# Boolean Logic and Logic Gates, the Gateway to Understanding FPGAs.

Internet of Things (IoT) at UCF

UNIVERSITY OF CENTRAL FLORIDA

*UCF RET Site: Collaborative Multidisciplinary Engineering Design Experiences for Teachers*

## IB Computer Science

**James R. Ebbert**
AP Computer Science A
IB Computer Science SL
IB Computer Science HL

July, 2017

# RET Site: CoMET Lesson/Unit Plan

**Course(s):** **AP Computer Science A, IB Computer Science SL, IB Computer Science HL**
**Grade Level:** **These courses are typically taught in 10th, 11th, and 12th grade.**
**Suggested Length of Lesson:** **11 to 15 days**

**Materials/Technology Needed**
- Handouts for class work and Long-Term Assignments
- Computer with Data Projector
- Student computers with Internet access.

**Where this Fits**
From the IB Curriculum Guide:
- Topic 2 – Computer Organization
  - Operating systems and application systems

**Lesson Objective(s)/Learning Goal(s)**
- Students will understand how to make truth tables and truth maps and how they relate to Boolean expressions.
- Students will understand multiple representations of Boolean expressions including truth tables and maps, Boolean algebraic expressions, and logic gate diagrams.
- Students will understand how to construct a circuit to perform simple calculations such as adding two binary numbers using logic gates.
- Students will understand the interrelationships between all of these topics as a form of multiple representations.
- Students will construct physical models of logic gates and be able to explain their function.
- Students will be able to write a Java method that adds two binary numbers using Boolean logic.

**Standard(s)/Benchmark(s) Addressed**
From the IB Curriculum Guide:
- 2.1.9 Define the terms: bit, byte, binary, denary/decimal, hexadecimal.
- 2.1.11 Define the Boolean operators: AND, OR, NOT, NAND, NOR and XOR.
- 2.1.12 Construct truth tables using the above operators.
- 2.1.13 Construct a logic diagram using AND, OR, NOT, NAND, NOR and XOR gates.

**Standards for Mathematical Practice**
- Make sense of problems and persevere in solving them.
- Reason abstractly and quantitatively.
- Model with mathematics.
- Use appropriate tools strategically.

**Evidence of Learning (Assessment Plan)**
- Students will have a pre- and post-test to check for learning.
- Students will take a culminating quiz.
- Topics from this unit could appear on any future tests as all of my tests are cumulative.

**Instructional Strategies**
- Spiral learning
- Multiple representations
- Scaffolding / Gradual Release Model
- Repetition (for long-term learning)
- Progress monitoring
- Think-pair-share
- Chunking

**Description of Lesson Activity/Experiences *(Numbers and letters in parenthesis are for document references)***

**Day 1:** *(the day prior to the actual start of the lesson)*

- **Pre-Test**
- Show the PowerPoint announcing the pre-test (1a).
- At the start of class on the day before the lesson starts, have students complete the pre-test (1b). This should take less than 30 minutes. After completing the pre-test students may work on long-term programming and reading assignments for this class.

**Day 2:**

- **Introduction to Logic Operators (And, Or, Not, Truth Tables)**
- Show the PowerPoint (2a) and explain each part. Be sure to get responses from students for items in red text. Also, provide time for students to complete parts of the in-class assignment as indicated.
- Have students do the in-class assignment (2b). Review the completed problems with students after #3, #5, and #6. Document (2c) is the answer key. The PowerPoint also has the blank truth-tables to use when reviewing problems #1-3.

**Day 3:**

- **More Truth Tables. Introduction to Truth Maps. Multiple Representations.**
- Show the PowerPoint (3a) and explain each part. Be sure to get responses from students for items in red text. Also, provide time for students to complete parts of the in-class assignment as indicated.
- Have students do the in-class assignment (3b). Review the completed problems with students after #2, and #4. Document (3c) is the answer key.
- Continue the PowerPoint introducing Boolean Algebra (3a)
- Assign the Long-Term Assignment (3d)

**Day 4:**

- **Logic Gates. Introduction to Building Circuits using Logic Gates.**
- Show the PowerPoint (4a) and explain each part. Be sure to get responses from students for items in red text. Also, provide time for students to complete parts of the in-class assignment as indicated.
- Have students do the in-class assignment (4b). Review the completed problems with students after #2, #4, #6, and #7. Document (4c) is the answer key.
- Students will complete #8 and "CodingBat Java Warmup-1 > sleepIn" for homework.
- Also remind students to continue working on their Long-Term Assignment.

**Day 5:**

- **Review the homework and CodingBat problem.**
- **Mechanical equivalent for AND and OR.**
- Introduce group investigation into multi-bit adder.
- Assign two new CodingBat problems for homework:
  - "CodingBat Java Warmup-1 > monkeyTrouble"
  - "CodingBat Java Warmup-1 > parrotTrouble"

**Day 6:**

- **Introduction to IoT**
- Long-Term Assignment: Summary Introduction to IoT.
- Continue working on previous assignments.

**Day 7:**

- **Introduction to Basys3**
- Introduction to Basys3. New Long-Term Assignment: Summary report on Basys

**Day 8:**

- **Programming a binary adder**
- Write a Java method to emulate adding two 4-bit binary numbers using only logic operators.
- Write a Java method to emulate adding two $n$-bit binary numbers using only logic operators.
- These problems both appear on: codingbat.com/home/jebbert@volusia.k12.fl.us

**Many days later:**

- **Long-Term Assignment debriefings**
- Discussion of summary reports on IoT and Basys3.
- Debriefing on binary adder.
- Relate Logic Circuit design to Basys3.

**Final Day:**

- **Post-test**
- Show the PowerPoint announcing the post-test (FDa).
- At the start of class have students complete the post-test (FDb). This should take less than 45 minutes. After completing the post-test students may work on long-term programming and reading assignments for this class.

**Follow-up:**

- **For Further Exploration**
- Long-Term Assignment: Getting creative with Logic and Logic Circuits.
- Students can choose to look deeper into one (or more) of three topics:
  - The Internet of Things in general (IoT)
  - FPGAs and specifically the Basys3
  - Design and build a mechanical logic gate device.
    - Using mechanical electric switches and battery power
    - Purely mechanical

**Recommended Assessment(s) and Steps**

- A pre-test is given prior to the first lesson that will help determine prior knowledge.
- Periodic questioning, in-class assignments, and long-term assignments assist in progress checking.
- Long-Term Assignments are collected and graded, then discussed in class. These events are not listed in the lesson plan to allow flexibility in scheduling. Some Long-Term Assignments might not be discussed until after the Post-Test. This will further enhance the spiral nature of these lessons.
- A post-test is given near the end of the unit to discover any deficiencies that need to be addressed.
- After the unit, students take quizzes and tests that include some of these topics and further serve to verify learning has taken place.
- As a follow-up, students can choose to delve further into one (or more) of three topics.
- By using cumulative testing for all quizzes and tests, retention of knowledge and skills is repeatedly encouraged and monitored.

**List of Materials/Resources Used**

- For physical model option: Battery holder, batteries, low voltage DC switches, low voltage lamps or LEDs, wiring solution of some form.
- Students need access to computers with Internet connections for the CodingBat problems.
- The teacher needs access to a computer with an Internet connection and a data projector.
- There are several daily assignment and long-term assignment handouts which need to be printed in advance.

# Important Vocabulary

| Term | Definition |
|---|---|
| Logic Gate | A logic gate is an elementary building block of digital circuits such as AND, OR, NOT. |
| Logic Operator | A logic operator is a mathematical or symbolic representation of a logic gate. |
| Logic Map | A logic map is an alternative way to represent truth tables using a rectangular array. |
| Conditional Statement | A conditional is a command that controls the flow of a program based on the state of a condition. An “if-else” construct is an example of a conditional statement. |
| FPGA | Field Programmable Gate Array |
| AND | (A AND B) is true if and only if both A and B are true. |
| OR | (A OR B) is true if either A or B or both are true. |
| NOT | (NOT A) is true if A is false; false if A is true. |
| XOR | (A XOR B) is true if A is true or if B is true but not if both are true. |
| NAND | (A NAND B) is true if both A and B are false. (This is a NOT of an AND) |
| NOR | (A NOR B) is true if both A and B are true. (This is a NOT of an OR) |

# Troubleshooting Tips

For the in-class assignment "Logic Operators and Truth Tables" (2b), problem #5 may cause some difficulties for students. That is really the point of that problem. Part of the purpose of that problem is to get students to realize the need for an agreed-upon order of operations for logic operators.

# Attachments

- PowerPoints: 1a, 2a, 3a, 4a, 5a, 6a, 7a, 8a, MDL, Final-Day, Follow-Up.
- Handouts: Pre-Test, 1b, 2b, 3b, 4b, 5b, Post-Test b, Follow-Up b
- Answer Keys: Pre-Test c, 1c, 2c, 3c, 4c, Post-Test c.
    - (Note): Pre- and Post-Tests are the same.

# Other Helpful Information

The teacher should be familiar with IoT, FPGAs, and the Basys3 prior to starting this series of lessons. While students are required to learn about these topics through research and reading, the teacher still needs to be knowledgable and able to discuss these topics with the students.

This series of lessons attempts to help students understand all of these Computer Science conceptual frameworks through the use of several different pedagogical approaches. Each of these approaches is addressed independantly. However, the intent is that throughout the lessons all of these approaches will fluidly merge into a cohesive whole.

**Scaffolding:**

This series of lessons will gradually build a framework for understanding with supportive instruction, questioning, and activities. However, as the scaffolding process progresses, care will be taken to include higher-order questioning throughout the process, even before students have obtained a full understanding of the underlying concepts in some cases. When explorations or questions are posed beyond the scaffolded support-level, care will be taken to make sure the questions are at the appropriate Zone of Proximal Development (ZPD).

It may be helpful to clarify what is meant by scaffolding:

> van de Pol (2010) explains:
>
> Borrowed from the field of construction, where a scaffold is a temporary structure erected to help with the building or modification of another structure, the use of scaffolding as a metaphor within the domain of learning refers to the temporary support provided for the completion of a task that learners otherwise might not be able to complete.

The intent of scaffolding is not to give a completed structure of knowledge to students, but rather to provide appropriate support and guidance while the students construct their own understanding of that structure.

In Stone's view (1993), students are more than passive participants in teacher–student interaction. Scaffolding should be a fluid process in which both participants are actively involved.

This means that scaffolding should be supportive, developmental, and fluidly adaptable based on the needs of students. The fluidity is difficult to express in a "stock" lessson plan such as this. In reality, my lessons are never staticly produced. For example, in this series of lessons, I will only print the pre-test and the first handout ahead of time. Based on the results of the pre-test and the first day's lesson, I will adapt the materials for the next day. This fluidity requires quite a bit of support. For example, the ability to print on demand, time to make modifications, and the ability of the teacher to analyze and adapt.

**Gradual Release Model:**

The Gradual Release Model is not difficult to understand. The basic premis is similar to scaffolding, in that the early stages of learning a particular topic may require significant support from the teacher, with a goal of gradually scaling back that support and releasing the responsibility of understanding to the students.

**Combining the Gradual Release Model with the Idea of Scaffolding:**

The Gradual Release Model and the scaffolding process seem to be highly related pedagogical approaches. The major differences seems to be that scaffolding, by itself, does not automatically imply a tranfer of responsibility from the teacher to the student. However, to me these seem to be different sides of the same coin.

Without a gradual release of responsibility, scaffolding becomes a perminant support rather than a temporary support during the learning process.

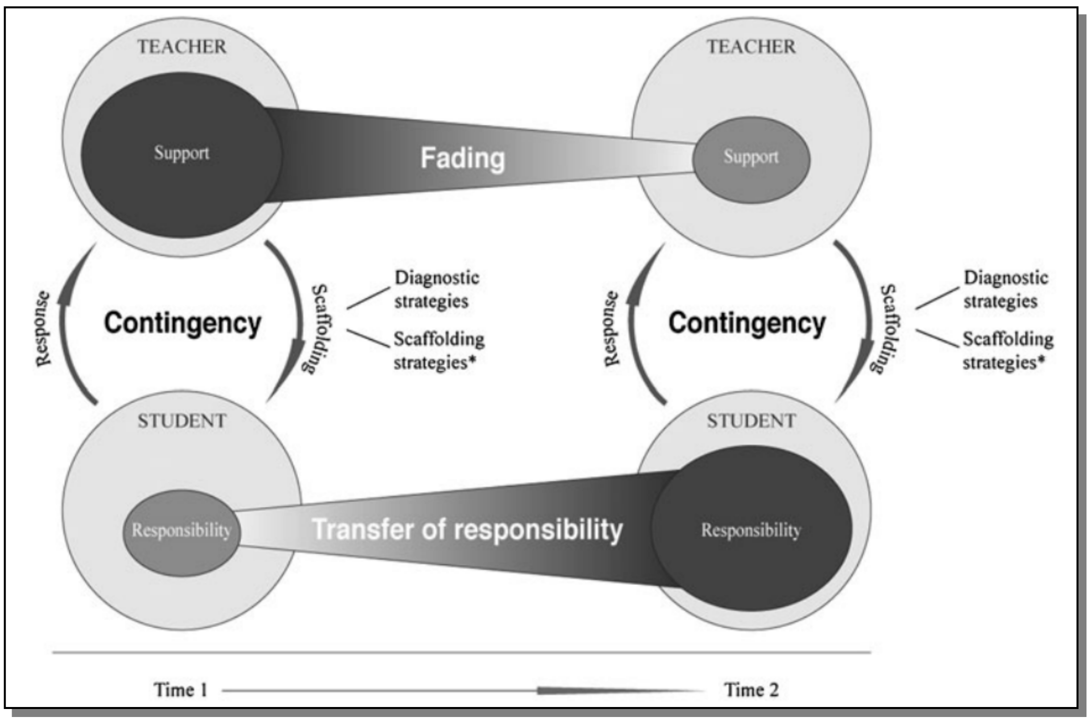


Figure 1. A visual model showing the transfer of responsibility in the scaffolding model (Pol, et. al. , 2010)

The above diagram is similar to my view of how scaffolding and the Gradual Release Model complement each other. That diagram only indicates the progression of one small learning goal within this framework. I prefer to visualize multiple versions of this occuring simultanously for a variety of related or even unrelated learning goals. I have been thinking of this process as "Spiral Learning" for over 25 years, but only recently learned that this has become an actual accepted terminology for this concept. In the case of scaffolding and the Gradual Release Model, we can focus on just the student-level responsibility as shown towards the bottom of the above diagram. By combining several of these diagrams with offset beginnings and endings, a visual representation of Spiral Learning within the scaffolded Gradual Release Model can be better understood. Each elongated triangle in my diagram represents a different learning goal. Each goal is scaffolded over time and responsibility and higher order understanding are gradually transferred to the students. The different lengths of triangles indicate that some topics may take longer than others. Also it is possible to start multiple topics at the same time or "finish" multiple topics at the same time. I put "finish" in quotes because learning goals are never really "finished" as ongoing review and support is needed for true long-term learning. So where the triangle ends on the right side should be thought of as the time at which students have a sufficiently developed understanding of the topic. But that understanding must be continually fertalized in order to maintain that understanding for the long-term.

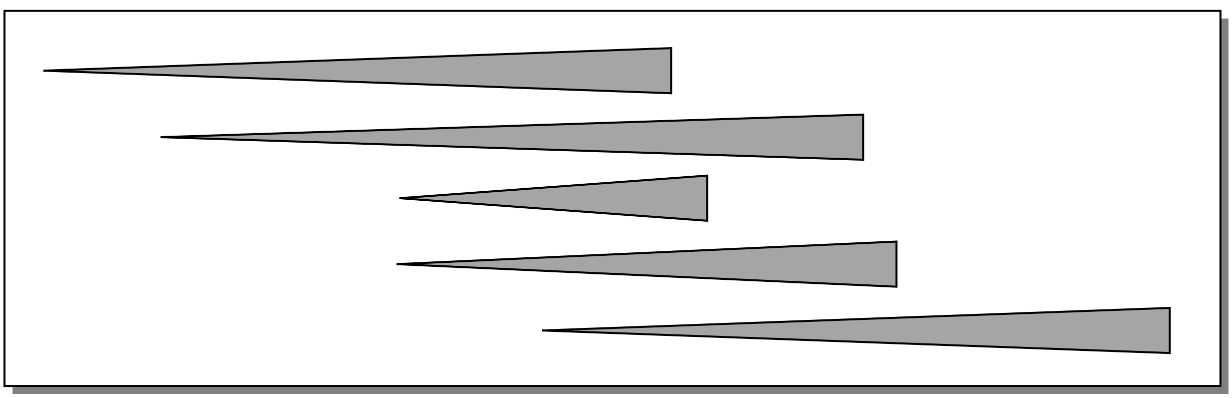

Figure 2. The relationship between Spiral Learning, scaffolding, and the Gradual Release Model. (author)

Perhaps an even better model could include a schematic representation of the feedback-and-adjust aspects of using formative assessment to adapt to the needs of the students. In this model, I use the same basic triangular outlines to show the goal of reducing required scaffolding and gradually releasing responsibility to students, but this time the triangles have a jagged edge to represent the occasional need to readjust and readdress based on errors or misconceptions that are discovered from formative assessment.

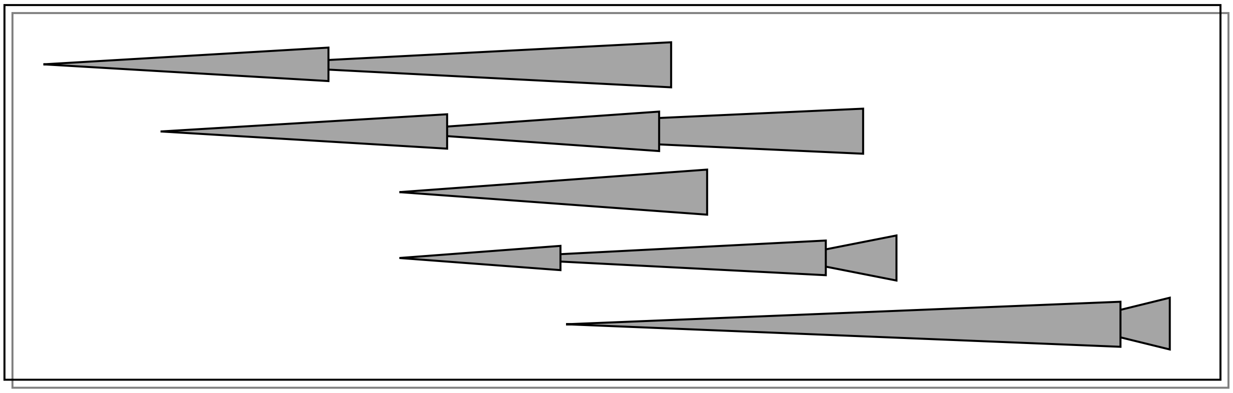

Figure 3. The relationship between Spiral Learning, scaffolding, and the Gradual Release Model, with adjustments based on formative assessment. (author)

**Multiple Representations:**

Multiple representations have numerous advantages. This is especially evident for the topics contained in my series of lessons since in addition to being multiple representations of the same underlying concepts, they also merge ideas from different areas of study that traditionally might have been taught seperately, particularly since they involve different types of computer programming in addition to physical circuit design.

Ainsworth (2006) explains:

> One of the main advantages proposed for the use of MERs is that by using combinations of representations, we can exploit their different properties to aid learning. Larkin & Simon (1987) propose that diagrams which contain the same information as equivalent written descriptions will still differ in their computational properties. For example, diagrams exploit perceptual processes, by grouping together relevant information, and hence make processes such as search and recognition easier. Further research has shown that common mathematical representations differ in their inferential power (e.g. Cox & Brna, 1995; Kaput, 1989). For example, tables tend to make information explicit, emphasise empty cells (thus directing attention to unexplored alternatives), and highlight patterns and regularity. The quantitative relationship that is compactly expressed by the equation y=x2+5 fails to make explicit the variation which is very

evident in an equivalent graph. Hence, by combining representations with different computational properties, we are no longer limited by the strengths and weaknesses of one particular representation.

However, there is normally a cost associated with multiple representations.

> Ainsworth (2006) notes that:
> Unfortunately, the benefits of an appropriate representation do not come for free. Learners are faced with complex learning tasks when they are first presented with a novel representation. They must understand how it encodes information and how it relates to the domain it represents. In addition, learners may need to select an appropriate representation or to construct one for themselves, which can provide advantages but also new cognitive tasks.

This series of lessen seeks to minimize that cost by combining topics within the International Baccalaureate Computer Science curriculum in such a way that related topics are taught concurrently as multiple representations of each other.

Hopefully by teaching these topics concurrently as multiple representations of the same underlying conceptual framework using a scaffolded approach while implementing the Gradual Release Model will lead to enhanced student success.

# References

"Computer Science Guide." IBO Curriculum Center. International Baccalaureate, n.d. Web. 5 July 2017. <http://occ.ibo.org/ibis/occ/Utils/getFile2.cfm?source=/ibis/occ/home/subjectHome.cfm&filename=dSpace%2Fen%2Fd_4_comsc_gui_1201_2_e%2Epdf>.

"Standards for Mathematical Practice." Common Core State Standards Initiative. N.p., n.d. Web. 5 July 2017. <http://www.corestandards.org/Math/Practice/>.

"Computer Science A Course Description." Collegeboard.org. The College Board, n.d. Web. 5 July 2017. <https://secure-media.collegeboard.org/digitalServices/pdf/ap/ap-computer-science-a-course-description.pdf>.

Carter, Deborah Power. "AP Computer Science Guide." AP Computer Science. College Board, n.d. Web. 5 July 2017. <https://apcentral.collegeboard.org/pdf/ap07-compsci-teachersguide.pdf?course>.

"Standards for Mathematical Practice." Common Core State Standards Initiative. N.p., n.d. Web. 5 July 2017. <http://www.corestandards.org/Math/Practice/>.

Ainsworth, Shaaron. "DeFT: A Conceptual Framework for considering Learning with Multiple Representations." DeFT: A Conceptual Framework for considering Learning with Multiple Representations - ScienceDirect. Elsevier, 2006. Web. 15 July 2017. <http://www.sciencedirect.com/science/article/pii/S0959475206000259>.

Ainsworth, Shaaron E., Peter A. Bibby, and David J. Wood. "Information Technology and Multiple Representations: New Opportunities – New Problems." Taylor & Francis. Taylor & Francis Online, 20 Dec. 2006. Web. 15 July 2017. <http://www.tandfonline.com/doi/abs/10.1080/14759399700200006>.

Panasuk, Regina M., and Matthew L. Beyranevand. "Algebra Students' Ability to Recognise Multiple Representations and Achievement." International Journal for Mathematics Teaching and Learning (2016): n. pag. 5 Oct. 2016. Web. 15 July 2017. <http://www.cimt.org.uk/journal/>.

van de Pol, Janneke, Monique Volman, and Jos Beishuizen. "Scaffolding in Teacher–Student Interaction: A Decade of Research." SpringerLink. Springer US, 29 Apr. 2010. Web. 15 July 2017. <https://link.springer.com/article/10.1007/s10648-010-9127-6>.

Stone, C. Addison. "Contexts for Learning." Google Books. Oxford University Press, 1993. Web. 15 July 2017. <https://books.google.com/books?id=mRFnDAAAQBAJ&pg=PA169&dq=%22What%2Bis%2Bmissing%2Bin%2Bthe%2Bmetaphor%2Bof%2Bscaffolding%3F%22%2Bstone&hl=en&sa=X&ved=0ahUKEwj95K7Q0YvVAhVBPiYKHQNbBTAQ6AEIJDAA#v=onepage&q=%22What%20is%20missing%20in%20the%20metaphor%20of%20scaffolding%3F%22%20stone&f=false>.

Helpful Websites:

http://www.corestandards.org/Math/Practice/

http://www.marzanocenter.com/blog/article/marzano-design-question-2-helping-students-interact-with-new-knowledge/

http://www.ldonline.org/article/5602/

Relevant topics from the IB Computer Science Curriculum:

| | Assessment statement | Obj | Teacher's notes |
|---|---|---|---|
| **Binary representation** | | | |
| 2.1.9 | Define the terms: bit, byte, binary, denary/decimal, hexadecimal. | 1 | |
| 2.1.10 | Outline the way in which data is represented in the computer. | 2 | To include strings, integers, characters and colours. This should include considering the space taken by data, for instance the relation between the hexadecimal representation of colours and the number of colours available.<br>**TOK, INT** Does binary represent an example of a lingua franca?<br>**S/E, INT** Comparing the number of characters needed in the Latin alphabet with those in Arabic and Asian languages to understand the need for Unicode. |
| **Simple logic gates** | | | |
| 2.1.11 | Define the Boolean operators: AND, OR, NOT, NAND, NOR and XOR. | 1 | **LINK** Introduction to programming, approved notation sheet. |
| 2.1.12 | Construct truth tables using the above operators. | 3 | For example, Maria won't go to school if it is cold and raining or she has not done her homework.<br>Not more than three inputs are used.<br>**LINK** Thinking logically.<br>**TOK** Reason as a way of knowing. |
| 2.1.13 | Construct a logic diagram using AND, OR, NOT, NAND, NOR and XOR gates. | 3 | Problems will be limited to an output dependent on no more than three inputs.<br>The gate should be written as a circle with the name of the gate inside it. For example:<br>(OR)<br>**LINK** Thinking logically, connecting computational thinking and program design, introduction to programming. |

("Computer Science Guide", n.d.)

# Acknowledgements

## Author

*James R. Ebbert*

## Acknowledgments

Special Thanks to: Dr. Reza Abdolvand, Sarah Shahraini, Xiaolong Guo, Stephen Williams, Ramin Izadpaneh, and S. Safa Bacanli. I would also like to thank everyone involved with producing this program and my fellow participants.

## Supporting Program

RET Site: COMET Program, College of Engineering and Computer Science, University of Central Florida. This content was developed under National Science Foundation grant #EEC-1611019.

## Contact information

James R. Ebbert
jebbert.deland@gmail.com
jebbert@volusia.k12.fl.us

## After teaching the series of lessons:

- Remember to do your 3R reflection include an updated copy of your lesson plan, developed assessment tools, presentation materials, to the evaluator. See implementation plan instructions developed by the evaluator. Send within a week after completing the lesson to bonnie.swan@ucf.edu

## UCF 3R Reflections:

The conceptual framework of the College of Education at UCF includes the development of its students as reflective practitioners. Students are required to complete a Professional Portfolio (LiveText) consisting of reflections and evidence gathered as they progress through the teacher education program.

These reflections must be typed in the 3-R Format. A reflection is not a report of factual information. It is an expression of your expectations, perceptions, and feelings of the experience represented by your evidence. It is essentially a journal of your personal growth. The 3-R Format is described below:

(1) Reaction (Affective Domain, To Feel)

As you examine this evidence, how do you feel about it now? Cite at least one example that illustrates your response.

(2) Relevance (Cognitive Domain, To Think)

How is the evidence related to teaching and learning? What are some alternative viewpoints or perspectives that you now have and/or what are some changes/improvements you might make based on the experiences you have had?

(3) Responsibility (Psychomotor Domain, To Do)

How will the knowledge gained from the experience be sued into your profession? Give examples of possible applications in your professional life, as well as an analysis of possible alternatives, other perspectives, or other meanings that might be related to the evidence. What are some questions you still have regarding this topic?