University of Dublin

TRINITY COLLEGE

Straightedge and Compass Construction within VivioJS

Eoin O'Neill
B.A.(Mod.) Computer Science
Final Year Project May 2018
Supervisor: Dr. Hugh Gibbons

School of Computer Science and Statistics

O'Reilly Institute, Trinity College, Dublin 2, Ireland

DECLARATION

I hereby declare that this project is entirely my own work and that it has not been submitted as an exercise for a degree at this or any other university

Eoin O'Neill

Name

05/03/2018

Date

Straightedge and Compass Construction within VivioJS

Eoin O'Neill

B.A. (Mod.) Computer Science

Final Year Project

Supervisor: Hugh Gibbons

May 4, 2018

Acknowledgements

Thank you Dr. Hugh Gibbons for your continual support, guidance and enthusiasm throughout the course of this project. Also, thank you Dr. Jeremy Jones for being so generous with your time and assisting me with any queries I had regarding VivioJS. Again thank you both, it has been a pleasure!

Abstract

Euclid's Elements, written c. 300 BC, is one of the most influential texts in the field of mathematics. The text describes a system of mathematics called Euclidean Geometry. To this day, Euclidean constructions still form a significant part of the syllabi for a number of subjects in second-level education. The goal of this project was to create animations for constructions which feature on the syllabi that could be useful to students taking these subjects. The animations would also show various Euclidean Geometry concepts and display the potential of this ancient mathematical system. The axioms stated in Euclid's elements are physically interpreted through the use of a collapsible compass and straightedge. VivioJS is a system built for producing interactive web-based reversible e-learning animations. It was chosen as the tool of choice for creating the animations in this project on the grounds that the animations were portable across multiple operating systems and architectures, the abstraction level of the Vivio language suited the needs of the project, the animations were web-based and the system allowed considerable amounts of animation playback control and features. Applying human factors to design, eighteen varied animations which met the desired standards were implemented.

Contents

List of Figures

"Euclid alone has looked on beauty bare."
— Edna St. Vincent Millay, *The Harp-Weaver and Other Poems*

1 Background

1.1 Euclidean Geometry

In around 300 BC, Greek mathematician Euclid published a treatise consisting of 13 books. This treatise, titled the Elements, contained definitions, postulates, propositions and mathematical proofs of these propositions which together formed both one of the earliest and most influential mathematical systems in history, with Euclid becoming known as "the father of geometry" and making Elements a cornerstone of mathematics and a key text which would be used in mathematical education until the mid-20th century. Great praise was given to the logical and structured approach taken by Euclid in which every theorem stated in Elements could be supported by a proof, with each proof being provided to the reader. Euclid is widely quoted as having said "what has been affirmed without proof can also be denied without proof". Elements would go on to give hope to future generations of thinkers by showing them an approach to which paths of certainty could be formed.

With the invention of the first European printing presses, Elements was one of the first mathematical texts to be widely printed to the public and over the years, many abridged and revised editions of Elements were released in an attempt to make the text more readable and easier to understand. One such attempt was by Prof. David E. Joyce of Clark University who created a set of animations for each proposition in the Elements. Unfortunately, these animations are not accessible any more as browser support has ended for the Java plugin which ran these animations. This formed part of the inspiration for carrying out this project.

1.2 Tools of Construction

The axioms stated by Euclid in The Elements can be physically interpreted through the use of a collapsible compass and straightedge. Using only these instruments, one can create a great number of Euclidean geometric constructions. By applying the methods used to construct Euclidean figures, one can understand and learn much about the teachings of Euclid.

It is worth mention that a more primitive form of the straightedge and compass is a piece of chalk attached to the end of a string. It is more awkward to use but can perform all of the same tasks.

1.2.1 Straightedge

A straightedge is an infinitely long straightedged tool used for drawing straight lines. It is different to a ruler as it does not have any markings. For this reason, it cannot be used to measure or transfer distances.

1.2.2 Compass

A compass is used to inscribe arcs or circles. In the time of Euclid, a collapsible compass which would collapse when lifted from the page would be used for constructions. This would mean that the compass would not be used to transfer distances. The collapsible compass has since dwindled in popularity with non-collapsible compasses being most commonly used as the instrument of choice for drawing arcs and circles. Either tool can be used; Euclid's first and second proposition show, through the execution of some additional construction steps, a collapsible compass can be used to construct the same circle as a non-collapsible modern compass. Euclid's third proposition shows a non-collapsible compass can be used to transfer distances [10]. For the constructions animated in this project, the more modern non-collapsible compass is used.

1.3 VivioJS

VivioJS is a system built by Dr. Jeremy Jones for producing interactive reversible e-learning animations online. Animations are created using the Vivio programming language which has "syntax, semantics and functionality [...] similar to languages like C++ or Java"[8]. In addition, the system provides a high-level graphics library and integrated development environment within which animations can be run. VivioJS can produce HTML5 and JavaScript code which allows one to easily embed animations on any webpage "making them portable across browsers, operating systems and machine architectures"[9].

A key feature of VivioJS is the ability to produce animations which can be played smoothly in both forwards and reverse. There are additional features which give the user more control. Checkpoints can be placed at key frames in animations which can be jumped to in both forwards and reverse directions. Animations can be stepped through frame by frame in both directions, paused and played at any point and playback speed can be increased or decreased. Mouse or keyboard input can be handled allowing for interactive elements in an animation. Together, these features allow for a deeper learning experience. So far, the system has been used to produce interactive animations for a number of computer science topics such as processor architectures, cache coherency protocols, sorting animations and tree animations.

In 2016, past Trinity student Samuel Mardirosian undertook a Final Year Project titled "Educational Euclidean Constructions in Vivio" within which a toolset for creating Euclidean construction animations was created. Unfortunately, the project used an older version of the Vivio system. The animations produced by Vivio 5.1 required the use of a Netscape plugin and ActiveX control to run. In the past few years, web browsers have pushed to become more secure and because of this, many browser plugins which are seen as having a high risk of being compromised are being blocked. For this reason, the toolkit designed by Samuel is obsolete. Nevertheless, this was the motivation to create VivioJS which does not face this issue due to the system making use of HTML5 and JavaScript.

1.4 Motivation

Motivation for this project came from learning that Prof. David E. Joyce's animations and Samuel Mardirosian's toolkit were not accessible any more as browser support for plugins has dwindled. I believed the tools provided by VivioJS to provide a worthy solution to this problem. I had used VivioJS as a learning tool in my studies of computer architecture and I had found the cache coherency and processor animations to be the most useful resources available for learning about how these worked as the animations were interactive and provided fine control over playback. Having control over playback was especially useful for viewing animations which comprised of multiple steps, as for each step, playback could be repeated, slowed down and finely reversed (in cases where I wished to go back to a particular frame) until the step was understood. Similarly, the methods for producing Euclidean constructions also consist of multiple steps, each of which need to be understood before moving onto the next step, and the playback control Vivio animations provide could be useful for learners in these situations.

Significant portions of each syllabus for Junior and Leaving Certificate subjects such as Mathematics, Design and Communication Graphics and Technical Graphics include construction geometry. Personal motivation for this project stemmed from the fact that I studied Technical Graphics and Mathematics in secondary school. I greatly enjoyed learning about both subjects and I wanted to create a tool which would have helped me in my studies of construction geometry. This formed the goal of this project to not only produce Euclidean construction animations but ones which were designed with a student learner in mind covering constructions which are taught at secondary-level.

Euclidean geometry is where art and mathematics meet a crossing point and these constructions can show the beauty of mathematics to great effect. It is a field which provide something that can even be appreciate by those who do not have a strong interest in mathematics. Although the system is ancient, Euclidean constructions can demonstrate complex mathematical concepts; all while using only a compass and straightedge (or something as simple as a chalk attached to a piece of string).

The goal of this project was to create a set of web-friendly geometric construction animations which gave fine playback control to the user and demonstrated a multitude of mathematical concepts, the majority of which appear on the syllabi for a number Junior and Leaving Certificate subjects and could help students studying these concepts. The animations would also demonstrate a wide variety of what Euclidean geometry was capable of, despite it being an ancient mathematical system.

1.5 Existing Technologies

There are a number of systems accessible online whose suitability for the needs of this project was examined. One such application is GeoGebra, which allows users to create 2D geometric constructions in a virtual canvas using a toolset. It was not considered for this project as its intended use is not for animation. However, it functions well for those seeking educational software which allows one to create Euclidean figures and constructions.

The MATLAB environment can be used to create animations and supplies functions which allows the user to create various 2D shapes. However, it lacks much of the playback control features that VivioJS offers and mostly operates at a lower abstraction level with not as many high level animation and drawing functions available to the user.

Robocompass is a website which allows the user to create and animate geometric constructions in a 3D environment. It functions well and allows those who may have less programming knowledge to create animations using the high-level drawing commands provided by RoboCompass. Its intended use is as a teaching tool for teachers who may not have any programming experience and is designed in such a way that constructions can be created and demonstrated easily. In comparing Robocompass to VivioJS, I found that it does not allow animations to be played backwards or stepped through frame by frame. Robocompass animations are less clear and more cluttered than those produced by VivioJS and involve the animation being viewed in a 3D environment. The preferred design for animations that would be created for this project was that only a compass and straightedge would be used, so that the limitations of both of these tools would be demonstrated but also to highlight what could be done using only these instruments. Certain features of Robocompass clash with this style of design as a ruler (i.e. not a straightedge) and a set square are both used when constructing. Although these tools allow one to transfer distance and to draw perpendicular and parallel lines easier, the only tools that should be used are a straightedge and compass.

In the end, I felt that VivioJS would be the most suitable system for the project. The system is designed as a tool for educational animations which suited the aims of this project. I had personal preference towards the C++/Java-like language design implemented in the Vivio programming language and the abstraction level of the language suiting the project's needs. I had personal experience with using VivioJS animations and it worked well as a learning tool, in particular for animations which consist of multiple steps. With VivioJS I could produce cleanly designed 2D animations that could run on any web page at high performance on a large amount and variety of devices. I found that VivioJS animations could be navigated very quickly, in particular when using the keyboard controls, and I found the level of control over animations to be most suited to the project needs. For example, more control is offered to the user through the use of checkpoints, playback speed settings and precise navigation in both directions. In addition, I wanted a clean 2D canvas which could best convey the limitations of only using a compass and straightedge to create these constructions. VivioJS best fulfilled the requirements for this.

2 Methodology

2.1 Design

An important aspect of this project was providing animations which were designed in such a way that they closely matched the user requirements. My studies of human factors were very applicable for the design process of this project, and these factors were considered throughout.

Many geometric construction diagrams can be unclear to their reader. Many display every line, arc and point used throughout the entire construction process and list the steps which must be performed in order to complete the construction. This approach to design can be followed by the reader but can be improved through the use of animation. The cluttered appearance of these static diagrams means that it can be difficult to determine a construction line's point of origin and the order in which each construction line was drawn. For this reason, it was important to create a system which would be more useful than the diagrams which are typically found in a geometry textbook.

One such aspect was the use of colour in the animations. This would help catch the user's attention and increase awareness of what was happening on-screen at any given moment. Few colours were used so that animations would still be clear but visually appealing. Black was used as the main colour as negative contrast would improve visibility and readability on a screen, with the red, blue and green being used for highlighting lines, arcs and points being used and to make the animation more pleasant in appearance. These three highlighting colours were chosen together for their membership in the triadic colour scheme within which each colour is vibrant and contrasts well with the others. This would allow the colours to be easily distinguishable to the user.

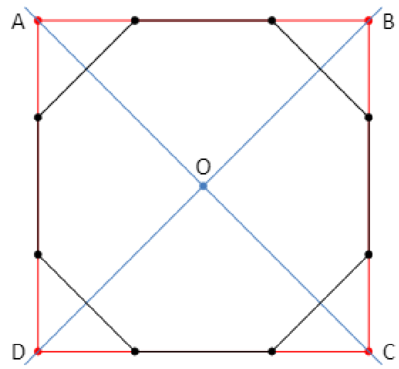

Figure 1: Use of colour in octagon animation

A crisp and simple design style was chosen with the aim being to avoid any on-screen clutter which might overload the user. This design choice is noticeable during animation playback in which each line, point and arc slowly fades out as it is not being used in construction. These lines do not need to be on-screen at all times and in order to offer a learning experience distinctive from that of someone learning construction methods through the use of a static diagram in which every construction line is visible, it was decided that these lines would not be made visible throughout the entire animation. Likewise, any student learners who wish to draw these geometric constructions while following these animations must draw every construction line on their page to complete the construction. By focusing the user's attention on individual constructions at each step, the animations offer an alternative view to what has been drawn on the page that is less cluttered. Depending on whether or not it was necessary, objects appearing on-screen such as points of intersection were faded in. This was done to avoid a jarring experience for the user. Fading in objects also makes it clearer to the user where these objects originated from.

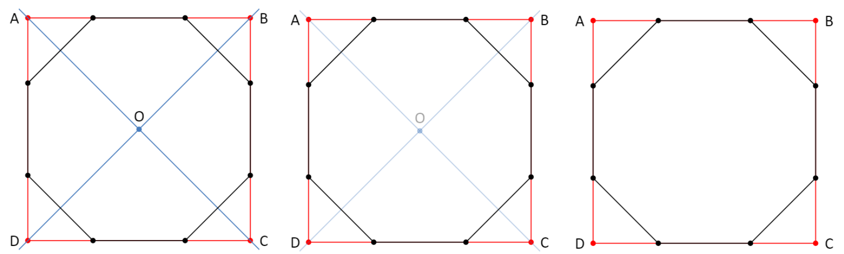

Figure 2: Two blue lines being faded out

A clear and simple font was used for any text which appeared in the animations. Only one font was used to keep the design uncomplicated to the user and as only a small amount of text would be used in most animations, there would be no justification for multiple font use. Effort was made to keep a consistent layout for each animation. Any text used to describe what appeared on-screen or to describe each step in the construction method was positioned to the side of each construction. A title was placed at the top of each animation with the name of the construction.

Many design choices were made in order to make the user aware of what was happening at any given time with the least amount of effort required from the user. As the construction of the pentagon is quite elaborate, the animation for the construction implements additional visual aids and makes use features not shown in the other animations. The animation was intended for use as an example which could show how with further work, these design decisions could be implemented for all animations. One such decision made was for whenever an arc or circle was being constructed. As these figures are being drawn on-screen, an arrow coming from the origin of the arc/circle points to where the arc is being drawn. This makes it more obvious to the user at what position the arc is centered.

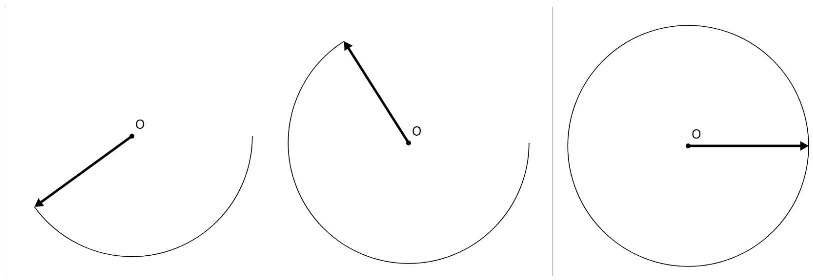

Figure 3: An arc with an arrow being drawn

Another design choice implemented in the pentagon is the use of text which appears at each step of the construction and describes what is happening. This text appears to the left of the construction and is numbered so that it is clearer to the user which step is being carried out at any given time. In addition, numbering the steps makes each step more distinguishable to the user. In this animation, a checkpoint is placed at each step. This allows the user to easily skip forwards and backwards through each step of the construction. This is particularly useful for animations which consist of many steps and saves significant time for a user who, for example, may only wish to view the last few steps of an animation. Having no checkpoints in this animation would waste time for the user as they would have to watch the animation in its entirety.

It was important to design in such a way that the user's attention could be held. The combination of fading objects and using colour was a useful way to make an object more or lass salient depending on whether or not the object's attention was required. Between each step of the animation, the program waits for a set period of time. This time period chosen so that it would be a short enough time for the user not to lose interest, but a long enough time so that a user would be able to understand the step which had just been shown to them.

Finally, rather than just having lines and shapes appear on-screen, a better alternative which would be more visually pleasing for the user and would make the constructions easier to understand was to simulate the act drawing.

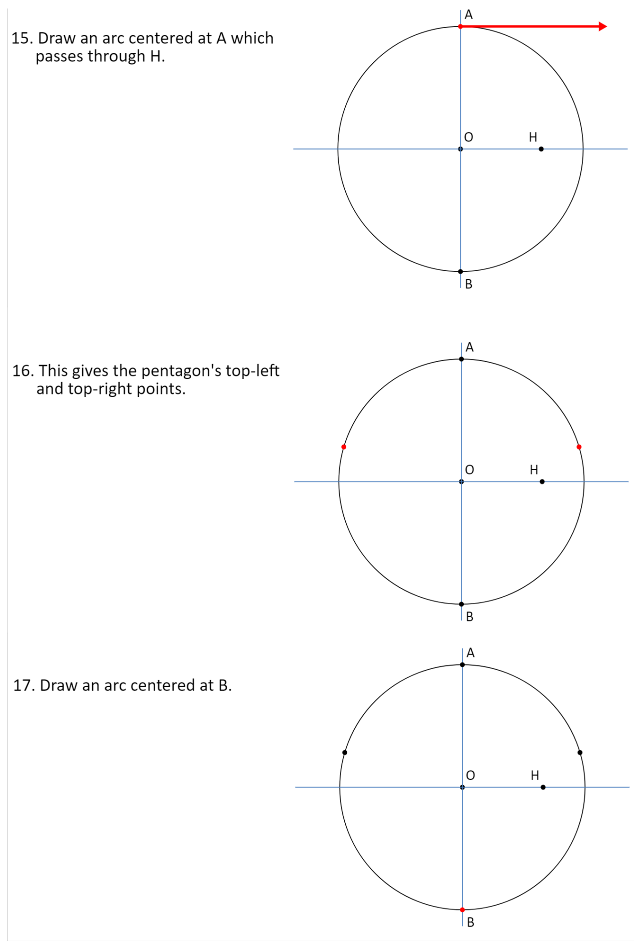

Figure 4: Descriptive text at each step

2.2 Drawing in VivioJS

The act of drawing a line in VivioJS involves the use of the line function and changing the line's endpoint position. Initially, lines appear on-screen with their starting point and endpoint at the same position, i.e. as a single point. The position of the endpoint is then changed so that it is positioned at the desired location on the canvas and the desired speed at which this should happen is specified. This gives the appearance of a line being drawn.

Figure 5: Drawing a line

Drawing a circle or arc follows a similar process. An arc with an angle of zero degrees is created first. Similar to how a line is drawn, this will appear on-screen as a dot. The angle of the arc is then set to the desired angle and the speed at which this angle should be changed is set. This appears as an arc being drawn.

Points such as origins, points of intersection, vertices and line endpoints are represented as a very small filled circle. This is done by using Vivio's ellipse function to create an ellipse with a major and minor axis of equal length, i.e./a circle. The circle can then be filled using Vivio's brush function.

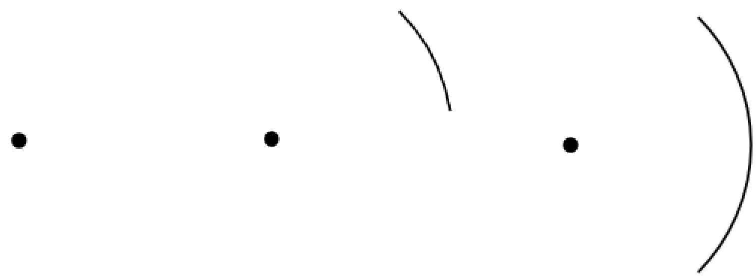

Figure 6: Drawing an arc

2.3 Basic Constructions

There are a number of basic subconstructions which together are repeated to create all compass and straightedge constructions. These five constructions are the following:

- Construction of a line which passes through two points.
- Construction of a circle which is centered at one point and passes through another point
- Construction of a point which is the intersection point of two lines
- Construction of a point which is the intersection point of two circles
- Construction of a point which is the intersection point of a line and a circle.

2.3.1 Perpendicular line bisector

The construction of a perpendicular line bisector is one of the five fundamental subconstructions which is applied very frequently in compass and straightedge construction. The following steps must be performed:

1. Given a line AB for which you wish to construct a perpendicular line which bisects line AB, construct an arc centered at point A with a radius which is greater than half the length of the line AB.

2. Construct an arc of the same radius at point B.

3. Mark points C and D where the arcs intersect.

4. Join points C and D.

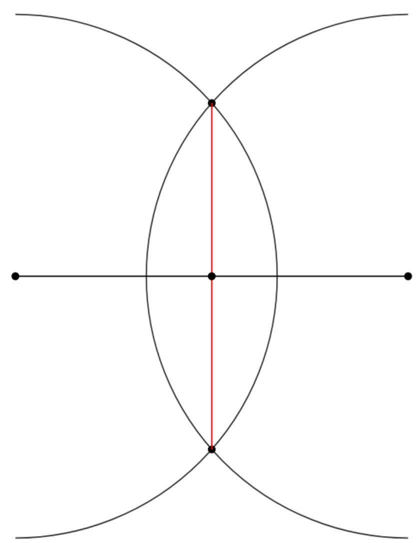

Figure 7: Perpendicular bisector

This method is used in the construction of a perpendicular to a line through a given point.

2.3.2 Perpendicular to a line through a given point

Given a line AB and a point P not on the line, a perpendicular to line AB can be constructed which passes through P:

1. Draw a circle centered at P which passes through line AB at two points. Label the two points C and D.

2. Construct a perpendicular line bisector to line segment CD. The line segment passes through point P.

This construction is used when constructing a parallel to a line through a given point.

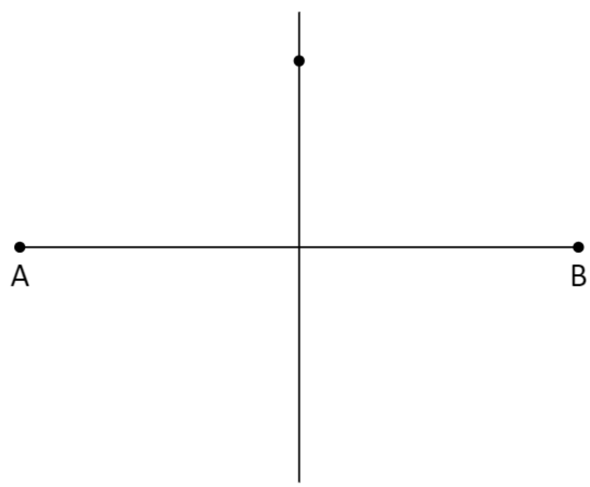

Figure 8: Perpendicular through a given point

2.3.3 Parallel to a line through a given point

The following steps are performed to construct a parallel to a line AB which passes through a given point P:

1. Construct a perpendicular to line AB through point P. Label point C where the perpendicular intersects with line AB.
2. Use a compass to obtain the length between point C and point P (length CP). Centered at point P, draw an arc with radius length CP which passes through the perpendicular line above point P. Label the point where the arc intersects the perpendicular line point D.
3. Construct a perpendicular to line segment CD. This perpendicular line will pass through point P and be parallel to line AB.

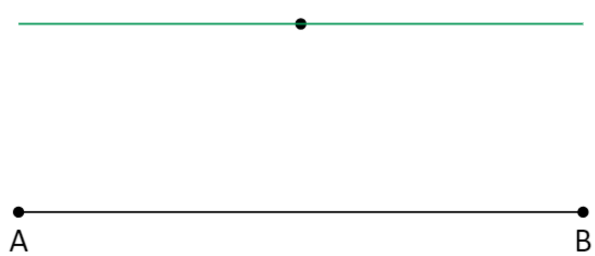

Figure 9: Parallel through a given point

2.3.4 Angle bisection

Given two lines which meet to form an angle, it is quite straightforward to bisect the angle:

1. Label the vertex of the angle point O. Draw an arc centered at O which passes through both angle legs. Label the points where the arc intersects each angle leg A and B respectively.

2. Draw an arc centered at point A which passes through point B.

3. Draw an arc centered at point B which passes through point A and intersects with the last arc. Label the point where both arcs intersect point C.

4. Join point O and point C.

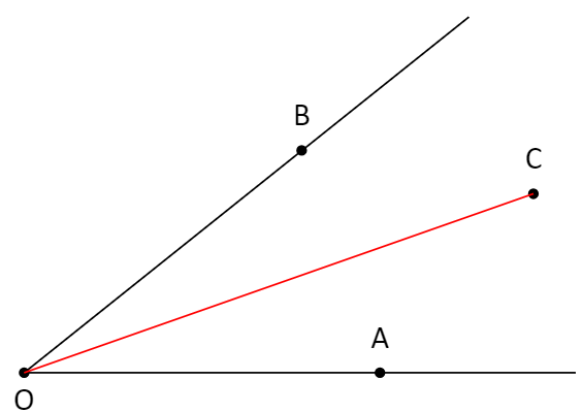

Figure 10: Bisection of an angle

2.4 Underlying Implementation

2.4.1 Cartesian Geometry

Euclidean geometry is an example of synthetic geometry in which coordinates are not used. It was originally desired to exclusively use Euclidean geometry for the constructions for this project to demonstrate what could be done using this ancient system of mathematics. In reality, when drawing compass-and-straightedge constructions one can easily transfer distances using a compass, perceive and mark off points of intersections between lines and curves. VivioJS uses a virtual coordinate space within which graphical objects are created. For this reason, Euclidean geometry alone cannot be used and one must rely on using Cartesian geometry (which uses a coordinate system). Distances between point and points of intersection are calculated algebraically in the VivioJS code using Cartesian equations.

2.4.2 Distance between two points

The distance between two Cartesian coordinates (x_1, y_1) and (x_2, y_2) on a plane is calculated using the following formula:

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

2.4.3 Line-line intersection

Given a line with endpoints (x_1, y_1) and (x_2, y_2), and a second line given by the points (x_3, y_3) and (x_4, y_4), the intersection between both lines represented by the point (P_x, P_y) is calculated using the following formula:

$$P_x = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}} \qquad P_y = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}}{\begin{vmatrix} \begin{vmatrix} x_1 & 1 \\ x_2 & 1 \end{vmatrix} & \begin{vmatrix} y_1 & 1 \\ y_2 & 1 \end{vmatrix} \\ \begin{vmatrix} x_3 & 1 \\ x_4 & 1 \end{vmatrix} & \begin{vmatrix} y_3 & 1 \\ y_4 & 1 \end{vmatrix} \end{vmatrix}}$$

2.4.4 Circle-line intersection

The intersection between a circle given by the general form equation

$$x^2 + y^2 + Ax + Bx + C = 0$$

and a line given by the equation

$$y = mx + b$$

where the points of intersection with x-coordinates $x_{1,2}$ and the y-coordinates $y_{1,2}$ are calculated using the following equations:

$$x_{1,2} = \frac{-(2md + A + Bm) \pm \sqrt{\partial_1}}{2(1+m^2)}$$

$$\partial_1 = (2md + A + Bm)^2 - 4(1+m^2)(d^2 + Bd + C)$$

$$y_{1,2} = \frac{2d - Am - Bm^2 \pm \sqrt{\partial_2}}{2(1+m^2)}$$

$$\partial_2 = (Am - 2d + m^2B)^2 - 4(1+m^2)(d^2 - Amd + m^2C)$$

These equations are also used to calculate the points of intersection between an arc and a line. Any points calculated by the circle-line equations which are not points of intersection between the arc and line are discarded.

2.4.5 Circle-circle intersection

A circle with center coordinates (a, b) and radius r_1, and a second circle with center coordinates (c, d) and radius r_2 are represented by the equations

$$(x-a)^2 + (y-b)^2 = {r_0}^2 \quad \text{and} \quad (x-c)^2 + (y-d)^2 = {r_1}^2$$

The intersection between both circles is calculated using the following equations:

$$D = \sqrt{(c-a)^2 + (d-b)^2}$$

$$x_{1,2} = \frac{a+c}{2} + \frac{(c-a)({r_0}^2 - {r_1}^2)}{2D^2} \pm 2\frac{b-d}{D^2}\partial$$

$$y_{1,2} = \frac{b+d}{2} + \frac{(d-b)({r_0}^2 - {r_1}^2)}{2D^2} \pm 2\frac{a-c}{D^2}\partial$$

$$\partial = \frac{1}{4}\sqrt{(D+r_0+r_1)(D+r_0-r_1)(D-r_0+r_1)(-D+r_0+r_1)}$$

These equations are also used to calculate the points of intersection between two arcs. Any points calculated by the circle-circle intersection equations which are not points of intersection between the two arcs are discarded.

3 Constructible Polygons

In addition to the existence of many figures which can be constructed using only a straightedge and compass, each figure can be constructed in a number of different ways. To ensure a wide variety of figures could be demonstrated, methods of construction were limited to one per each figure for which VivioJS animations have been created. The following sections detail the methods used for each construction.

3.1 Triangle

Given a side AB, very few steps are required to construct an equilateral triangle:

1. Construct an arc centered at A passing through B.
2. Construct a second arc centered at B passing through A which passes through the first arc. Label the intersection between both arcs as point C.
3. Join points A and C.
4. Join points B and C.

3.2 Square

Although it may seem simple, the procedure for constructing a square on given a side AB is more involved than one might expect:

1. Construct an arc centered at A which passes through B.

2. Construct a second arc centered at B which passes through A. Label the point of intersection between both arcs point C.

3. Centered at C, construct a third arc to the left which intersects with the first arc constructed. Label the point of intersection point D.

4. Centered at D, draw an arc which passes through A and C and intersects at the top of the third arc. Label the point of intersection point E.

5. Join point A and point E. Label the point of intersection between this line and the first arc constructed as point F. Point F is the top-left vertex of the square.

6. Draw an arc centered at point F which passes through A and intersects with the second arc constructed. This point of intersection is the top-right vertex of the square. Join all points to complete.

3.3 Hexagon

Given a circle with center O and any point P on the circle, a hexagon is constructed using the following method:

1. Construct a new circle centered at P which passes through point O.

2. Mark a point A and B where the circles intersect.

3. Construct a diameter on the original circle by constructing a line which begins at point A (passing through center O).

4. Construct a second diameter on the original circle which begins at point P.

5. Construct a third diameter on the original circle which beings at point B.

6. Join the endpoints of the three diameters (each endpoint located on the circumference of the original circle) to construct a hexagon.

3.4 Octagon

The following method is used to construct an octagon in a given square ABCD (points are labeled counter-clockwise with the top-left vertex labeled point A):

1. Draw a straight line which passes through point A and C.

2. Draw a second straight line which passes through point B and D. Label the point of intersection between both lines point O.

3. Draw an arc centered at point A passing through point O which passes through the edges of the square. Mark the two points (which are vertices of the octagon being constructed) where the arc intersects with the edges of the square.

4. Draw an arc centered at point B passing through point O which passes through the edges of the square. Mark the two points (which are vertices of the octagon being constructed) where the arc intersects with the edges of the square.

5. Draw an arc centered at point C passing through point O which passes through the edges of the square. Mark the two points (which are vertices of the octagon being constructed) where the arc intersects with the edges of the square.

6. Draw an arc centered at point D passing through point O which passes through the edges of the square. Mark the two points (which are vertices of the octagon being constructed) where the arc intersects with the edges of the square.

7. Complete the octagon by joining the vertices (the eight points which have been marked).

4 The Golden Ratio

4.1 Introduction

The golden ratio (represented by the symbol φ) exists between two quantities a and b (where $a > b > 0$) if the ratio between a and b is the same as the ratio between $a + b$. This relationship can be represented as:

$$\frac{a+b}{a} = \frac{a}{b} \stackrel{\text{def}}{=} \varphi$$

φ is an irrational number:

$$\varphi = \frac{1+\sqrt{5}}{2} = 1.6180339887\ldots$$

The golden ratio has had great influence on creators throughout history with there being many examples existing of architects, musicians, painters and others who have incorporated the ratio into their work. An interesting phenomenon is the appearance of the the golden ratio in patterns in nature, e.g. in the structure of certain plants and at an atomic level in certain crystals. In particular, the discovery of the ratio's appearance in certain constructible polygons was of interest to Euclid.

4.2 Pentagon

The following method was applied to construct a pentagon:

1. Construct a circle with origin O.

2. Draw a diameter in the circle (passing through origin O). Mark points A (which is the top vertex of the pentagon) and B at the endpoints of the diameter.

3. Construct a perpendicular to diameter AB, labeling points C and D for the intersection between the arcs used in constructing the perpendicular.

4. Draw an arc centered at C which passes through origin O and intersects the circle at two points. Label these points E and F.

5. Join points E and F. Label the point of intersection where line segment EF and CO intersect as point G.

6. Draw an arc centered at G which passes through point A and B. Label a point H where the arc intersects with the perpendicular CD.

7. Draw an arc centered at A which passes through H and intersects the circle at two points. Mark the two points of intersection (which are the top-left and top-right of the pentagon being constructed).

8. Centered at B, draw an arc of radius OH which intersects the circle at two points. Mark these two points (which are the bottom-left and bottom-right vertices of the pentagon).

9. Complete the pentagon by joining all of the vertices which have been marked.

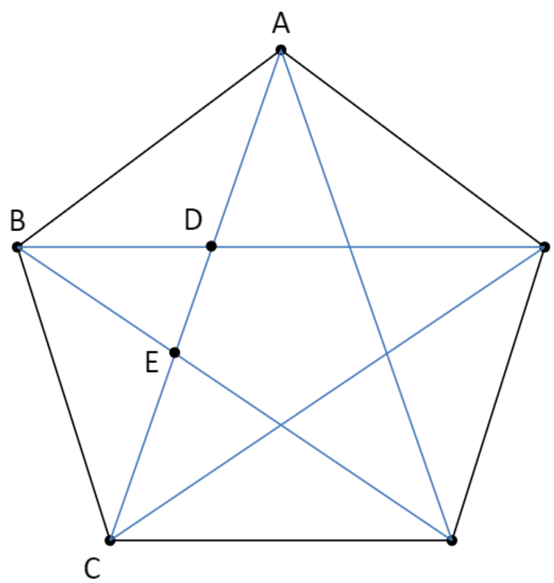

Figure 11: Golden ratio appearance in pentagon

Once the pentagon has been constructed, the diagonals of pentagon are joined to form a pentagram, best demonstrating the appearance of the golden ratio in the pentagon:

$$AC : AB = AD : DE = CE : ED = \varphi$$

4.3 Golden Rectangle

It is possible to construct a golden rectangle, i.e. a rectangle whose side lengths are in the golden ratio. From the construction of a square (the method for which has already been shown), we can construct a golden rectangle. The following method was used:

1. Construct a square ABCD (the vertices are labeled as following: top-left is A, top-right is B, bottom-left is C, bottom-right is D).

2. Construct a perpendicular bisector to AB which also intersects CD. Label

the intersection on AB as point E. Label the intersection on CD as point F.

3. Extend AB to the right. Draw an arc centered at E which passes through C and D intersecting this extended line. Label the point of intersection as point G.

4. Extend CD to the right. Draw an arc centered at F which passes through point A and B intersecting this extended line. Label the point of intersection as point H.

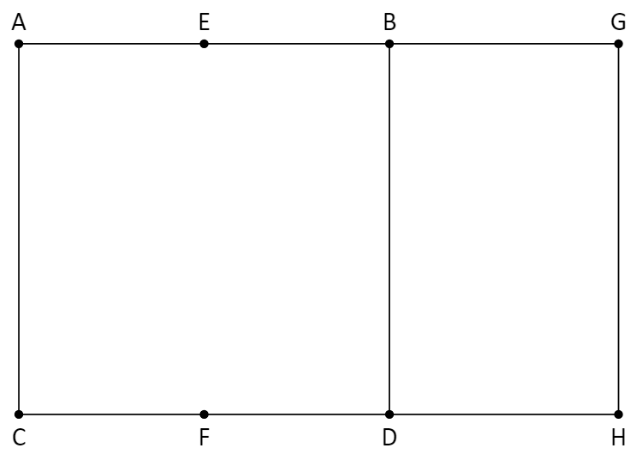

Figure 12: Golden rectangle

The ratio between the lengths of the rectangle's sides is the golden ratio:

$$AG : BG = CD : DH = \varphi$$

Removing the square section of the golden rectangle leaves another smaller golden rectangle, i.e. a golden ratio exists between the lengths of the sides in the smaller rectangle just as it exists in larger rectangle. Continually removing square sections of each smaller rectangle and marking particular vertices on each square that is come across produces an infinite sequence of points which when joined form a golden spiral. The golden spiral has been widely reported to have been found in nature.

An approximation of the golden spiral known as the Fibonacci spiral can be produced. In the Fibonacci spiral, each spiral is a quadrant. The spirals grow in relation to the Fibonacci sequence:

$$1, 1, 2, 3, 5, 8, 13, 21...$$

The sequence has applications in the field of computing, e.g. for sorting algorithms, random number generators, data structures and optimization methods.

5 More constructions taught at second-level

Along with most of the constructions shown previously, the following constructions appear on the most recent syllabi for the second-level subjects of Junior and Leaving Certificate Mathematics, Design and Communication Graphics and Technical Graphics [6, 4, 3, 5].

5.1 Incenter and incircle of a triangle

A triangle's incircle is a circle which is inscribed in the triangle. The circle touches the triangle's sides but does not cross them. The incenter is the center point of the incircle.

The following procedure was carried out to construct the incenter and incircle of a given triangle:

1. Construct the bisectors for any two angles in the triangle. Label the point of intersection between the bisectors as point O.

2. Construct a perpendicular to any side through O. Label the point of intersection between this perpendicular and triangle side as point P.

3. Centered at O, draw a circle passing through point P.

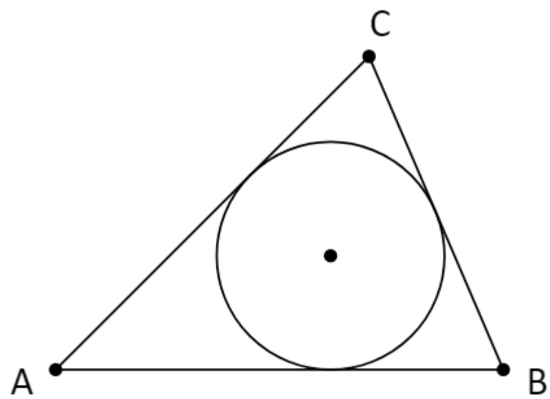

Figure 13: Incenter and incircle of a triangle

5.2 Circumcenter and circumcircle of a triangle

A triangle's circumcircle is a circle which is touches all of the triangle's vertices. The circumcenter is the circumcircle's centre.

The following procedure was carried out to construct the circumcenter and circumcircle of a given triangle ABC:

1. Construct the perpendicular bisectors for any two sides in the triangle. Label the point of intersection between the perpendicular bisectors as point O.

2. Centered at O, draw a circle which passes through each of the triangle's vertices.

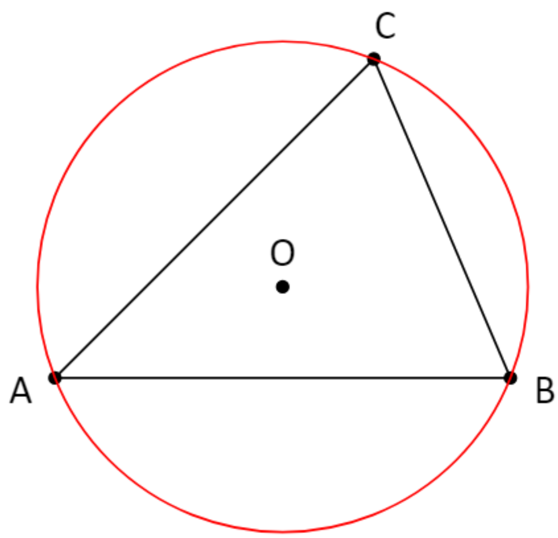

Figure 14: Circumcenter and circumcircle of a triangle

6 Euclidean Arithmetic

6.1 Constructible Numbers

A real number r is constructible if and only if, given a line segment of unit length, a line segment of length $|r|$ can be constructed with compass-and-straightedge in a finite number of steps.

A basic example of a constructible number is $\sqrt{2}$. Given a right-angled triangle with legs length 1, the hypotenuse length can be obtained by Pythagorean theorem, which is calculated to be $\sqrt{2}$. Since this figure can be easily constructed using a compass-and-straightedge, $\sqrt{2}$ is a constructible number.

Using only a compass-and-straightedge, many arithmetic operations can be performed on constructed line segments, e.g. addition, subtraction, division, multiplication, extraction of square roots [7].

The ability to algebraically solve various questions about geometry allowed ancient mathematicians to use simple instruments to solve complex mathematical problems.

6.2 Addition

The following procedure was used to add two constructible segments A and B:

1. Construct a line segment A of length a.

2. Centered at A's endpoint, draw an arc of radius b.

3. Extend line A until it intersects the arc. The extended line will be of length $a + b$.

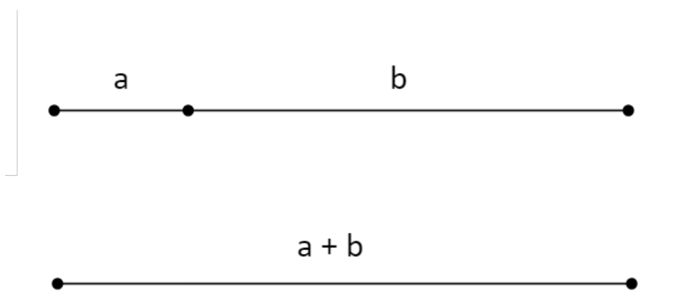

Figure 15: Addition of constructible numbers

6.3 Subtraction

The following procedure was used to add two constructible segments A and B:

1. Construct a line segment B of length b.

2. Centered at B's starting point, draw an arc of radius a passing through line segment B. The length between the point of intersection of the arc and line segment B's endpoint is equal to $b - a$.

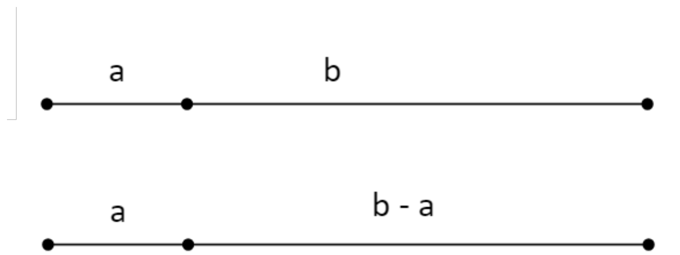

Figure 16: Subtraction of constructible numbers

6.4 Multiplication

The following construction was performed to multiply two constructible segments of lengths a and b respectively:

1. Originating from a point O, draw a horizontal line segment to the right which we will refer to as leg 1.

2. Also originating from point O, draw a second line segment of similar length at roughly 30 degrees to the first line (referred to as leg 2).

3. Centered at point O, draw an arc of length unit distance which intersects with leg 2. Label the point of intersection as point P.

4. Again centered at point O, draw an arc of length b which intersects with leg 2. Label the point of intersection as point B. $|OB| = b$.

5. Centered at O, draw an arc of length a which intersects with leg 1. Label the point of intersection as point A. $|OA| = a$.

6. Join A and P.

7. Construct a parallel to AP, passing through B and intersecting leg 1. Label the point of intersection as point Q. $|OQ| = ab$.

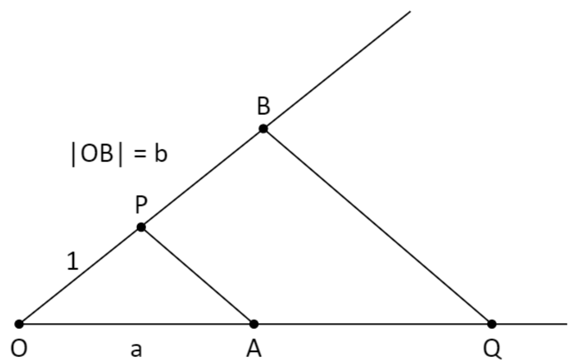

Figure 17: Multiplication of constructible numbers

Proof. Triangle OPA is similar to OBQ. So,

$$\frac{1}{a} = \frac{b}{|OQ|}$$

$$ab = |OQ|$$

□

6.5 Division

The following construction was performed to divide two constructible segments of lengths a and b respectively:

1. Originating from a point O, draw a horizontal line segment to the right which we will refer to as leg 1.

2. Also originating from point O, draw a second line segment of similar length at roughly 30 degrees to the first line (referred to as leg 2).

3. Centered at point O, draw an arc of length unit distance which intersects with leg 2. Label the point of intersection as point P.

4. Again centered at point O, draw an arc of length b which intersects with leg 2. Label the point of intersection as point B. $|OB| = b$.

5. Centered at O, draw an arc of length a which intersects with leg 1. Label the point of intersection as point A. $|OA| = a$.

6. Join A and B.

7. Construct a parallel to AB, passing through P and intersecting leg 1. Label the point of intersection as point Q. $|OQ| = a/b$.

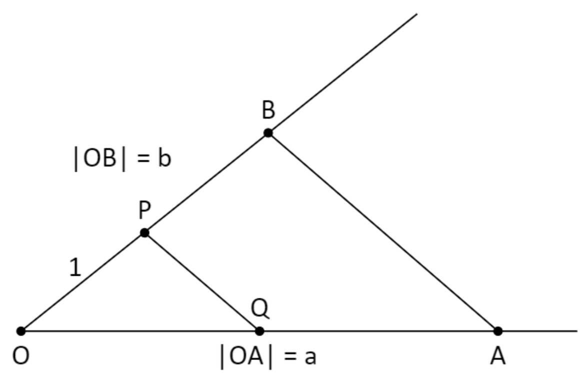

Figure 18: Division of constructible numbers

Proof. Triangle OPQ is similar to OBA. So,

$$\frac{|OQ|}{1} = \frac{a}{b}$$
$$\frac{a}{b} = |OQ|$$

□

6.6 Square Root

The following procedure is performed to produce a line segment of length $\sqrt{a}$:

1. Draw a horizontal line and mark a point O on the line.

2. Centered at O, draw an arc to the left of length unit distance and label the point of intersection between the arc and line as point P.

3. Centered at O, draw an arc to the right of length a.

4. Construct the midpoint of PA.

5. Draw a semicircle centered at PA's midpoint passing through P and A.

6. Construct a perpendicular to line PA passing through point O and the semicircle. Label the point of intersection between the perpendicular and semicircle as point Q. $|OQ| = \sqrt{a}$.

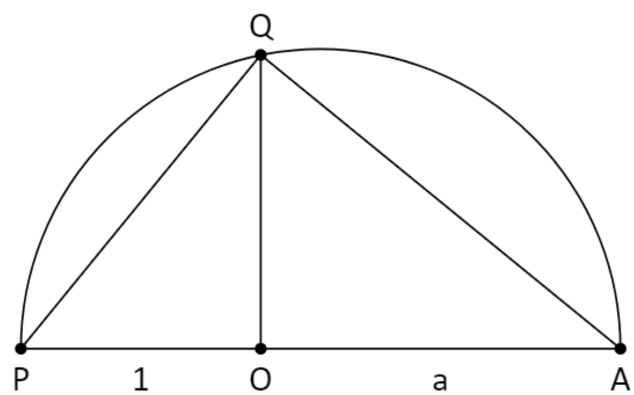

Figure 19: Extraction of square roots with constructible numbers

Proof. Construct a triangle PQA. Triangle OPQ is similar to OQA. So,

$$\frac{|OQ|}{|OA|} = \frac{|OP|}{|OQ|}$$
$$|OQ|^2 = \frac{|OP|}{|OA|}$$

□

6.7 Computing the Golden Ratio

As mentioned before, the Golden Ratio (φ) can be represented as:

$$\varphi = \frac{1+\sqrt{5}}{2} = 1.6180339887\ldots$$

φ is an irrational number but it is constructible, i.e. given a unit length, a segment of length φ can be constructed using a compass-and-straightedge. The following method was used to demonstrate this:

1. Construct a horizontal line PO of length unit distance.

2. Extend line PO to the right. Centered at O, construct an arc of radius unit distance which intersects with the extended line.

3. Centered at the intersection, construct another arc of unit distance which intersects with the extended line.

4. Repeat the process three more times until five arcs have been constructed to the right of O. Label the intersection between last rightmost arc and line OA as point A. $|OA| = 5$.

5. Construct the midpoint of PA.

6. Draw a semicircle centered at PA's midpoint passing through P and A.

7. Construct a perpendicular to line PA passing through point O and the semicircle. Label the point of intersection between the perpendicular and semicircle as point Q. $|OQ| = \sqrt{5}$.

8. Centered at O, draw an arc which passes through point Q and intersects with line PA. Label this point of intersection as point A and erase the original point A and point O. $|PA| = 1 + \sqrt{5}$.

9. Construct a midpoint O of line PA by bisecting the line (dividing line PA in two). $|PO| = (1 + \sqrt{5})/2 = \varphi$.

7 Limitations of Euclidean Geometry

7.1 Polygon constructibility

Many other polygons can be constructed using a compass and straightedge. The Gauss-Wantzel theorem states the criteria that must be met by a polygon for it to be deemed constructible:

> A regular n-gon is constructible with ruler and compass iff n is an integer greater than 2 such that the greatest odd factor of n is either 1 or a product of distinct Fermat primes. [10]

The theorem can be stated another way:

> A regular n-gon is constructible with straightedge and compass if and only if $n = 2^k p_1 p_2 \ldots p_t$ where k and t are non-negative integers, and the p_i's (when $t > 0$) are distinct Fermat primes. [2]

A Fermat prime is a prime number which is in the form

$$F_n = 2^{2^n} + 1$$

where $n > 0$. Only five Fermat primes have been discovered:

$$3, 5, 17, 257, 65537.$$

Thus, the following list contains numbers of edges of regular polygons which are constructible:

> 3, 4, 5, 6, 8, 10, 12, 15, 16, 17, 20, 24, 30, 32, 34, 40, 48, 51, 60, 64, 68, 80, 85, 96, 102, 120, 128, 136, 160, 170, 192, 204, 240, 255, 256, 257, 272, 320, 340, 384, 408, 480, 510, 512, 514, 544, 640, 680, 768, 771, 816, 960, 1020, 1024, 1028, 1088, 1280, 1285, 1360, 1536, 1542, 1632, 1920, 2040, 2048, $\ldots$ [11]

The following list contains numbers of edges of regular polygons which are not constructible:

7, 9, 11, 13, 14, 18, 19, 21, 22, 23, 25, 26, 27, 28, 29, 31, 33, 35, 36, 37, 38, 39, 41, 42, 43, 44, 45, 46, 47, 49, 50, 52, 53, 54, 55, 56, 57, 58, 59, 61, 62, 63, 65, 66, 67, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 81, 82, 83, 84, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 97, 98, 99, 100, 101, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 121, 122, 123, 124, 125, 126, 127, ... [12]

Incidentally, it is worth mentioning mathematician Oswald Hermes. In 1894, he completed a 200 page manuscript which detailed a procedure for constructing a regular 65537-gon using a compass and straightedge. It took ten years of Hermes' life to complete.

7.2 Impossibilities

There are a number of constructions in Euclidean geometry for which the methods have been found to be impossible to perform.

7.2.1 Angle trisection

In general, if given an angle, it is impossible to trisect the angle using a compass and straightedge. [1] The angle trisection problem was proven to be impossible to solve in 1837 by Pierre Wantzel. One such angle which is impossible to trisect is $\theta = 60°$. This is because $\cos(20°)$ is not a constructible number. Following this, there are infinitely many such angles which can also not be trisected.

7.2.2 Squaring the circle

Squaring the circle means to produce a square that has area π. This is impossible to do with a straightedge and compass as it requires construction of the number $\sqrt{\pi}$. π is a transcendental number meaning that it is a real number which is not algebraic. Non-algebraic numbers are non-constructible which prevents one from squaring a circle.

7.2.3 Doubling the cube

It is impossible to double or duplicate a cube using only a straightedge and compass. In other words, say a unit cube is given (the sides of the cube are length 1). Hence, the volume of this cube is $1^3 = 1$. It is impossible to construct a cube with twice the volume of the original cube (so 2×1). Hence, the side-length of the doubled cube will equal $\sqrt[3]{2}$ which is not constructible. The equation $x^3 - 2$ has no rational roots. A cubic equation with rational coefficients and no rational root has no constructible roots. [10] Therefore, it is impossible to construct a doubled cube using only a straightedge and compass.

8 Further Work

Taking the amount of work that was completed and the amount which could have been completed given more time, there are a number of areas in which the project could be furthered.

8.1 Domain-specific language

For this project, a lot of code has been created which be used to build a language specialised to the domain of Euclidean geometric construction animations. Having a language of this sort would enable others to create construction animations much quicker and easier than someone solely using VivioJS. Also, a domain-specific language would be much more succinct and easier to understand. This improved accessibility may be beneficial to those in the educational space with less experience with programming, in particular younger students or teachers from a non-technical background.

8.2 Additional construction animations

There are a number of other animations which could be completed for this project that demonstrate further capabilities of Euclidean Geometry which were not completed within the project timespan. The following is a list of topics which could be of interest:

- Rusty compass constructions, i.e. constructions in which the compass opening can be set only once.
- Polygons with very high numbers of sides. Someone attempting to construct shapes of this sort can face many difficulties in accurately reproducing each step of the construction due to a the high level of precision; an animation which has been programmed would not face this issue.
- Compass alone or ruler alone. It was proven in 1672 by Georg Mohr that any constructions created with a straightedge and compass could also be created using just a compass or just a ruler.
- Other constructions which contain proportions that are in the golden ratio.
- Continued fractions
- Inscribed circles
- Tangent to a circle
- Construction of the area of a square
- Miscellaneous constructions which are visually appealing

8.3 Teaching Platform

As the animations are web-based, an idea might be to compile these animations on a webpage for those who wish to learn about Euclidean geometric construction. A webpage which focused on geometric constructions taught at second-level may be a good starting point. In addition, many of the constructions already featured are on secondary-level syllabi.

8.4 Further improvement on design

There were a number of alternative design choices which could have been made in creating the animations for this project. One such decision was to display the physical compass and straightedge on-screen as they are being used. This may make animations clearer to follow.

9 Conclusion

Animations of satisfactory quantity, variety and quality were produced for this project. Anecdotal feedback for the animations was good; constructions were well understood and many appreciated the clarity of each animation. Some expressed a wish that the animations had been available to them when they were in secondary school. A desired wide range of topics was covered with the majority of constructions featured being a part of second-level syllabi. In view of this, the objective of creating helpful construction animations for second-level students was met. A wide range of Euclidean Geometry concepts were touched upon which demonstrated the variety of what the system was capable of. Not only can many complex figures be constructed using only a compass and straightedge, but many topics one might not expect to come from an ancient system of mathematics can also be explored, e.g. the appearance of the golden ratio in the pentagon or how Euclidean arithmetic can be used to show how ϕ is constructible. Regarding the quantity of work completed, the project has been left in a place where one could easily continue on with the work and perhaps persue some of the tasks mentioned in Section 8.

References

[1] Tzer-Lin Chen. "Proof of the Impossibility of Trisecting an Angle with Euclidean Tools". In: *Mathematics Magazine* 39.4 (1966), pp. 239–241. ISSN: 0025570X, 19300980. URL: http://www.jstor.org/stable/2688093.

[2] *Constructible polygon*. Apr. 2018. URL: https://en.wikipedia.org/wiki/Constructible_polygon.

[3] Curriculum and Assessment Policy. *Leaving Certificate, Design and Communications Graphics Syllabus*. URL: https://www.education.ie/en/Schools-Colleges/Information/Curriculum-and-Syllabus/Senior-Cycle-/Syllabuses-and-Guidelines/lc_design_comm_sy.pdf.

[4] Curriculum and Assessment Policy. *Mathematics - Leaving Certificate - Project Maths Syllabus 2015*. Department of Education and Skills (Ireland). URL: https://www.education.ie/en/Schools-Colleges/Information/Curriculum-and-Syllabus/Senior-Cycle-/Syllabuses-and-Guidelines/.

[5] Curriculum and Assessment Policy. *The Junior Certificate - Technical Graphics Syllabus*. Department of Education and Skills (Ireland). URL: https://www.education.ie/en/Schools-Colleges/Information/Curriculum-and-Syllabus/Junior-Cycle-/Syllabuses-Guidelines/jc_technicalgraphics_sy.pdf.

[6] Department of Education et al. *Mathematics - Junior Certificate - Maths Syllabus 2015*. Department of Education and Skills (Ireland). URL: https://www.education.ie/en/Schools-Colleges/Information/Curriculum-and-Syllabus/Junior-Cycle-/Syllabuses-Guidelines/Mathematics-Junior-Certificate-Project-Maths-Syllabus-for-2015.pdf.

[7] John B. Fraleigh. *A first course in abstract algebra*. Pearson Education Limited, 2003, pp. 293–295.

[8] J. Jones. "Vivio - A System for Creating Interactive Reversible E-Learning Animations for the WWW". In: *2004 IEEE Symposium on Visual Languages - Human Centric Computing* (). DOI: 10.1109/vlhcc.2004.63. URL: https://www.scss.tcd.ie/Jeremy.Jones/vivio%205.1/documentation/vivio%20IEEE.pdf.

[9] Jeremy Jones. *VivioJS - Interactive Reversible E-Learning Animations for the WWW*. URL: https://www.scss.tcd.ie/Jeremy.Jones/VivioJS/vivio.htm.

[10] George E. Martin. *Geometric constructions*. Springer, 1998.

[11] N. J. A. Sloane and R. K. Guy. *A003401*. URL: https://oeis.org/A003401.

[12] N. J. A. Sloane and Branislav Kisacanin. *A004169*. URL: https://oeis.org/A004169.