



7-17-2022

Finding Approximate Pythagorean Triples (and Applications to LEGO Robot Building)

Ronald I. Greenberg

Loyola University Chicago, Rgreen@luc.edu

Matthew Fahrenbacher

Niles Township High School, matfah@d219.org

George K. Thiruvathukal

Loyola University Chicago, gkt@cs.luc.edu

Follow this and additional works at: https://ecommons.luc.edu/cs_facpubs



Part of the [Geometry and Topology Commons](#), [Other Computer Sciences Commons](#), [Other Education Commons](#), and the [Robotics Commons](#)

Recommended Citation

Ronald I. Greenberg, Matthew Fahrenbacher, and George K. Thiruvathukal. Finding approximate pythagorean triples (and applications to LEGO robot building). <http://rig.cs.luc.edu/~rig/PythagoreanLegoCSTA-2022> linked from <http://csteachers.org/nifty>, July 2022, dot:10.6084/m9.figshare.21638066

This Other is brought to you for free and open access by the Faculty Publications and Other Works by Department at Loyola eCommons. It has been accepted for inclusion in Computer Science: Faculty Publications and Other Works by an authorized administrator of Loyola eCommons. For more information, please contact ecommons@luc.edu.



This work is licensed under a [Creative Commons Attribution-Noncommercial-No Derivative Works 3.0 License](#).
© CSTA, 2022.

Finding Approximate Pythagorean Triples (and Applications to LEGO Robot Building)

Ronald I. Greenberg, Loyola University Chicago (rig@cs.luc.edu)

Matthew Fahrenbacher, Niles Township High School District 219 (matfah@d219.org)

George K. Thiruvathukal, Loyola University Chicago (gkt@cs.luc.edu)

Abstract

This assignment combines programming and data analysis to determine good combinations of side lengths that approximately satisfy the Pythagorean Theorem for right triangles. This can be a standalone exercise using a wide variety of programming languages, but the results are useful for determining good ways to assemble LEGO pieces in robot construction, so the exercise can serve to integrate three different units of the Exploring Computer Science high school curriculum [3]: "Programming", "Computing and Data Analysis", and "Robotics". Sample assignment handouts are provided for both Scratch and Java programmers. Ideas for several variants of the assignment are also provided.

[Presentation slides are available here.](#)

Metadata

Summary	Determine approximate pythagorean triples that are useful for building with LEGO. No LEGO is required, but this application can be a good motivator and can be used to test and demonstrate results obtained from the program.
Topics	<ul style="list-style-type: none">• variables• nested loops (e.g., "for" or "while" style)• simple math operations (arithmetic, rounding, and absolute value)• if-then with conditions involving inequalities and Boolean logic• sorting data (e.g., using a spreadsheet)
Standards	<p>CSTA K-12 Computer Science Standards, Revised 2017:</p> <p>Basic variants:</p> <ul style="list-style-type: none">• 2-AP-12: Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.• 3A-AP-14: Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.• 2-AP-11: Create clearly named variables that represent different data types and perform operations on their values.• 3B-DA-06: Select data collection tools and techniques to generate data sets that support a claim or communicate information.• 3A-DA-11: Create interactive data visualizations using software tools to help others better understand real-world phenomena.• 2-DA-07: Represent data using multiple encoding schemes. <p>Also, it is a small enough project to complete in multiple languages; support is available here for Scratch, Java, VBA, and Javascript:</p> <ul style="list-style-type: none">• 3B-AP-24: Compare multiple programming languages and discuss how their features make them suitable for solving different types of problems.
Audience	From high school through CS1 (or even a computer architecture course for the latter part of variant 3).
Difficulty	The topics stated above are the only programming concepts needed. Students should be able to finish within a few hours as long as they have been introduced to the basic programming concepts. Scratch solutions should remain within a size that is reasonably manageable and readable.
Strengths	<ul style="list-style-type: none">• Can be used to integrate three different units of the Exploring Computer Science high school curriculum (Unit 4: Programming, Unit 5: Computing and Data Analysis, and Unit 6: Robotics).• Does not require any robotics materials but allows for students who are doing Lego building to see immediate applications• Simple enough to program in Scratch, and well-suited to other programming languages as well.• No need for data files or starter code in basic variant of assignment.

	<ul style="list-style-type: none"> Ideas are provided for many variants of the assignment (and input data could be varied for the variant on exploring gear pairings) so as to impede improper sharing of solutions by students.
Weaknesses	<ul style="list-style-type: none"> Using Scratch, as in the first sample assignment handout, data output is somewhat cumbersome. In the most basic variant of the assignment, students could share solutions, but many variations could be introduced as noted below. Students could possibly bypass the instruction of working with nested loops, by devising a solution with a single loop or recursion, but this would probably also be a good learning experience.
Dependencies	<p>In addition to basic familiarity with programming constructs mentioned above (or readiness to find them in Scratch palettes), students will need sufficient mathematical comfort to use the Pythagorean Theorem and the "absolute value" operation. Students are also expected to store and retrieve files and to access a spreadsheet program that can open a CSV file, e.g., Microsoft Excel, Numbers for Mac, or Google Sheets. The sample assignment handout with Scratch programming guidance is written at the level of assuming that students know how to work with variables but that they may otherwise need some guidance in selecting Scratch blocks. The sample assignment handout for Java includes guidance regarding calculations involving square root, absolute value, and rounding.</p>
Variants	<ol style="list-style-type: none"> The data for the spreadsheet can be easily generated using any other general-purpose programming language or using VBA macros in Excel. Spreadsheet sorting operations also could be programmed in the same language used for data generation or through mechanisms such as piping to a UNIX sort command. In addition, different restrictions on side length or slope could be specified for use during the data generation, and different sorts of filtering, sorting, or other data analysis could be specified for use afterwards. In a programming language in which two-dimensional arrays are convenient, students can be asked to generate key data in a 2D array and produce an interesting visualization. Or they could be asked to do a large-scale version of the assignment in which they can explore the effect of subtle programming changes on performance (due to caching). The basic variation is motivated by basic LEGO constructions in which all parts anchored to an underlying integer grid, but more sophisticated constructions with half-unit or quarter-unit spacing can also be considered with little increase in programming complication. An optional additional exercise, exploring the available spacings and gear ratios from pairings of LEGO gears, can provide a programming exercise using structures/records (not in Scratch).

Sample assignment details

Following are sample assignment materials for (1) programming in Scratch and data analysis using Microsoft Excel, Numbers for Mac, or Google Sheets, or (2) programming in Java and more detailed data analysis instructions using Google Sheets. In both cases, the assignment handouts explain LEGO basics as motivation, though students who have already started LEGO building should be familiar with the LEGO parts:

- [html format sample assignment based on Scratch](#). A [sample Scratch solution](#) for variant #4 above can be made available to instructors upon request.
- [docx format sample assignment based on Java and Google Sheets](#) (A [grading rubric](#), [example spreadsheet output](#), and [example solution code](#) can be made available to instructors on request.)

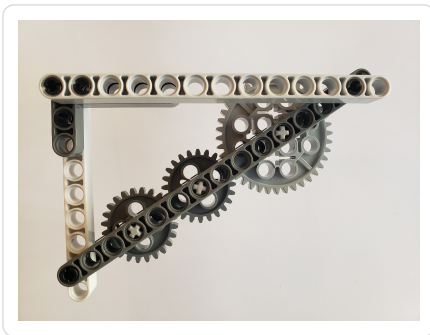
These assignment writeups focus on the programming exercise and data manipulation; the second writeup also gives some examples of data interpretation questions that can be asked. It would also make sense to ask students why the assignment suggests organizing the data in the way that it does. When Lego parts are available, students can also be asked to try creating constructions that involve small and large errors in the Pythagorean approximation and to offer further evaluation of how the data can be used.

Assignment variants

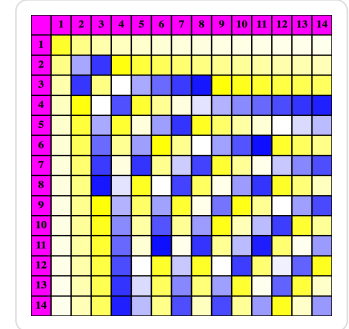
See the metadata above for a concise listing of the main assignment variants envisioned.

- The first two variants are pretty self-explanatory. (For sample code using VBA, see reference [1] and/or [2] below.)
- For the third variant, students could be asked to generate a 2D array of error or absolute error values relative to the nearest integral hypotenuse, where the row and column indices are the lengths of the short leg and long leg. An interesting display of such an array would be one in which cells get increasing levels of shading as the absolute error increases and/or different colors according to positive or negative error. Here are examples, produced from JavaScript implementations, that don't give away too much; the left image shows just absolute error and only one decimal place; the right image uses different colors for positive and negative errors and doesn't print the error values at all. (Javascript implementations can be made available to instructors upon request (with a .txt filename extension added in the link to show the content of the .html files instead of having the browser render them: [errdisplay.html](#) and [abserrdisplay.html](#).) Alternatively, instead of generating a pretty display, students could be asked to generate the 2D array with a very large range of leg lengths, and they can explore the performance effects due to caching when they change the order of the nested loops for row/column access.
- To motivate an assignment variant along the lines of the fourth idea, involving working with constructions in which some of the triangle side lengths may be measured in half-units or quarter-units, see the background in reference [1] and/or [2] below.

- To develop an exercise exploring gear pairings (a possible use of diagonal structures as in the image below), see the background in the reference [1] below. Students could hard-code data about the available LEGO gears, or they could be asked to read information from a data file or files; below, data are provided in two separate files for ordinary LEGO gears typically available in robotics kits and for double-bevel gears.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0.4	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1	0.0	0.0	0.0	0.0	0.0
2	0.2	0.2	0.4	0.5	0.4	0.3	0.3	0.2	0.2	0.2	0.2	0.2	0.2	0.1
3	0.2	0.4	0.2	0.0	0.2	0.3	0.4	0.5	0.5	0.4	0.4	0.4	0.3	0.3
4	0.1	0.5	0.0	0.3	0.4	0.2	0.1	0.1	0.2	0.2	0.3	0.4	0.4	0.4
5	0.1	0.4	0.2	0.4	0.1	0.2	0.4	0.4	0.3	0.2	0.1	0.0	0.1	0.1
6	0.1	0.3	0.3	0.2	0.2	0.5	0.2	0.0	0.2	0.3	0.5	0.4	0.3	0.2
7	0.1	0.3	0.4	0.1	0.4	0.2	0.1	0.4	0.4	0.2	0.0	0.1	0.2	0.3
8	0.1	0.2	0.5	0.1	0.4	0.0	0.4	0.3	0.0	0.2	0.4	0.4	0.3	0.1
9	0.1	0.2	0.5	0.2	0.3	0.2	0.4	0.0	0.3	0.5	0.2	0.0	0.2	0.4
10	0.0	0.2	0.4	0.2	0.2	0.3	0.2	0.2	0.5	0.1	0.1	0.4	0.4	0.2
11	0.0	0.2	0.4	0.3	0.1	0.5	0.0	0.4	0.2	0.1	0.4	0.3	0.0	0.2
12	0.0	0.2	0.4	0.4	0.0	0.4	0.1	0.4	0.0	0.4	0.3	0.0	0.3	0.4
13	0.0	0.2	0.3	0.4	0.1	0.3	0.2	0.3	0.2	0.4	0.0	0.3	0.4	0.1
14	0.0	0.1	0.3	0.4	0.1	0.2	0.3	0.1	0.4	0.2	0.2	0.4	0.1	0.2



Optional data files

Data on LEGO gears typically available in robotics kits is provided in the following two files. Each file is a CSV file with a header to indicate the meanings of the fields in the succeeding lines. Each row contains a short name (corresponding to the description used in the Botball educational robotics program [4]), the LEGO design number, the name on lego.com [4,5] (slightly modified if necessary so there won't be a comma in the field), and the radius in LEGO units.

- [CSV format data for ordinary gears](#)
- [CSV format data for double-bevel gears](#)

Experience with students

The Java version, used as an assignment in an AP CS A high school course has worked well and has been observed to generate good conversations about why the inner loop should have its control variable dependent on the outer loop. Smaller groups of high school students on robotics teams have enjoyed the Scratch version of the assignment. For beginning programmers, it may be helpful to employ roaming instructors/TAs while students work in class. For example, a subtlety of Scratch is that there are comparison operators for less-than, greater-than, and equal, but not for less-than-or-equal, so students may want to think about various ways of putting together compound Boolean tests and perhaps even using DeMorgan's Law. The idea within variant #3 of generating a very large 2D array of data, and exploring performance effects of changing the order of the nested loops, was used as a project for a university student in a computer architecture class (to see caching effects in practice). One potential pitfall in understanding that did arise in this case was that the student thought changing the order of the loops meant running the loop indices from high values to low values rather than reversing which loop was nested inside of the other. (Interestingly, this student found a significant performance gain from the change he made, which might be explainable by the type of conditional branching generated in the compiled code.)

References

- [1] Ronald I. Greenberg. Pythagorean approximations for LEGO: Merging educational robot construction with programming and data analysis. In *Proceedings of the 8th International Conference on Robotics in Education, RiE 2017*, volume 630 of *Advances in Intelligent Systems and Computing*, pages 65–76. Springer-Verlag, April 2017. Available at http://ecommons.luc.edu/cs_facpubs/169. Expanded version of reference [2].
- [2] Ronald I. Greenberg. Pythagorean combinations for LEGO robot building. In *Proceedings of the 2016 Global Conference on Educational Robotics (GCER)*. KISS Institute for Practical Robotics, July 2016. Available at http://ecommons.luc.edu/cs_facpubs/172.
- [3] Joanna Goode and Gail Chapman. Exploring computer science (version 9.0). <http://www.exploringcs.org/curriculum>, 2018.
- [4] KISS Institute for Practical Robotics. Botball educational robotics program. <http://www.botball.org>, 2019. Accessed Oct 17, 2019.
- [5] The LEGO Group. Pick a brick. <http://shop.lego.com/en-US/Pick-a-Brick>. Accessed June 1, 2022.
- [6] The LEGO Group. Bricks & pieces. <http://service.lego.com/replacementparts>. Accessed June 1, 2020.