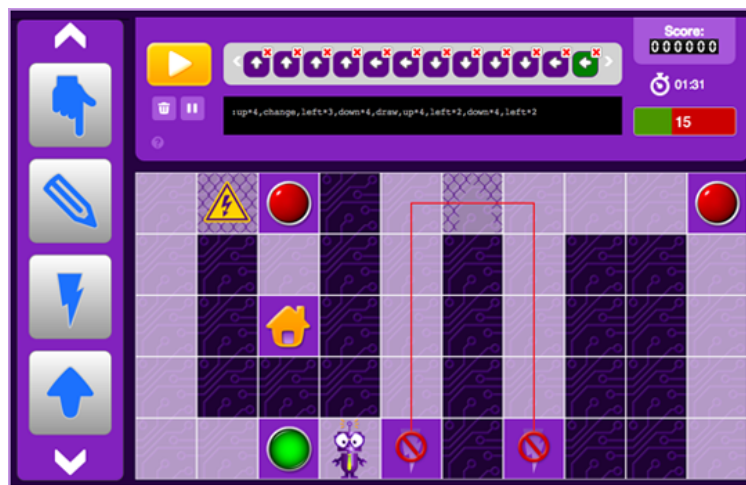


Figure 12: One of the levels from BotLogic

## 0.6.7 Math Baseball

**URL** <http://www.funbrain.com/math/index.html>

**Description** Math Baseball is an extremely simple HTML game. Initially, the player selects what type of arithmetic they'd like to do, as well as the grade level. Then, they assume the role of a baseball player at bat. For each throw, the players are given unlimited time to answer one arithmetic question. If they get the question right, the player earns a randomly selected single, double, or triple and gets a runner on base, as well as advancing any other runners. The player's score is determined by how many runs they get in. If they don't get the question right, then they receive a strike. After three strikes, the player is "out," and the game ends.

**Educational Content** The education in "Math Baseball" is very straightforward. Players can learn addition, subtraction, multiplication, and division, with the player's knowledge of the topic reflected in their higher score of the game.

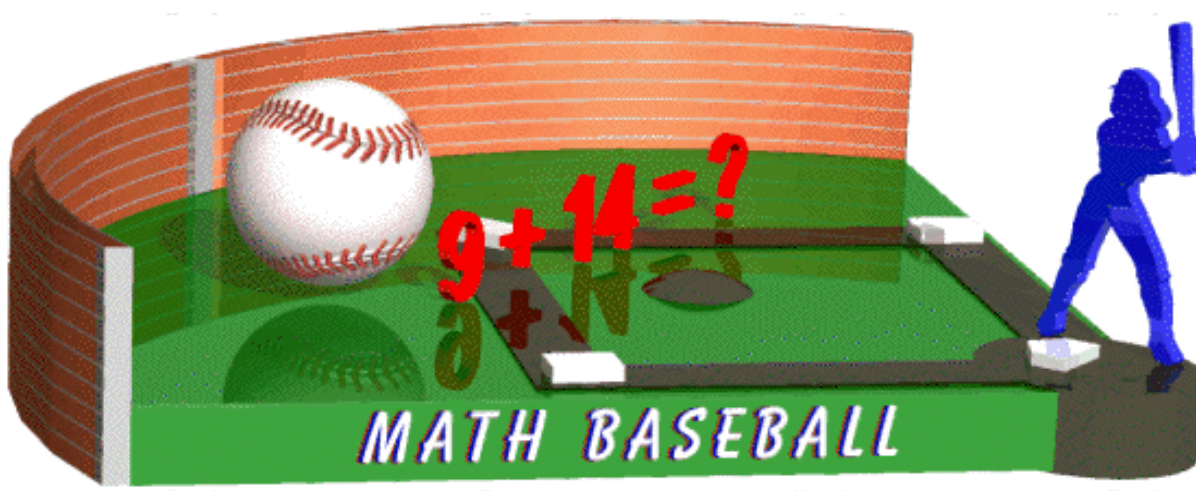


Figure 13: Math Baseball's title screen



Figure 14: A screenshot from the gameplay of Math Baseball

## 0.6.8 Notpron

**URL** <http://notpron.org/notpron/>

**Description** Notpron is an ARG (alternate-reality game) for the browser. The player begins Level 1 with an image of a house with a partially open door in front, as well as some slightly opaque text that says "Enter the door." The player needs to click the door (not the image, but the door itself) to advance to Level 2. In Level 2, a finger points to the address bar, where the player can replace "level2.htm" with "level3.htm" to advance to the next level. In Level 3, the player must change "false" in the URL to "true" to advance to the next level. The game continues like this, adding in new elements each level. There are a total of 140 levels, and only 31 people have completed all 140 levels, out of about 16 million players.

**Educational Content** From the Notpron site:

"[Players who finish the game] have persisted with a broad range of complex ways of thinking, while maintaining focus and dedication over a long period. [Their] detective skills have been tested to the limits, yet the smallest hint proved sufficient to solve the most complicated tasks. Furthermore, competence in the following areas have been displayed: Sound editing, Graphic editing,



Musical understanding, Insight into HTML programming, Rapid learning of new programs, Efficient online research techniques, [and] Insight into the complex workings of a computer.”



Figure 15: The Notpron opening screen



Figure 16: The first level of Notpron

### 0.6.9 Lemmings

**URL** <http://www.elizium.nu/scripts/lemmings/>

**Description** Lemmings, a PC game, was originally developed in 1991. The game plays from a 2D side-scrolling perspective. The player directs "lemmings," small, humanoid creatures that are reminiscent of the mammal in their behavior. They operate almost entirely on their own, walking in one direction until they run into something, then reversing direction. They begin by dropping out of the entrance door, and successfully exit the level through another door in the level. However, the lemmings are very susceptible to dying; falling too far without a parachute will kill them, and numerous obstacles litter the courses, such as spike pits and smashers. The game contains four difficulty levels, with roughly 20 levels per difficulty level.

**Educational Content** Lemmings teaches players problem-solving, multitasking, and resource management. Because a fixed number of lemmings are required to finish the level, players must learn the proper methods for getting around obstacles by using the minimum number of lemmings possible.



Figure 17: The Lemmings box art





Figure 19: The title screen from The Incredible Machine

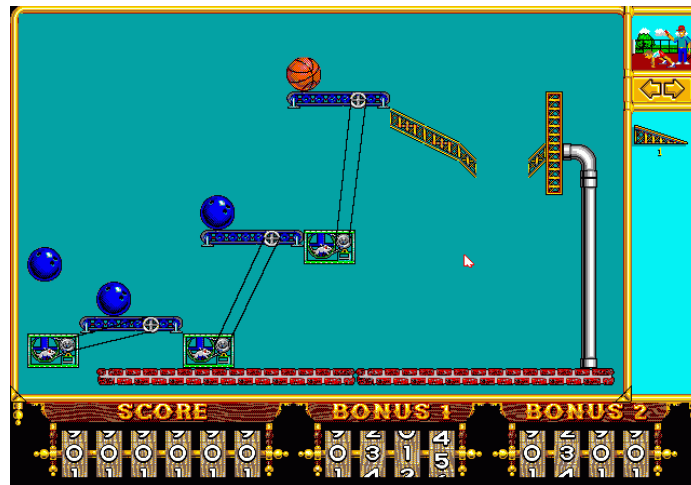


Figure 20: One of the levels from The Incredible Machine

## 0.7 Survey

### 0.7.1 Basic Survey

**What is your gender?**

- Male
- Female

**What is your age?**

(An empty text box where workers can only enter numbers)

**How many years have you been playing video games?**

(An empty text box where workers can only enter numbers)

**Fun games can be educational.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Most fun games are educational.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Games are more fun when they are cooperative (instead of competitive).**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Games are more fun when you play by yourself (instead of with others).**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Games are more fun when you play online (instead of with others in the same room).**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Educational games can be fun.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Most educational games are fun.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Games are more educational when they are cooperative (instead of competitive).**

- Strongly Agree
- Agree

- Disagree
- Strongly Disagree

**Games are more educational when you play by yourself (instead of with others).**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**Games are more educational when you play online (instead of with others in the same room).**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**List some fun games.**

(An empty text box where workers can write anything)

**List some educational games.**

(An empty text box where workers can write anything)

## **0.7.2 Rubric**

**Does this game have an encyclopedia of game content?  
If so, where?**

1. Game contains no encyclopedia of game content
2. Game information is located online, in a non-central location
3. Game has an outside manual (or wiki) of game content
4. Game doesn't have an in-game encyclopedia, but points to a central location elsewhere

5. Game has an in-game encyclopedia of game content

**If this game contains an encyclopedia of game content, what kind of content does it contain?**

1. Game contains no encyclopedia or Encyclopedia contains no content
2. Encyclopedia contains content only related to game mechanics
3. Encyclopedia contains information about game mechanics, and also limited historical/factual information
4. Encyclopedia contains content related to game mechanics and historical/factual information
5. Encyclopedia contains content related to game mechanics and historical/factual information and outside links or references

**How many game objects or events reference real-life objects or events?**

1. No game objects or events are references to real-life objects or events
2. At least one event or object is a reference to a real-world event or object
3. Several non-connected events or objects are references to a real-world event or object
4. At least one group of objects or events are references to real-life objects or events
5. Numerous groups of objects or events are references to real-life objects or events

**Of the objects/events that are in the game, how recognizable/popular are they?**

1. Existing references are extremely obscure
2. Some of the references are popular



3. About half of the references are popular
4. Many of the references are popular
5. Most or all of the references are popular

**How much does knowing the real-life objects/events affect the gameplay?**

1. Knowing the referential material is purely irrelevant; doesn't affect the gameplay at all
2. Knowing the referential material is a little useful; only affects the gameplay a small amount
3. Knowing the referential material is somewhat useful; moderately affects the player's choices during gameplay
4. Knowing the referential material is very useful; usually affects the player's choices
5. Knowing the referential material always significantly affects gameplay

**How many difficulty levels are there in the game, and how is the difficulty changed?**

1. Game only has one difficulty
2. Game has several difficulties, but players can only select difficulty at beginning of game
3. Game has several difficulties, and players can change difficulty mid-game
4. Game has several difficulties, and prompts the player to increase or decrease the difficulty as needed
5. Game has several difficulties, and automatically adjusts the difficulty of the game as needed

**How often are tutorials offered in the game?**

1. Tutorials aren't given in the game
2. Tutorials are given at the beginning of the game

3. Tutorials are given every few levels/sections of the game
4. Tutorials are given at the beginning of every level/section
5. Tutorials are offered continuously

**If the player fails, how many game resources do they lose?**

1. Game resource penalty is extreme (e.g. restart the game)
2. Game resource penalty is large (e.g. 1/3 lives, 10 health points / 100)
3. Game resource penalty is moderate
4. Game resource penalty is small
5. Game resource penalty is nonexistent (e.g. unlimited lives)

**If the player fails, how long is the wait until they restart play?**

1. Reset time is very long (e.g. reloading level takes a long time)
2. Reset time is long (e.g. greater than 10 seconds)
3. Reset time is moderate (e.g. up to 10 seconds)
4. Reset time is short (e.g. a few seconds)
5. Reset time is very short (e.g. near instant)

**How often are there in-game checkpoints?**

1. Zero checkpoints
2. Checkpoints are few and far between (e.g. levels are the only places to restart)
3. Checkpoints are moderate
4. Checkpoints are numerous (e.g. players can restart at the beginning of each puzzle)
5. Checkpoints are frequent (e.g. players can restart part of the way through puzzles)

**How much freedom of choice is there in the game, including both game world and choice of lessons?**

1. Players are placed in a strictly linear world or lesson progression
2. Players are allowed just a few large-scale choices in their game world
3. Players have the option to make choices about the direction of their progression in the world, but it is largely linear
4. Players can choose from many choices within the game world to explore, including lessons
5. Players are free to choose the direction they want, both educationally and within the game world; allowed to jump between parallel lessons

**How much and how often is there feedback regarding progress on the game?**

1. Game gives no feedback other than high-level progression through the game
2. Game gives feedback after each level
3. Game gives feedback at various points through a level, after a series of puzzles
4. Game gives feedback after each puzzle
5. Game constantly gives feedback (e.g. during a puzzle)

**Are problems able to be solved multiple ways, or circumvented entirely?**

1. There is only one way to solve any given problem, with one given progression that is valid as a solution
2. Some problems within the game may be solved more than one way
3. Multiple solutions are available for each problem, but players are limited to using one of those solutions

4. Players can solve each problem via any solution, so long as they do not circumvent it
5. Players can solve a problem any way they like, or even circumvent the problem, and be given full (or bonus) points

**This game was fun.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**I had fun playing this game.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**This game was educational.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**I learned something from playing this game.**

- Strongly Agree
- Agree
- Disagree
- Strongly Disagree

**What did you learn from playing this game?**

(An empty text box where workers can write anything)

**Provide some comments and feedback on the game.**

(An empty text box where workers can write anything)

**Provide some comments and feedback on this survey.**

(An empty text box where workers can write anything)

**Estimate the number of minutes that you've played this game.**

(An empty text box where workers can only enter numbers)

**Estimate your highest score/level from your play session.**

(An empty text box where workers can only enter numbers)

### **0.7.3    darfur Quiz**

**When did the violence in Darfur first start?**

1. 2001
2. 2002
3. 2003
4. 2004
5. 2005

**Who is the fighting between?**

1. two different tribes
2. arabs and non-arabs
3. Hutu and Tutsi
4. Blacks and Whites
5. rich and poor

**A genocide is defined as killing, causing serious bodily or mental harm and preventing births of ...**

1. a specific national group
2. a specific ethnical group
3. a specific racial group
4. a specific religious group
5. all of the above

**How did the conflict start?**

1. bombing in a main city
2. government's drafting of soldiers
3. accidental killing of 5 civilians by police
4. rebellion attacks of government targets
5. public protest against new law

**How many people have died as a result of the violence, hunger, and disease?**

1. up to 3,000
2. up to 30,000
3. up to 300,000
4. up to 3,000,000
5. up to 3.5 million

**Where is Darfur?**

1. northern Sudan
2. southern Sudan
3. eastern Sudan
4. western Sudan
5. central Sudan

**When was the height of the violence in Darfur?**

1. 2003-2004
2. 2003-2005
3. 2004-2005
4. 2004-2006
5. 2005-2006

**What group is referred to as Devils on horseback?**

1. Sunnis
2. Shiite
3. Tumoids
4. Janjaweed

**Darfur is a region in the African country of Sudan. Which countries border Sudan?**

1. Rwanda, Somalia, Tanzania
2. Algeria, Morocco, Tunisia
3. Central African Republic, Chad, Libya
4. Kenya, Ethiopia, Malawi

**Which two major powers united with the Sudanese government to keep the United Nations out of Darfur?**

1. United States and Canada
2. Iraq and Iran
3. Egypt and Libya
4. Russia and China

### **0.7.4 oregon Quiz**

**How many miles long was the Oregon Trail?**

1. 500 Miles
2. 1000 Miles
3. 2000 Miles
4. 3000 Miles
5. 20000 Miles

**What is the name of a disease like malaria that the pioneers might catch?**

1. ague
2. pneumonia
3. epilepsy
4. measles

**What year was the first Oregon Trail wagon train organized?**

1. 1847
2. 1843
3. 1899
4. 1876
5. 1836

**What disease killed more people on the trail than any other?**

1. smallpox
2. plague
3. cholera
4. scarlet fever



**When was the first transcontinental railroad finished that eventually ended the Oregon Trail?**

1. 1867
2. 1870
3. 1899
4. 1869

**How many modern states did the travelers travel through when crossing the trail?**

1. 6
2. 8
3. 3
4. 10

**How many people died on the Oregon Trail?**

1. 50,000-60,000
2. 20,000-30,000
3. 90,000-100,000
4. 10,000-20,000

**Where did the Oregon Trail begin?**

1. Independence, Mississippi
2. Independence, Missouri
3. Independence, Michigan
4. Independence, Montana
5. Independence, Massachusetts

**Where did the Oregon Trail end?**

1. Vancouver, Washington
2. The Dalles, Oregon
3. Portland, Oregon
4. Stevenson, Washington
5. Oregon City, Oregon

**How many people came west on the Oregon Trail?**

1. around 500
2. around 5000
3. around 50000
4. around 500000
5. around 5000000

### **0.7.5 munchers Quiz**

**$39 + 33 =$**

1. 67
2. 87
3. 73
4. 81
5. 72

**$15 + 82 =$**

1. 112
2. 106
3. 97
4. 81
5. 113

**-77 - -94 =**

1. 14
2. 36
3. 35
4. 17
5. 10

**68 - 92 =**

1. -8
2. -15
3. -24
4. -29
5. -38

**18 \* 9 =**

1. 172
2. 162
3. 150
4. 173
5. 157

**15 \* 8 =**

1. 121
2. 131
3. 122
4. 120
5. 119

**$14 * 6 =$**

1. 88
2. 71
3. 84
4. 68
5. 72

**$192 / 16 =$**

1. 24
2. 18
3. 27
4. 25
5. 12

**$143 / 11 =$**

1. 23
2. 13
3. 31
4. 30
5. 14

**$276 / 12 =$**

1. 38
2. 42
3. 41
4. 23
5. 20

### 0.7.6 Lightbot Quiz

What does this conditional evaluate to? (not ((true) and (false))) and (not ((false) or (not false)))

1. true
2. false

What does this conditional evaluate to? ((not false) and (not true)) and ((false) and (not true))

1. true
2. false

What does this conditional evaluate to? not (((not true) and (false)) and ((true) and (false)))

1. true
2. false

What does this conditional evaluate to? ((true) and (true)) or ((false) and (not true))

1. true
2. false

Examine the following code:

```
int count = 0; while ( count <= 6 ) { System.out.print(
count + " "); count = count + 2; } System.out.println( );
```

What does this code print on the monitor?

1. 1 2 3 4 5 6

2. 0 2 4 6 8
3. 0 2 4
4. 0 2 4 6

Examine the following code:

```
int count = 7; while ( count >= 4 ) { System.out.print(
count + " " ); count = count - 1; } System.out.println( );
```

What does this code print on the monitor?

1. 1 2 3 4 5 6 7
2. 7 6 5 4
3. 6 5 4 3
4. 7 6 5 4 3

Examine the following code:

```
int count = -2; while ( count < 3 ) { System.out.print(
count + " " ); count = count + 1; } System.out.println( );
```

What does this code print on the monitor?

1. -2 -1 1 2 3 4
2. -2 -1 1 2 3
3. -3 -4 -5 -6 -7
4. -2 -1 0 1 2

Examine the following code:

```
int count = 1; while ( count <= 5 ) { System.out.print( count
+ " " ); } System.out.println( );
```

What does this code print on the monitor?

1. 1 2 3 4
2. 1 2 3 4 5
3. 2 3 4
4. 1 1 1 1 1 1 1 1 1 1 . . . .

Examine the following code:

```
function foo() { bar(); print('foo'); } function bar() { print('bar'); }  
}
```

What does the code "foo(); bar()" print on the monitor?

1. foobar
2. barfoobar
3. foobarfoo
4. barbarfoo

Examine the following code:

```
function foo() { bar(); print('foo'); } function bar() { print('bar'); }  
}
```

What does the code "bar(); foo()" print on the monitor?

1. foobar
2. barfoobar
3. foobarfoo
4. barbarfoo

## 0.8 Conclusion