Project Number: BMV-1007 · 46

# The History of the Calculus and the Development of Computer Algebra Systems

An Interactive Qualifying Project Report
submitted to the Faculty
of the
Worcester Polytechnic Institute
in partial fulfillment of the requirements for the
Degree of Bachelor of Science
By

Daniel Ginsburg

Brian Groose

Josh Taylor

Date: December 1997

Approved:
Professor Bogdan M. Vernescu, Advisor

On the web @ *http://www.math.wpi.edu/IQP/BVCalcHist/*

## Abstract

A technical examination of the Calculus from two directions: how the past has led to present methodologies and how present methodology has automated the methods from the past. Presented is a discussion of the mathematics and people responsible for inventing the Calculus and an introduction to the inner workings of Computer Algebra Systems.

# 0. Introduction

In most modern Calculus courses, the history behind the useful mathematical results are often left ignored. Though the pragmatic uses for Calculus are numerous, without a fundamental understanding of the origins of its methods, the student is left applying memorized techniques--often lacking an understanding of why those techniques work. It is our intent to explore the historical path, in significant mathematical detail, to the elementary methods of the Calculus.

We have grown accustomed to utilizing calculators and mathematical software to aid in solving Calculus problems. Increases in raw computational power have led to incredible savings in the amount of time required by mathematicians to perform laborious tasks. In addition, computational improvements have made solvable problems that were previously thought unsolvable. Many of the uses for computational systems in mathematics seemed obvious applications of such power. However, an area that struck us as a particularly interesting use for computers in mathematics is Computer Algebra Systems. The symbolic capabilities of programs such as Maple and Mathematica mysteriously simulate human problem-solving techniques. We will explore the methods behind such Computer Algebra Systems, and in order to truly understand the complexity of such methods, write our own system that mimics some of the elementary capabilities of these robust packages.

The project consists, then, of exploring two basic ideas. First, we believe that in order to truly understand any science, it is necessary to study the path that led to its creation. In this case, we have chosen to research the history of the mathematics from the

17th century that led to the basic methods of Calculus. In doing so, we hope to both improve our own understanding of Calculus and provide a comprehensible guide to others who wish to improve their understanding. Second, technology--specifically the computer--has had a large effect on the way mathematics is taught and used. We are taught to rely on Computer Algebra Systems and Graphing Calculators as an aid to solving problems. Rather than briefly describe the many ways that computer have influenced mathematics, we have chosen to explore one area in-depth. We will provide a discussion of the methods behind Computer Algebra Systems, a description of our own approach to writing a Computer Algebra System, and a Web accessible version of our software for those interested in seeing first-hand how the system works. It is our opinion that the best way for us to understand how computers have influenced mathematics is to ourselves develop a breed of software that has significantly influenced our own education in mathematics.

The project consists of two major portions that are intended for those with an introductory background in Calculus. The paper is written in such a way that anyone who has taken the usual introductory Calculus sequence should be able to understand the mathematics involved. In addition to examining the mathematics, we will explore the people that were pivotal in developing the Calculus. Though we believe understanding their mathematics is crucial, it is of equal importance to study the personal characteristics of the revolutionary thinkers that enhanced our understanding of nature.

The intent is to provide a look at Calculus from two directions: examining the technicalities of how the past has led to present methodologies and how present technology has automated the methods from the past.

# 1. History of the Integral from the 17th Century

## 1.1 Introduction

The path to the development of the integral is a branching one, where similar discoveries were made simultaneously by different people. The history of the technique that is currently known as integration began with attempts to find the area underneath curves. The foundations for the discovery of the integral were first laid by Cavalieri, an Italian Mathematician, in around 1635. Cavalieri's work centered around the observation that a curve can be considered to be sketched by a moving point and an area to be sketched by a moving line.

## 1.2 Cavalieri's Method of Indivisbles

In order to deal with the geometrical notion of a moving point, Cavalieri worked with what he called "indivisibles". That is, if a moving point can be considered to sketch a curve, then Cavalieri viewed the curve as the sum of its points. By this notion, each curve is made up of an infinite number of points, or "indivisibles". Likewise, the "indivisibles" that composed an area were an infinite number of lines. Though Cavalieri was not the first person to consider geometric figures in terms of the infinitesimal (Kepler had done so before him), he was the first to use such a notion in the computation of areas (Hooper 248-250).

In order to introduce Cavalieri's method, consider finding the area of a triangle. For many years, it had been known that the area of a triangle was ½ the area of a rectangle which has the same base and height.

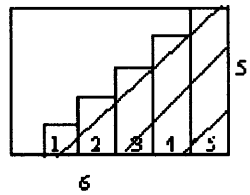

*Figure 1.1*

In Figure 1.1, the rectangle has a base of 6 units and a height of 5 units (A = bh, so the total area is 30 units). The total area of the inner rectangular regions can easily be computed by taking the sum of all the inner rectangles. Comparing the two areas:

$$\frac{\textit{Area of shaded region}}{\textit{Area of entire rectangle}} = \frac{0+1+2+3+4+5}{5*6} = \frac{15}{30} = \frac{1}{2}$$

Using the same methodology, the ratio for a larger rectangle with a greater number of inner subdivisions is computed:

$$\frac{\textit{Area of shaded region}}{\textit{Area of entire rectangle}} = \frac{0+1+2+3+...+10}{10*11} = \frac{55}{110} = \frac{1}{2}$$

The total area of the inner regions is always one-half the area of the total rectangle. This can be shown formally by using the closed form of the summation for the numerator:

$$\sum_{i=0}^{n} i = 0+1+2+...+n = \frac{n(n+1)}{2}$$

Using the closed form, it can be seen that:

$$\frac{\textit{Area of shaded region}}{\textit{Area of entire rectangle}} = \frac{\sum_{i=0}^{n} i}{n(n+1)} = \frac{\frac{1}{2}n(n+1)}{n(n+1)} = \frac{1}{2}$$

Cavalieri now took a step of great importance to the formation of the integral calculus. He utilized his notion of "indivisibles" to imagine that there were an infinite number of shaded regions. He saw that as the individual shaded regions became small enough to simply be lines, the jagged steps would gradually define a line. As the jagged steps became a line, the shaded region would form a triangle. As the number of shaded regions increases, the ratio remains simply one-half.

Cavalieri's methodology agreed with the long-held result that the area of a triangle was one-half the product of the base and height. He had also shown that his notion of "indivisibles" can be used to successfully describe the area underneath the curve. That is, as the areas of the rectangles turn into lines, their sum does indeed produce the area underneath the curve (in this case, a line). Cavalieri went on to use his method of "indivisibles" to find the area underneath many different curves. However, he was never able to formulate his techniques into a logically consistent foundation that others accepted. Though Cavalieri's techniques clearly worked, it was not until Sir John Wallis of England that the limit was formally introduced in 1656 and the foundation for the integral calculus was solidified (Hooper 249-253).

In order to fully understand Wallis' contributions to the integral calculus, it is first necessary to see how Cavalieri's theoretical techniques can be applied to find the area underneath a curve more complicated than a line. In order to do so, this technique will be applied to find the area underneath the parabola $y = x^2$.

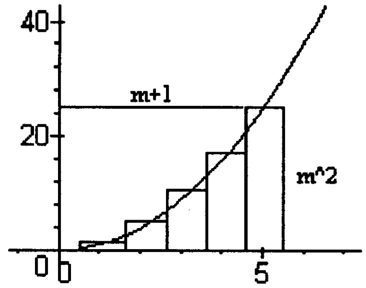

*Figure 1.2*

Each rectangular region has a base of 1 unit along the x-axis and height of $x^2$ (obtained from the definition of the parabola). The number of rectangular regions will be defined to be $m$. Cavalieri again attempted to express the area underneath the curve as the ratio of an area that was already known. He considered the area enclosing all of the $m$ rectangles. It can easily be seen from the diagram that the base of this rectangle will be $m+1$ (there are $m$ rectangles, the first starting at ½ and the last one ending at m + ½). The height of the enclosing rectangle will be $m^2$, from the definition of the parabola. The ratio can now be expressed with the following equation:

$$\frac{\textit{Total area of m rectangles}}{\textit{Area of bounding rectangle}} = \frac{1^2 + 2^2 + 3^2 + \ldots + m^2}{(m+1)m^2}$$

Recall that the area of a rectangle is defined by the product of its base and height. It was stated that the bounding rectangle had a base of $m+1$ and a height of $m^2$, which accounts for the denominator. The numerator is easily explained as well: each of the $m$ rectangles has a base of 1 and a height of its x value squared. Cavalieri now proceeded to calculate the ratio for different values of $m$. In doing so, he noticed a pattern and was able to establish a closed form for the ratio of the areas:

$$\frac{\textit{Total area of m rectangles}}{\textit{Area of bounding rectangle}} = \frac{1}{3} + \frac{1}{6m}$$

Cavalieri then utilized his important principle of "indivisibles" to make another important leap in the development of the calculus. He noticed that as he let $m$ grow larger, the term $1/6m$ had less influence on the outcome of the result. In modern terms, he noticed that

$$\lim_{m \to \infty}\left(\frac{1}{3}+\frac{1}{6m}\right)=\frac{1}{3}.$$

That is, as he lets the number of rectangles grow to infinity, the ratio of the areas will become closer to $\frac{1}{3}$. Though Cavalieri did not formally introduce the notation for limits, he did utilize the idea in the computation of areas. After using the concept of infinity to describe the ratios of the area, he was able to derive an algebraic expression for the area underneath the parabola. For at any distance x along the x-axis, the height of the parabola would be $x^2$. Therefore, the area of the rectangle enclosing the rectangular subdivisions at a point x was equal to $x(x^2)$ or $x^3$. From his earlier result, the area underneath the parabola is equal to 1/3 the area of the bounding rectangle. In other words:

$$\textit{Area under } x^2 = \frac{1}{3}x^3$$

With this technique, Cavalieri had laid the fundamental building block for integration.

### 1.3 Wallis' Law for Integration of Polynomials

John Wallis' contribution to the integral calculus was to derive an algebraic law for integration that alleviated the necessity of going through such analysis for each curve. Through examining the relationship between a function and the function that describes its area (henceforth referred to as the area-function), he was able to derive an algebraic law for determining area-functions. Rather than simply present the algebraic relationship (which the reader is doubtless familiar with if (s)he has studied a minimal amount of calculus), we will perform a similar analysis as to what led Wallis to derive his law.

First, consider the graph of the function $y = k$ or $y = kx^0$:

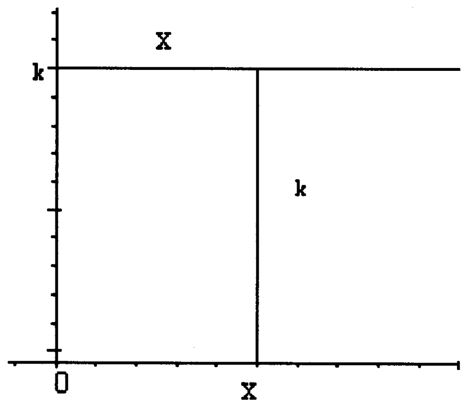


*Figure 1.3*

Clearly, it can be seen from the diagram, that the area underneath the line at any point along the x-axis will be kx or $A = \frac{1}{1}kx$.

Next, consider the graph of the function $y = kx$:

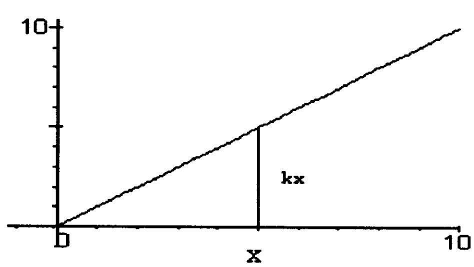


*Figure 1.4*

At any point x along the x-axis, the height will be equal to kx. Since the area forms a triangle, the area underneath the curve can be expressed as ½ the base times the height or $A = \frac{1}{2}kx^2$. As was already shown above, the area underneath a parabola $y = kx^2$, can be expressed as $A = \frac{1}{3}kx^3$. Wallis noticed an algebraic relationship between a function and its associated area-function. That is, the area-function of $y = kx^n$ is $A = \frac{1}{n+1}kx^{n+1}$. Wallis went on to show that not only does this hold true where n is a natural number (which had been the extent of Cavalieri's work), but that it also worked for negative and fractional exponents. Wallis also showed that the area underneath a polynomial composed of terms with different exponents (e.g. $y = 4x^3 + 3x^2 + x + 1$) can be computed by using his law on each of the terms independently (Hooper 255 - 260).

### 1.4 Fermat's Approach to Integration

One of the first major uses of infinite series in the development of calculus came from Pierre De Fermat's method of integration. Though previous methods of integration had used the notion of infinite lines describing an area, Fermat was the first to use infinite series in his methodology. The first step in his method involved a unique way of describing the infinite rectangles making up the area under a curve.

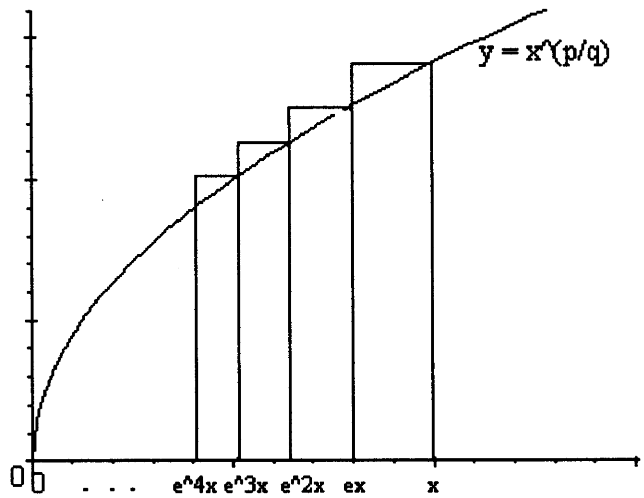


*Figure 1.5*

Fermat noticed that by dividing the area underneath a curve into successively smaller rectangles as x became closer to zero, an infinite number of such rectangles would describe the area precisely. His methodology was to choose a value 0 < e < 1, such that a rectangle was formed underneath the curve $y = x^{p/q}$ at each power of e times x (see Figure 1.5, NOTE: e was simply Fermat's choice of variable names, **not** e = 2.71828...).

Fermat then computed each area individually:

$$(x - ex)x^{p/q} = x(1-e)x^{p/q} = (1-e)x^{p+q/q}$$
$$(ex - e^2x)x^{p/q} = ex(1-e)(ex)^{p/q} = (1-e)e^{p+q/q}x^{p+q/q}$$
$$(e^2x - e^3x)x^{p/q} = e^2x(1-e)(e^2x)^{p/q} = (1-e)e^{2(p+q/q)}x^{p+q/q}$$

.

.

.

The first equation represents the area of the largest rectangle, the second equation the next rectangle to the left, and so on. The areas are simply found by multiplying the base times the height. The base is known by the power of e, and the height by evaluating $y = x^{p/q}$ at the given x value. The simplifications of each area expression are given in a form that will be useful when attempting to find the infinite sum. Fermat's next step was to compute the infinite sum of these rectangles as the power of e approached infinity.

$$
\begin{aligned}
&(1-e)x^{p+q/q} + (1-e)e^{p+q/q}x^{p+q/q} + (1-e)e^{2(p+q/q)}x^{p+q/q} + \ldots \\
&1\ term \ = (1-e)x^{p+q/q}(1) \\
&2\ terms = (1-e)x^{p+q/q}(1+e^{p+q/q}) \\
&3\ terms \ = (1-e)x^{p+q/q}(1+e^{p+q/q}+e^{2(p+q/q)}) \\
&\ldots \\
&= (1-e)x^{p+q/q}(1+e^{p+q/q}+e^{2(p+q/q)}+e^{3(p+q/q)}+\ldots.)
\end{aligned}
$$

By determining the sum of each increasing finite series, he was able to develop an expression for the infinite sum.

In order to find a closed form for the expression

$$(1+e^{p+q/q}+e^{2(p+q/q)}+e^{3(p+q/q)}+\ldots.)$$

...note that the sum is a geometric series of the form:

$$(1+x+x^2+x^3+\ldots.)$$

If $0 < x < 1$, the sum is $\frac{1}{1-x}$ (this can be shown to be true by long dividing (1-x) into 1). Therefore, by substituting $e^{p+q/q}$ back in for x and inserting into the overall equation, the area can be expressed as:

$$A = \frac{1}{1-e^{p+q/q}}(1-e)x^{p+q/q}$$

Fermat now wished to express the area entirely in terms of x, and in order to do so substituted $e = E^q$, which by simplification and factoring out (1-E):

$$A = \frac{(1-E^q)x^{p+q/q}}{1-E^{p+q}} = \frac{(1-E)(1+E+E^2+...+E^{q-1})x^{p+q/q}}{(1-E)(1+E+E^2+...+E^{p+q-1})} = \frac{(1+E+E^2+...+E^{q-1})x^{p+q/q}}{(1+E+E^2+...+E^{p+q-1})}$$

Fermat now made a step that with the benefit of current knowledge is explainable, but at that time was not properly justified. That is, Fermat said let E = 1 and since $E^q = 1^q = 1$ and because $E^q = e$ then e must also equal 1. By substituting 1 for E in the area expression above:

$$A = \frac{(1+1+1^2+...+1^{q-1})x^{p+q/q}}{(1+1+1^2+...+1^{p+q-1})} = \left(\frac{q}{p+q}\right)x^{p+q/q}$$

Although this methodology yielded the appropriate result for the area underneath the curve, Fermat's justification of letting E = 1 was not properly formulated. What he actually was doing was taking the limit as E approaches 1 and as E approaches 1 so too will e. As e approaches 1, then e raised to any power will also approach 1, and the infinite sum of the areas underneath the curve has been determined. The notion of a limit was hinted at in Fermat's work, but it was not formally defined until later (Boyer 162 - 169).

Wallis and Fermat's work had laid the groundwork for the modern concept of the integral. However, what Fermat and Wallis had failed to recognize was the relationship between the differential and the integral. That idea would be developed simultaneously by two men: Newton and Leibniz. This would later be known as the Fundamental Theorem of Calculus and, as the name implies, it is a landmark discovery in the history of the Calculus. However, before proceeding on to describe this important theorem, it is first necessary to examine the development of the differential.

# 2. History of the Differential from the $17^{th}$ Century

## 2.1 Introduction

The problem of finding the tangent to a curve has been studied by many mathematicians since Archimedes explored the question in Antiquity. The first attempt at determining the tangent to a curve that resembled the modern method of the Calculus came from Gilles Persone de Roberval during the 1630's and 1640's. At nearly the same time as Roberval was devising his method, Pierre de Fermat used the notion of maxima and the infinitesimal to find the tangent to a curve. Some credit Fermat with discovering the differential, but it was not until Leibniz and Newton rigorously defined their method of tangents that a generalized technique became accepted.

## 2.2 Roberval's Method of Tangent Lines using Instantaneous Motion

The primary idea behind Roberval's method of determining the tangent to a curve was the notion of Instantaneous Motion. That is, he considered a curve to be sketched by a moving point. If, at any point on a curve, the vectors making up the motion could be determined, then the tangent was simply the combination (sum) of those vectors.

Roberval applied this method to find the tangents to curves for which he was able to determine the constituent motion vectors at a point. For a parabola, Roberval was able to determine such motion vectors.

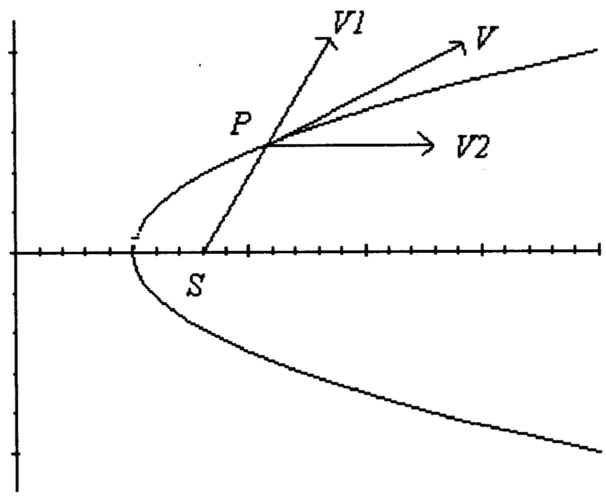


Figure 2.1

Figure 2.1 depicts the graph of a parabola showing the constituent motion vectors V1 and V2 at a point P. Roberval determined that at a point P in a parabola, there are two vectors accounting for its instantaneous motion. The vector V1, which is in the same direction as the line joining the focus of the parabola (point S) and the point on the parabola (point P). The other vector making up the instantaneous motion (V2) is perpendicular to the y-axis (which is the directrix, or the line perpendicular to the line bisecting the parabola). The tangent to the graph at point P is simply the vector sum V = V1 + V2.

Using this methodology, Roberval was able to find the tangents to numerous other curves including the ellipse and cycloid. However, finding the vectors describing the instantaneous motion at a point proved difficult for a large number of curves. Roberval was never able to generalize this method, and therefore exists historically only as a precursor to the method of finding tangents using infinitesimals (Edwards 133-138).

### 2.3 Fermat's Maxima and Tangent

Pierre De Fermat's method for finding a tangent was developed during the 1630's, and though never rigorously formulated, is almost exactly the method used by Newton and Leibniz. Lacking a formal concept of a limit, Fermat was unable to properly justify his work. However, by examining his techniques, it is obvious that he understood precisely the method used in differentiation today.

In order to understand Fermat's method, it is first necessary to consider his technique for finding maxima. Fermat's first documented problem in differentiation involved finding the maxima of an equation, and it is clearly this work that led to his technique for finding tangents.

The problem Fermat considered was dividing a line segment into two segments such that the product of the two new segments was a maximum.

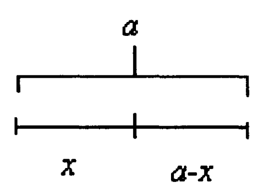


Figure 2.2

In Figure 2.2, a line segment of length $a$ is divided into two segments. Those two segments are $x$ and $(a - x)$. Fermat's goal, then, was to maximize the product $x\,(a - x)$. His approach was mysterious at the time, but with the benefit of the current knowledge of limits, Fermat's method is quite simple to understand. What Fermat did was to replace each occurrence of $x$ with $x + E$ and stated that when the maximum is found, $x$ and $x + E$ will be *equal*. Therefore, he had the equation:

$$x(a - x) = (x + E)(a - x - E)$$

Through simplifying both sides of the equation and canceling like terms, Fermat reduced it:

$$\begin{aligned} &E^2 - aE - 2xE = 0 \\ &E(E - a - 2x) = 0 \\ &E - a - 2x = 0 \end{aligned}$$

At this point, Fermat said to simply let E = 0, and as such one is left with:

$$x = \left(\frac{a}{2}\right)$$

This says that to maximize the product of the two lengths, each length should be half the total length of the line segment. Though this result is correct, Fermat's method contains mysterious holes that are only rectified by current knowledge. Fermat simply lets E = 0, then in the step where he divides through by E, he would have division by zero. However, though Fermat formulated his method by saying E = 0, he was actually considering the limit of E as it *approaches* zero (which explains why his algebra works properly). Fermat's method of extrema can be understood in modern terms as well. By substituting $x + E$ for $x$, he is saying that $f(x+E) = f(x)$, or that $f(x+E) - f(x) = 0$. Since f(x) is a polynomial, this expression will be divisible by $E$. Therefore, Fermat's method can be understood as the definition of the derivative (when used for finding extrema):

$$\lim_{E \to 0} \frac{f(x+E) - f(x)}{E} = 0$$

Although Fermat was never able to make a logically consistent formulation, his work can be interpreted as the definition of the differential (Edwards 122-125).

Using his mysterious $E$, Fermat went on to develop a method for finding tangents to curves. Consider the graph of a parabola.

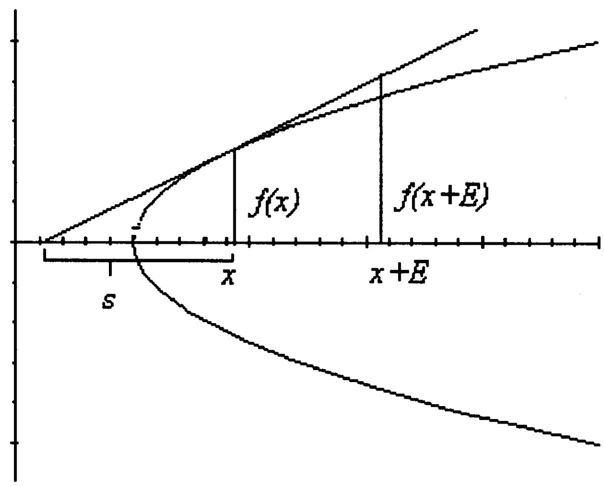


Figure 2.3

Fermat wishes to find a general formula for the tangent to $f(x)$. In order to do so, he draws the tangent line at a point $x$ and will consider a point a distance $E$ away. As can be seen from figure 2.3, by similar triangles, the following relationship exists:

$$\frac{s}{s+E}=\frac{f(x)}{f(x+E)}$$

By isolating $s$, Fermat found that

$$s=\frac{f(x)}{[f(x+E)-f(x)]/E}$$

Fermat again lets the quantity $E = 0$ (in modern term, he took the limit as $E$ approached 0) and recognized that the bottom portion of the equation was identical to his differential in his method of mimina. Consequently, in order to find the slope of a curve, all he needed to do was find f(x)/s. For example, consider the equation $f(x) = x^3$:

$$s = \frac{f(x)}{[f(x+E) - f(x)]/E} = \frac{x^3}{[(x+E)^3 - x^3]/E} = \frac{x^3}{(3x^2 + 3xE + E^2)}$$

Again, Fermat lets *E=0* and finds that:

$$s = \frac{x^3}{3x^2} = \frac{x}{3}$$

Now, returning to the original equation:

$$f'(x) = [f(x+E) - f(x)/E] = \frac{f(x)}{s} = \frac{x^3}{x/3} = 3x^2$$

Here the modern notation for the derivative *f'(x) is* used, which Fermat recognized to be equal to *[f(x+E) - f(x)]/E* when he let *E=0.* Using this method, Fermat was able to derive a general rule for the tangent to a function $y = x^n$ to be $nx^{n-1}$. As described in the Integration section, Fermat had now developed a general rule for polynomial differentiation and integration. However, he never managed to see the inverse relationship between the two operations, and the logical inconsistencies in his justification left his work fairly unrecognized. It was not until Newton and Leibniz that this formulation became possible (Boyer 155-159).

### 2.4 Newton and Leibniz

Newton and Leibniz served to complete three major necessities in the development of the Calculus. First, though differentiation and integration techniques had already been researched, they were the first to explain an "algorithmic process" for each operation.

Second, despite the fact that differentiation and integration had already been discovered by Fermat, Newton and Leibniz recognized their usefulness as a general process. That is, those before Newton and Leibniz had considered solutions to area and tangent problems as specific solutions to particular problems. No one before them recognized the usefulness of the Calculus as a general mathematical tool. Third, though a recognition of differentiation and integration being inverse processes had occurred in earlier work, Newton and Leibniz were the first to explicitly pronounce and rigorously prove it (Dubbey 53-54).

Newton and Leibniz both approached the Calculus with different notations and different methodologies. The two men spent the latter part of their life in a dispute over who was responsible for inventing the Calculus and accusing each other of plagiarism. Though the names Newton and Leibniz are associated with the invention of the Calculus, it is clear that the fundamental development had already been forged by others. Though generalizing the techniques and explicitly showing the Fundamental Theorem of Calculus was no small feat, the mathematics involved in their methods are similar to those who came before them. Sufficiently similar are their methods that the specifics of their methodologies are beyond the scope of this paper. In terms of their mathematics, it is only their demonstration of the Fundamental Theorem of Calculus that will be discussed.

### 2.5 The Ellusive Inverses – the Integral and Differential

The notation of Leibniz most closely resembles that which is used in modern calculus and his approach to discovering the inverse relationship between the integral and differential will be examined. Though Newton independently arrived at the same conclusion, his path to discovery is slightly less accessible to the modern reader.

Leibniz defined the differential as being

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x}$$

From the earlier works of Cavalieri, Leibniz was already familiar with the techniques of finding the area underneath a curve. Leibniz discovered the inverse relationship between the area and derivative by utilizing his definition of the differential.

Consider the graph of the equation $y = x^2+1$:

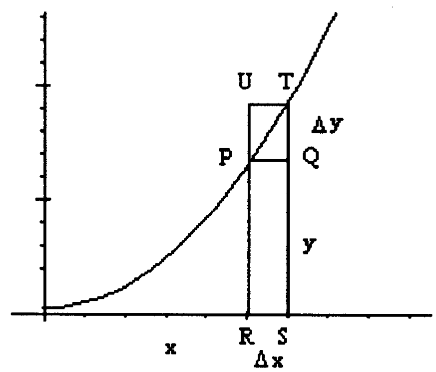


*Figure 2.4*

Leibniz's idea was to use his differential on the area-function of the graph. Consider adding a Δ(area) underneath the graph of the curve. The Δ(area) is defined by the lower rectangle PQRS with area is y(Δx) plus a fraction of the upper rectangle SRUT whose area is simply Δx(Δy). In other words, Δ(area) lies somewhere in between y(Δx) and the total enclosing rectangle PQUT whose area is (y + Δy )(Δx). Leibniz then considered the ratio Δ(area)/ Δx and saw that since the Δ(area) is between y(Δx) and (y + Δy)(Δx) the ratio will be between y and (y + Δy). From the diagram, it can be seen that Δx and Δy are closely related to each other. That is, as Δx approaches 0 so too does Δy. That means that the ratio Δ(area)/ Δx lies between y and a value that approaches y (since y + Δy approaches y as Δy goes to 0). Written in terms of Leibniz's definition of the derivative:

$$\frac{dy}{dx} = \lim_{\Delta x \to 0} \frac{\Delta area}{\Delta x} = y = x^2 + 1$$

Leibniz has shown the inverse relationship between the differential and the area-function. Namely that the differential of the area-function of a function $y$ is equal to the function itself. In this case, the derivative of the area-function of $y = x^2+1$ is indeed $y = x^2+1$.

Leibniz's influence in the history of the integral spreads beyond finding this groundbreaking relationship. He was also responsible for inventing the notation that is used by most students of calculus today. Leibniz used the symbol ∫ (which was simply how "S" was written at the time) to denote an infinite number of sums. This was closely related to what he called the "integral", or the sum of a number of infinitely small areas. The area underneath a function *y,* or integral of *y,* was expressed as ∫ y (dx).

What Leibniz's notation was really saying was to sum up all of the areas dx * y as dx approached 0. As dx approaches 0, there are an infinite number of such areas, hence the symbolism ∫ representing an infinite number of sums. Integration of this kind is also known as the *indefinite integral* or *anti-derivative* due to the inverse relationship found by Leibniz. That is, the derivative of the indefinite integral of a function yields the function itself. . Leibniz also developed a notation for *definite integrals*, or integrals which produced the area underneath a curve between two bounding values (rather than a symbolic answer). His notation for the *definite integral* was to supply the lower and upper-bounding x-values with the integral symbol:

$$\int_a^b f(x)dx = A(b) - A(a)$$

Where A is the area-function produced by the *anti-derivative*. The area function A was computed by using Wallis' law.

## 3. Selected Problems from the History of the Infinite Series

### 3.1 Introduction

Mathematicians have been intrigued by Infinite Series ever since antiquity. The question of how an infinite sum of positive terms can yield a finite result was viewed both as a deep philosophical challenge and an important gap in the understanding of infinity. Infinite Series were used throughout the development of the calculus and it is thus difficult to trace their exact historical path. However, there were several problems that involved infinite series that were of significant historical importance. This section contains selected problems that represent an introduction to the historical significance of the Infinite Series.

### 3.2 James Gregory's Infinite Series for arctan

Most of Gregory's work was expressed geometrically, and was difficult to follow. He had all the fundamental elements needed to develop calculus by the end of 1668, but lacked a rigorous formulation of his ideas. The discovery of the infinite series for arctan $x$ is attributed to James Gregory, though he also discovered the series for tan $x$ and sec $x$. Here is how one can find the derivative of arctan $x$:

$$
\begin{aligned}
&y = \arctan x \\
&\tan y = \tan(\arctan x) \\
&\tan y = x \\
&\frac{dy}{dx}\sec^2 y = 1 \\
&\frac{dy}{dx} = \frac{1}{\sec^2 y} = \frac{1}{\tan^2 y + 1} \\
&\frac{dy}{dx} = \frac{1}{x^2+1} = \frac{1}{1+x^2}
\end{aligned}
$$

The above is a modern proof, Gregory used the derivative of arctan from the work of others. The infinite series for $\frac{1}{1+x^2}$ can be found by using long division.

$$\frac{1}{1+x^2} = 1 - x^2 + x^4 - x^6 + \ldots$$

Integrating this infinite series term-by-term produces,

$$\arctan x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \ldots$$

which is the infinite series for arctan.

Prior to Leibniz and Newton's formulation of the formal methods of the calculus, Gregory already had a solid understanding of the differential and integral, which is shown here. Although the solution above is in modern notation, Gregory was able to solve this problem with his own methods. Gregory was one of the first to relate trigonometric functions to their infinite series using calculus, although he is primarily only remembered noted for finding the infinite series for the inverse tangent. (Boyer 429)

### 3.3 Leibniz's Early Infinite Series

One of Leibniz's earlier experiences with infinite series was to find the sum of the reciprocals of the triangular numbers, or $\frac{2}{n(n+1)}$. By using partial fraction decomposition, the fraction can be split so that $\frac{2}{n(n+1)} = 2\left(\frac{1}{n} - \frac{1}{n+1}\right)$. The first n terms of the series are:

$$2\left(\frac{1}{1} - \frac{1}{1+1}\right) + 2\left(\frac{1}{2} - \frac{1}{2+1}\right) + 2\left(\frac{1}{3} - \frac{1}{3+1}\right) + \ldots 2\left(\frac{1}{n} - \frac{1}{n+1}\right)$$

By factoring out the 2 and by rearranging the terms:

$$2\left(\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\ldots+\frac{1}{n}-\frac{1}{1+1}-\frac{1}{2+1}-\frac{1}{3+1}-\ldots-\frac{1}{n+1}\right)$$

all but the first and last term cancel, and the sum reduces to $2\left(\frac{1}{1}-\frac{1}{n+1}\right)$ since

$\lim_{n\to\infty} 2\left(\frac{1}{1}-\frac{1}{n+1}\right)=2$. Therefore, the sum of the reciprocals of the triangular numbers is 2.

This problem was historically significant as it served as in inspiration for Leibniz to explore many more infinite series. Since he successfully solved this problem, he concluded that a sum could be found of almost any infinite series. (Boyer, 446-447)

### 3.4 Leibniz and the Infinite Series for Trigonometric Functions

After having already developed methods for differentiation and integration, Leibniz was able to find an infinite series for sin(z) and cos(z). He began the process by starting with the equation for a unit circle:

$$x^2+y^2=1 \quad \textit{where}\ x=\cos\Theta\ \ y=\sin\Theta$$

and differentiating with respect to x:

$$\frac{d\Theta}{dx}=-\frac{1}{\sin\Theta}$$
$$d\Theta=-\frac{1}{\sin\Theta}dx$$

By the equation of the unit circle given above $\sin \Theta = \sqrt{(1-\cos^2 \Theta)}$ and $\cos\Theta = x$

so

$$d\Theta = -\frac{1}{\sqrt{1-x^2}}dx$$

Prior to Leibniz attempting to solve this problem, Newton had discovered the binomial theorem. Therefore, by simple application of Newton's rule, Leibniz was able to expand the equation into an infinite series:

$$-\frac{1}{\sqrt{1-x^2}}dx = -(1+\frac{x^2}{2}+\frac{3x^4}{8}+\frac{5x^6}{16}+...)dx$$

Leibniz then integrated both sides. The right side of the equation can be integrated term-by-term and the left side of the equation is equal to arcsin(x). This can easily be shown:

$$\begin{aligned}
&y = \arcsin x \\
&\sin y = \sin(\arcsin x) \\
&\sin y = x \\
&\cos y \frac{dy}{dx} = 1 \\
&\frac{dy}{dx} = \frac{1}{\cos y} = \frac{1}{\sqrt{1-\sin^2 y}} = \frac{1}{\sqrt{1-x^2}}
\end{aligned}$$

Therefore, integrating both sides yields:

$$\arcsin x = x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + ...$$

At this point, Leibniz had found the infinite series for arcsin(x), a result which Newton had found as well. Leibniz then used a process he and Newton both discovered independently: Series Reversion. That is, given the infinite series for a function, he found a way to calculate the infinite series for the inverse function.

In this case, the process worked by first taking the sin (the inverse function for arcsin) of both sides of the equation:

$$y = \arcsin x = x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + \ldots$$
$$\sin y = \sin(\arcsin x) = \sin(x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + \ldots)$$
$$\sin y = x = \sin(x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + \ldots)$$

Now Leibniz assumed that an infinite series for sin(y) exists that is of the form:

$$\sin y = a_1 y^1 + a_2 y^2 + a_3 y^3 + \ldots + a_n y^n + \ldots$$

Leibniz had said that sin y = x, therefore for each instance of x in $\sin(x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + \ldots)$ he substituted the assumed infinite series for $\sin y$. He knew that the result of substituting in the this series for x must yield y, as it was stated:

$$\sin y = \sin(x + \frac{x^3}{6} + \frac{3x^5}{40} + \frac{5x^7}{112} + \ldots)$$

Therefore, he knew the coefficient of the first term $a_1$=1 and all of the other coefficients must add up to 0. In order to further explain, the first 3 coefficients of the expansion will be solved for. When the series is substituted, the only possible way to have a $y^1$ term is when it is substituted for x. The first term in the expansion will therefore be $(a_1 1)y^1$. There will be no $y^2$ term, and the $y^3$ term will be obtained by both the 3rd power y term being plugged into x, and the 1st power y term being plugged into $x^3$ thus yielding $(a_1{}^3 \frac{1}{6} y^3 + a_3 1 y^3)$ where the sum of the coefficients must be 0 (because there is no $y^3$ term left over in the expansion). The same process yields the equation for the 5th order term which is $(a_5 1 y^5 + a_1^{\,5}\frac{3}{40}y^5 + 3a_1^{\,2}a_3\frac{1}{6}y^5)$. At this point the resulting expansion is:

$$(a_1 1)y + (a_1^{\,3}\frac{1}{6} + a_3 1)y^3 + (a_5 1 + a_1^{\,5}\frac{3}{40} + 3a_1^{\,2}a_3\frac{1}{6})y^5 + \ldots = y$$

Now equations for each coefficient can be set up:

$$a_1 = 1$$
$$a_1^{\,3}\frac{1}{6} + a_3 1 = 0$$
$$a_5 1 + a_1^{\,5}\frac{3}{40} + 3a_1^{\,2}a_3\frac{1}{6} = 0$$

Solving the equations using the previous results in each calculation yields:

$$a_1 = 1 = \frac{1}{1!}$$
$$a_3 = -\frac{1}{6} = -\frac{1}{3!}$$
$$a_5 = \frac{1}{120} = \frac{1}{5!}$$

Substituting the coefficients back into the assumed infinite series for $\sin y$, he determined that:

$$\sin y = y - \frac{1}{3!}y^3 + \frac{1}{5!}y^5 + \ldots$$

By simply differentiating this equation term-by-term Leibniz was also able to find the infinite series for $\cos y$ (Boyer and Merzbach 448 - 449).

Leibniz not only laid the groundwork for the Taylor series, but he (and simultaneously Newton) was the first to discover the series for these trigonometric functions. He invented his own method for finding the infinite series of a function's inverse. For a more thorough description of the process of Series Reversion please refer to *Eric's Treasure Trove of Mathematics* on the Web, the URL can be found in the Bibliography.

### 3.5 Euler's Sum of the Reciprocals of the Squares of the Natural Numbers

Much work was done with infinite series by Euler. He was able to use infinite series to solve problems that other mathematicians were not able to solve by any methods. Neither Leibniz nor Jacques Bernoulli were able to find the sum of the inverse of the squares - they even admitted as much. The sum was unknown until Euler found it through the manipulation of an infinite series:

$$\frac{1}{1^2}+\frac{1}{2^2}+\frac{1}{3^2}+\ldots$$

In order to find this sum, Euler started by examining the infinite series for sin $z$.

$$\sin z = z-\frac{z^3}{3!}+\frac{z^5}{5!}-\frac{z^7}{7!}+\ldots$$

Equating sin $z$ to zero gave Euler the roots of the infinite expansion

$$\sin z = 0 = z-\frac{z^3}{3!}+\frac{z^5}{5!}-\frac{z^7}{7!}+\ldots.$$

That is, the roots of this equation are $z=\pi,2\pi,3\pi,4\pi\ldots$ Now, left with the equation

$$0 = z-\frac{z^3}{3!}+\frac{z^5}{5!}-\frac{z^7}{7!}+\ldots \text{ with roots } z=\pi,2\pi,3\pi,4\pi\ldots$$

dividing by $z$ results in

$$0 = 1-\frac{z^2}{3!}+\frac{z^4}{5!}-\frac{z^6}{7!}+\ldots$$

substituting $z^2 = w$ yields

$$0 = 1 - \frac{w}{3!} + \frac{w^2}{5!} - \frac{w^3}{7!} + \ldots \text{ and } w = (\pi)^2, (2\pi)^2, (3\pi)^2, (4\pi)^2 \ldots$$

By using properties involving polynomials, it is known that the sum of the reciprocals of the roots is the negative of the coefficient of the linear term, assuming the constant term is 1. Applying this here, we get

$$\frac{1}{6} = \frac{1}{(\pi)^2} + \frac{1}{(2\pi)^2} + \frac{1}{(3\pi)^2} + \frac{1}{(4\pi)^2} + \ldots$$

multiplying through by $\pi^2$, we get

$$\frac{\pi^2}{6} = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots$$

Which is the sum of the inverse of the squares. Starting with $\cos x$ instead of $\sin x$, he obtained the sum for the sum of the squares of the odd natural numbers. He solved problems using infinite series that could not be done in any other way, and developed new ways to manipulate them. (Boyer 496-497)

# 4. Short Biographies

## 4.1 Introduction

The following chapter is meant to provide brief biographies of the mathematicians that significantly contributed to the development of the Calculus.

## 4.2 Gregory of St. Vincent (1584-1667)

A Jesuit teacher in Rome and Prague, he later became a tutor in the court of Philip IV of Spain. He tried to "square the circle" (constructing a square equal in area to circle using only a straight edge and compass) throughout his life, and discovered several interesting theorems while doing so. He discovered the expansion for $\log(1 + x)$ for ascending powers of $x$. Eventually, he thought he had squared the circle, but his method turned out to be equivalent to the modern method of integration. He successfully integrated $x^{-1}$ in a geometric form which is equivalent to the natural logarithm function.

## 4.3 Rene Descartes (1596-1650)

Rene Descartes was a philosopher of great acclaim. The idea that humans may make mistakes in reasoning is the foundation of his philosophy. He cast aside all traditional beliefs and tried to build his philosophy from the ground up, based on his reasoning alone. In his search for a base on which he might begin to build his reconstructed view of the world, he doubted the reality of his own existence. The existence of his doubt persuaded him to formulate the famous maxim, “I think, therefore I am.”

The precision and clarity of mathematics and mathematical reasoning impressed Descartes. He hoped to make use of it in the development of his philosophy and thereby reduce the susceptibility to flaws of his own reasoning. He spent a number of years studying mathematics and developing systematic methods for distinguishing between truth and falsehood. His contribution to the field of geometry can be thought of as an example of how his methods can be applied to reveal new truths, but to the mathematicians to follow him, Descartes' analytical geometry was powerful tool in its own right. Descartes' work in geometry laid the foundation for the calculus that was to come after him.

Due to his wariness of mistakes in reasoning, Descartes' tended to de-emphasize his formal education and instead focus on learning by first-hand experience. His philosophy of experience led him to travel outside of his native France, serve in the military, and eventually live in Holland. During his time in Holland, Descartes tutored Princess Elisabeth, but devoted most of his time to contemplation of his philosophy and his writing. He was summoned to Sweden in 1646 to tutor Queen Christine, but the Swedish winters were too difficult for him and Descartes died in 1650.

### 4.4 Bonaventura Cavalieri (1598-1647)

Cavalieri became a Jesuate (not a Jesuit as is frequently stated) at an early age and it was because of that that he was made a professor of mathematics at Bologna in 1629. He held the position until his death in 1647. Cavalieri published tables of sines, tangents, secants, and versed sines along with their logarithms out to eight decimal places, but his most well known contribution is in the invention of the principle of indivisibles. His principle of indivisibles, developed by 1629, was first published in 1635 and was again published in 1653, after his death, with some corrections. The principle of indivisibles is based on the assumption that any line can be divided up into an infinite number of points, each having no length, a surface may be divided into an infinite number of lines, and a volume can be divided into an infinite number of surfaces.

### 4.5 Pierre de Fermat (1601-1665)

Pierre de Fermat was born in France, near Montauban, in 1601, and he died at Castres on January 12, 1665. Fermat was the son of a leather merchant, and he was educated at home. He became a councilor for the local parliament at Toulouse in 1631, a job where he spent the rest of his life. Fermat's life, except for a dispute with Descartes, was peaceful and unremarkable. The field of mathematics was a hobby for Fermat. He did not publish much during his lifetime regarding his findings. Some of his most important contributions to mathematics were found after his death, written in the margins of works he had read or contained within his notes. He did not seem to intend for any of his work to be published, for he rarely gave any proof with his notes of his discoveries. Pierre de Fermat's interests were focused in three areas of mathematics: the theory of numbers, the use of geometry of analysis and infinitesimals, and probability. Math was a hobby for Fermat – his real job was as a judge. Judges of the day were expected to be aloof (so as to resist bribery), so he had a lot of time for his hobby.

### 4.6 Gilles Persone de Roberval (1602-1675)

Held chair of Ramus at the Collège Royale for 40 years from 1634. He developed a method of indivisibles similar to that of Cavalieri, but did not disclose it. Roberval became involved in a number of disputes about priority and credit; the worst of these concerned cycloids. He developed a method to find the area under a cycloid. Some of his more useful discoveries were computing the definite integral of sin $x$, drawing the tangent to a curve, and computing the arc length of a spiral. Roberval called a cycloid a trochoid, which is Greek for wheel.

### 4.7 John Wallis (1616-1703)

A professor of geometry at Oxford, he had several very important publications, which advanced the field of indivisibles. He studied the works of Cavalieri, Descartes, Kepler, Roberval, and Torricelli. He introduced ideas in calculus that went beyond those he read of. He discovered methods to evaluate integrals that were later used by Newton in his work on the binomial theorem. Wallis was the first to use the modern symbol for infinity. It is interesting to note that Wallis rejected the idea that negative numbers were less than nothing but accepted the notion that they were greater than infinity.

### 4.8 Blaise Pascal (1623-1662)

Pascal was a French student of Desargues. Etienne Pascal, Blaise's father, kept him away from mathematical texts early in his life until Blaise was twelve, when he studied geometry on his own. After this, Etienne, himself a mathematician, urged Blaise to study. Pascal invented an adding machine, to aid in his father's job as a tax collector. Its development was hindered by the units of currency used in France and England at the time. Two hundred and forty deniers equaled one livre, which is a difficult ratio for conversion.

Pascal turned to religion at the age of twenty-seven, ceasing to work on any mathematical problems. When he had to administer his father's estate for a time, he returned to studying the pressure of gasses and liquids, which got him into many arguments because he believed that there is a vacuum above the atmosphere; an unpopular belief at the time. During this time he also founded the theory of probability with Fermat. Late in his life, he turned to the study of the cycloid when he had a toothache. The tooth ache went away immediately upon pondering a cycloid, and he took this as a sign to study more on the subject of cycloids.

In Pascal's *Pensées*, one of his large religious papers, Pascal made a famous statement known as Pascal's Wager: "If God does not exist, one will lose nothing by believing in him, while if he does exist, one will lose everything by not believing." His conclusion was that "...we are compelled to gamble..."

**4.9 Christiaan Huygens (1629-1695)**

Descartes took interest in Huygens at an early age and influenced his mathematical education. He developed new methods of grinding and polishing telescope lenses, and using a lens he made, he was able to see the first moon of Saturn. He was the one to discover the shape of the rings around Saturn using his improved telescopes. Huygens patented the first pendulum clock, which was able to keep more accurate time than current clocks because he needed a way to keep more accurate time for his astronomical observations. He was elected to the Royal Society of London and also to the Académie Royale des Sciences in France. Leibniz was a frequent visitor to the Académie and learned much of his mathematics from Huygens. Throughout his life he worked on pendulum clocks to determine longitude at sea.

In one of his books, he describes the descent of heavy bodies in a vacuum in which he shows that the *cycloid* is the *tautochrone*, which means it is the shortest path. He also shows that the force on a body moving in a circle of radius $r$ with a constant velocity of $v$ varies directly as $v^2$ and inversely as $r$.

### 4.10 Isaac Barrow (1630-1677)

An Englishman, he was ordained and later made a professor of geometry at Gresham College in London. Barrow developed a method for determining tangents that closely approached the methods of calculus. He was also the first to discover that differentiation and integration were inverse operations. He thought that algebra should be part of logic instead of mathematics, which hindered his search for analytic discoveries. Barrow published a method for finding tangents, which turned out to be an improvement on Fermat's method of tangents. He worked with Cavalieri, Huygens, Gregory of St. Vincent, James Gregory, Wallis, and Newton.

### 4.11 James Gregory (1638-1675)

A Scotsman, he was familiar with the mathematics of several countries. Gregory worked with infinite series expansion, and infinite processes in general. He sought to prove, through infinite processes, that one could not square the circle, but Huygens, who was regarded as the leading mathematician of the day, believed that pi could be expressed algebraically, and many questioned the validity of Gregory's methods. Two hundred years later, it was proved that Gregory was right.

Much of his work was expressed in geometric terms, which was more difficult to follow than if it has been expressed algebraically. Because of this, Newton was the first to invent Calculus, even though Gregory knew all the important elements of Calculus, they were not expressed in a form that was easily understandable. Gregory only has the infinite series for arctangent attributed to him, even though he also discovered the infinite series for tangent, arcsecant, cosine, arccosine, sine, and arcsine. Using his infinite series for arctangent, he was able to find an expansion for $\pi/4$ several years before Leibniz.

**4.12 Sir Isaac Newton (1642 - 1727)**

Newton's father was a farmer, and it was intended that he follow in the family business. Instead of running the farm, an uncle decided that he should attend college, specifically Trinity College in Cambridge, where the uncle has attended college. Newton's original objective was to obtain a law degree. He attended Barrow's lectures and originally studied geometry only as a means to understand astronomy. In 1665, Trinity College closed down because of the plague in England. During the year it was closed he made several important discoveries. He developed the foundation for his integral and differential calculus, his universal theory of gravitation and also some theories about color. Upon his return to Trinity, Barrow resigned the Lucasian chair in 1669, and recommended Newton for the position. He continued to work on optics and mathematical problems until 1693, when he had a nervous breakdown. He took up a government position in London, ceasing all research. In 1708, Newton was knighted by Queen Anne; he was honored for all his scientific work. He was elected president to the Royal Society in 1703 and held the position until his death in 1727.

### 4.13 Gottfried Wilhelm Leibniz (1646 - 1716)

Leibniz was a law student at the University of Leipzig and the University of Altdorf. In 1672, he traveled to Paris in order to try and dissuade Louis XIV from attacking German areas. He stayed in Paris until 1676. During this time he continued to study law but also studied physics and mathematics under Huygens. During this time, he developed the basic version of his calculus. In 1676, he moved to Hanover, where he spent the rest of his life. In one of his manuscripts dated November 21, 1675, he used the current-day notation for the integral, and also gave the product rule for differentiation. By 1676, he had discovered the power rule for both integral and fractional powers. In 1684 he published a paper containing the now-common *d* notation, the rules for computing derivatives of powers, products and quotients. One year later, Newton published his *Principia*. Because Newton's work was published after Leibniz's, there was a great dispute over who discovered the theories of calculus first which went on past their deaths.

### 4.14 Leonhard Euler (1707 - 1783)

Euler was the son of a Lutheran minister and was educated in his native town under the direction of John Bernoulli. He formed a life-long friendship with John Bernoulli's sons, Daniel and Nicholas. Euler went to the St. Petersburg Academy of Science in Russia with Daniel Bernoulli at the invitation of the empress. The harsh climate in Russia affected his eyesight; he lost the use of one eye completely in 1735. In 1741, Euler moved to Berlin at the command of Frederick the Great. While in Berlin, he wrote over 200 articles and three books on mathematical analysis. Euler did not get along well with Frederick the Great, however, and he returned to Russia in 1766. Within three years, he had become totally blind. Even though he was blind, he continued his work and published even more works. Euler produced a total of 886 books and papers through his life. After he died, the St. Petersburg Academy continued to publish his unpublished papers for 50 years. Euler used the notations f(x), i for the square root of -1, $\pi$ for pi, $\Sigma$ for summation, and e for the base of a natural logarithm. Euler died in 1783 of apoplexy.

# 5. Computer Algebra Systems

## 5.1 Introduction - What is a Computer Algebra System?

A Computer Algebra system is a type of software package that is used in manipulation of mathematical formulae. The primary goal of a Computer Algebra system is to automate tedious and sometimes difficult algebraic manipulation tasks. The principal difference between a Computer Algebra system and a traditional calculator is the ability to deal with equations symbolically rather than numerically. The specific uses and capabilities of these systems vary greatly from one system to another, yet the purpose remains the same: manipulation of symbolic equations. Computer Algebra systems often include facilities for graphing equations and provide a programming language for the user to define his/her own procedures.

Computer Algebra systems have not only changed how mathematics is taught at many universities, but have provided a flexible tool for mathematicians worldwide. Examples of popular systems include Maple, Mathematica, and MathCAD. Computer Algebra systems can be used to simplify rational functions, factor polynomials, find the solutions to a system of equation, and various other manipulations. In Calculus, they can be used to find the limit of, symbolically integrate, and differentiate arbitrary equations.

Attempting to expand the equation

$$(x-100)^{1000}$$

using the binomial theorem by hand would be a daunting task, nearly impossible to do without error. However, with the aid of Maple, this equation can be expanded in less than two seconds. Differentiating the result term-by-term can then be performed in milliseconds. The usefulness of such a system is obvious: not only does it act as a time saving device, but problems which simply were not reasonable to perform by hand can be performed in seconds.

Leibniz and Newton developed calculus in terms of algorithmic processes. Computer Algebra systems can now take these methods and remove the human from the process. However, in studying Calculus and even simple algebraic operations, it would seem that computers would be extraordinarily inept at performing such tasks. After all, most of us consider there to be a great deal of problem solving involved in the mathematics taught in grade school and beyond. How is it that a computer, a mindless composition of binary digits, is able to perform such complex tasks? It would seem that the computer would be unsuitable for such tasks, but the success of popular Algebra software packages show that this is not the case. On the contrary, Computer Algebra systems often know how to perform more operations on equations than the user!

Rather than discuss the many ways that Computer Algebra systems have altered the education and use of Calculus, we were most intrigued by how these systems actually worked. Our approach was to begin by researching the theories and issues involved in creating a Computer Algebra system. Coinciding with our research, we began writing our own Computer Algebra system in C++. The rest of this section is dedicated to a summary of our research and the specifics of the implementation we chose.

## 5.2 Data Structures

### 5.2.1 Introduction

In order for a computer program to even begin manipulating a symbolic equation, it first must store that equation somewhere in memory. At the heart of any Computer Algebra system is a data structure (or combination of data structures) responsible for describing a mathematical equation. Equations can exist in several variables, contain references to other functions, and can themselves be rational functions. There is no perfect solution to a data structure representation of an equation. One representation might be efficient for certain mathematical operations, but poor for others. Another representation might be inefficient in time and space complexity, but easy to program. Such tradeoffs need to be considered when choosing a representation; there is no absolute answer to the problem.

### 5.2.2 Polynomials in one variable - Coefficients

In order to begin a discussion of the issues involved in storing a symbolic equation, polynomials of single-variables will be considered. That is, equations that are only of the form $f(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_0 x^0$ where $a_k$ is an integer or fractional coefficient. Even in storing such a simple equation, there are numerous issues involved in choosing a data structure.

The coefficient itself can not be stored as a simple data type. The amount of storage for an integer in most languages is typically 16 or 32-bits. In a 32-bit representation of an integer, only $2^{32}$ or approximately 4.3 billion different numbers can be represented. Though this might seem large, for many mathematical operations (such as the simplification described in the Introduction), numbers of much larger size must be possible to represent. Therefore, a numeric data type that allows for expansive growth in representation (e.g. a data type that grows dynamically with the size of the number) needs to be used. For fractional coefficients, simply storing the numerator and denominator separately in two such data types is adequate.

**5.2.3 Polynomials in one variable - Terms**

Having dealt with the issue of storing coefficients, the more significant problem of how to actually store each term must be dealt with. The first issue to contend with is that of finding a canonical form. That is, consider that a user types in the following single-variable equations:

$x^2 - x$ and $-x + x^2$

$(x+2)(x-2)$ and $x^2 - 4$

In both instances, a human can easily expand and re-order the equations to determine equivalence. However, for a computer, this is far from a trivial task. The Computer Algebra system must represent these equations in a canonical form, one in which only one representation exists for each equation. That is, in the above examples, the Computer Algebra system would simplify both pairs of equation into the same representation in internal memory.

For a single-variable polynomial, once fully simplified (more on this later), finding a canonical form is not difficult. In the internal representation, simply sort the terms by degree of their exponent, and each version of the equation will correspond to the same representation. The next step is to determine a way to store each term in the computer's memory. One approach would be to create an array of the size equal to the largest exponent in the equation. An array is simply a collection of items of the same type, the size of which does not change after instantiation. For example, consider a user enters the equation $x^5 + 2x + 1$. The program would create an array of 6 elements. At each element in the array, it would store the coefficient of the corresponding term (and place a 0 in all the unused terms):

| 0 | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 2 | 0 | 0 | 0 | 1 |

Figure 5.1

This representation is known as dense because it stores each of the terms, independent of whether the coefficient of the term is 0. An alternative representation would be to create a list or similar data structure that at each node stores the coefficient and the exponent of the term. This way, only the terms that actually are used require storage in memory:

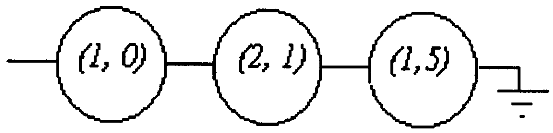


Figure 5.2

At each node in the list, the first number in the pair represents the coefficient and the second the exponent of each term. This representation is sparse: only the terms which have non-zero coefficients are stored in memory.

Typically, a sparse representation is preferable to a dense one, but there are advantages and disadvantages to both. In a dense representation, there often can be large amounts of computer memory wasted. For example, the equation $x^{2000} + x$ would require an array of 2001 elements just to store two coefficients. Comparatively, the sparse version would require only two nodes in a list. Additionally, from the programmer perspective, it is difficult to make changes to a dense representation. For example, if one were to add a new term to a polynomial that was not included in the range of the original array, the array would have to be completely recreated (in most languages, the size of an array can not be modified without completely recreating it). The sparse representation provides much more flexibility, as adding a new term is simply a matter of adding a new node to the list in the correct place (to maintain a canonical form). The choice between a sparse and dense representation is completely dependent on the task for which it will be used. In some cases, a system will shift between representations in order to optimize for specific algorithms (Davenport 59-70).

### 5.2.4 Polynomials in one variable - Recursive definition

A Computer Algebra system supporting only equations of the form $f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_0 x^0$ would be quite inflexible. One enhancement, while still remaining in the realm of polynomials of one variable, is to allow equations with recursive definitions. That is, polynomials where the coefficient to a term or numerous terms can be a polynomial itself. Some examples of such equations would be $(x+1)(x+2)$, $(3x^2+3x)x$, and $[(3x \ \ +1)2x]x^5$. The sparse representation can be easily modified to support equations of such a form. Instead of simply storing pairs of coefficients and their exponents, the coefficients themselves can also point to lists of polynomials. In order to find the canonical form, the polynomial is simplified and all recursive polynomial coefficients are removed from the representation. In order to support rational functions, all that is needed is to store two polynomials: the numerator and denominator. Of course, simplification of a rational function into canonical form introduces a host of new issues, but these issues will be discussed later.

### 5.2.5 Multivariate Polynomials

The sparse representation can be extended into multivariate polynomials without too much effort in terms of representation. Rather than storing simply a coefficient and degree at each node in the list, the degree will be replaced with a list of variables and their respective degrees. The difficult issue that arises when switching to a multivariate representation is finding a canonical form. One approach that is commonly used is to first sort the terms lexicographically and then by degree. As an example, consider the equation

$$y^2 + z + xy + x^2 z + yz + yz^2$$

would be represented in the following manner after sorting

$$xy + x^2 z + y^2 + yz^2 + yz + z$$

The selection of sorting is irrelevant (whether first by lexicographic order and then degree or vice versa) as long as the choice is consistent (Davenport 71-74).

### 5.2.6 The Syntax Tree - Our Data Structure Implementation

In the spirit of sparse representation, we chose to use a syntax tree as the internal data structure for our symbolic calculator. A syntax tree is a kind of tree, which in turn is a kind of linked data structure. Briefly, a linked data structure is an object which contains references, or links, to other like objects. A simple example is a linked list, where each element contains the data for it list entry and a link to the next list element. A tree is a linked structure that starts with a single "root" node. One or more "child" nodes are referenced from the root node, and each of these child nodes may in turn have children of their own. This linking pattern produces a branching data structure, as seen in the following diagram; hence the name "tree".

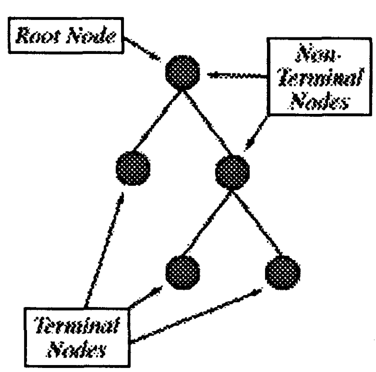


Figure 5.3

Trees are acyclic, which means that nodes cannot be linked in a loop. Each node has exactly one "parent" node, that is, one node of which it is a child. The exception is the root node, which has no parent. Nodes with no children are called terminal nodes, or "leaf" nodes. A syntax tree is a type of tree where each non-terminal node represents an operator or function, and its children represent its operands or arguments. In our program, mathematical operators such as addition and multiplication, or mathematical functions such as *sin* or *log* are represented by these non-terminal nodes. Leaf nodes represent the terminal symbols of an expression, such as numbers, constants or variables. The structure of a syntax tree represents syntactic information about the data it contains. In our case, the syntax tree represents the syntactic steps, or order of operations involved in evaluating a symbolic expression. For example, the simple expression $a + b * c$ could be represented by the following syntax tree:

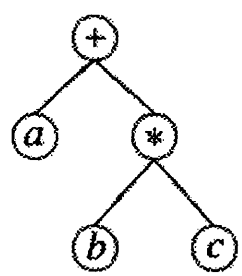


Figure 5.4

The root of this tree is the addition operation, and the children are its operands. The hierarchy of operators and arguments establishes a clear precedence of operations. The syntax tree for the expression $(a+b)*c$ is shown below:

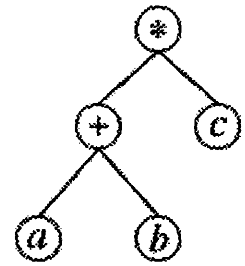

Figure 5.5

These two syntax trees are different, as are the expressions they represent. Syntax trees offer a clear and unambiguous way to store a wide variety of expressions.

### 5.2.7 The Syntax Tree - Advantages

The major advantage of the syntax tree is that it is flexible. It can represent a wide variety of different expressions that cannot be easily captured in the previously discussed data structures. Take, for example, an expression as simple and common as $e^x$. The polynomial representations are limited to just that - simple polynomials. Perhaps the sparse multivariate polynomial could be extended to think of *e* as a kind of "special" variable - namely a constant. Furthermore, and more difficult, the representation would have to be extended to allow for non-integer exponents. Even then, what happens when the exponent is more complicated than a single non-terminal symbol? What if the exponent is itself a polynomial expression, replete with its own exponents, and so on and so forth? The problem quickly grows out of any attempts of reasonable management. This doesn't even touch on the matter of functions within expressions, as in $x^2+\sin(x)$.

Granted, the representation was designed with polynomials in mind, but we wanted something more general. All of these expressions are easily represented in abstract syntax trees:

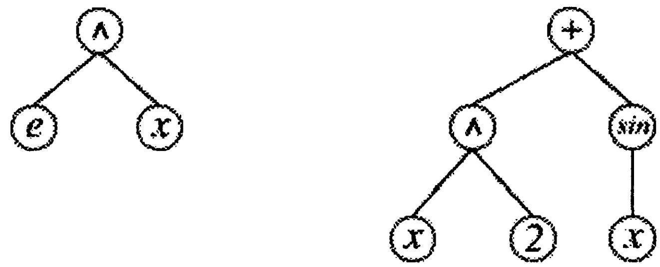


Figure 5.6

### 5.2.8 The Syntax Tree - Disadvantages

The generality that makes the syntax tree so appealing is also its biggest problem. While it's possible to represent numerous different expressions with a syntax tree, it's also possible to represent a single expression a number of different ways. For example, the expression $\frac{ab}{cd}$ can be represented in a number of different ways:

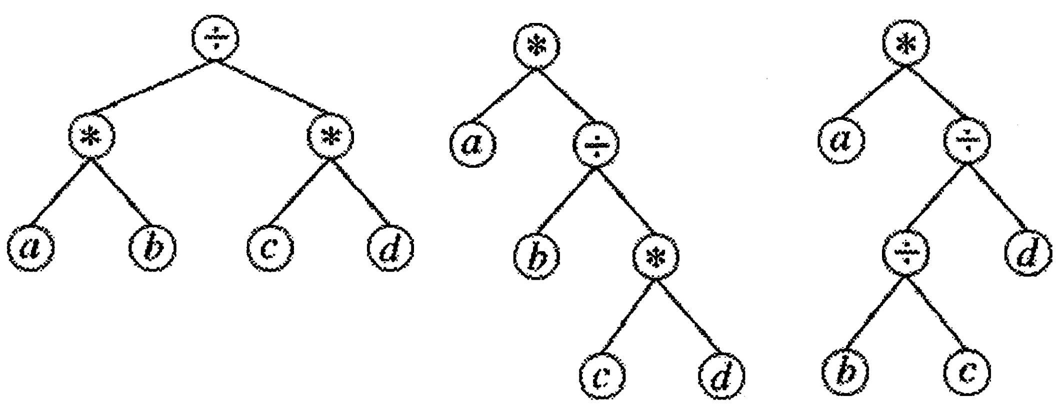

Figure 5.7

All are mathematically equivalent, but to a computer, they look nothing alike. As mentioned before, it's important to define a canonical form. Changing an expression to canonical form can be a difficult task in itself, but due to the wide variety of expressions syntax trees can represent, it's hard to define exactly what canonical form should be. Certain types of expressions tend to fit some forms better than others. A polynomial fits nicely in a form that orders terms by their degree. What happens when the exponents are more complicated? Both $x^{\left(y^2+y+1\right)}$ and $x^{\left(y^2+y+2\right)}$ could be said to have the same "degree" - that is, a polynomial in y of degree 2. Defining a general algorithm for ordering becomes complicated. Consider, also, the example of $x$ as compared to $1 * x$ or $x^1$. The trees of these expressions are as follows:

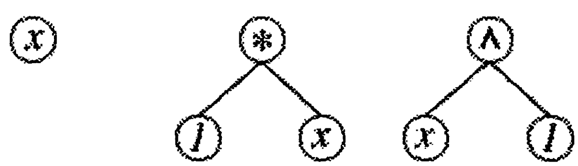

Figure 5.8

While it may be unlikely for a user to enter $x^1$ instead of $x$, it is not hard to imagine that such an expression may be obtained in the process of manipulating the expression symbolically. For example, $\frac{x^3}{x^2}$ might be simplified to $x^1$. The procedure to convert expressions to canonical form must take many factors into account.

## 5.3 Simplification

### 5.3.1 General Issues in Simplification

A very common task that any computer algebra system must perform is to simplify an expression. Simplifying expressions makes other tasks much easier, especially comparing expressions entered in different forms to see if they are equivalent. The system has to know how to add terms that should be added and how to add exponents together when the multiplicands have the same base. Computer algebra systems must always orders all the terms to arrive at a canonical form. There must be a consistent order that everything is sorted by. If exponents can be polynomials, such as $x^{x^3+3+x}$, then those exponents also need to be sorted. This could go on indefinitely, with each additional power being another complex polynomial. Sorting will need to take place on several levels to make it consistent. The uppermost exponents must be simplified first so that the lower level ones will sort properly. One can not simply sort from the lowest level first.

Systems must also decide whether or not to perform some simplifications. Identities for operations must be taken into account, so adding zero and multiplying by one will be dealt with properly. Either strip out all such identities or add them in to every operation, which would probably require more work and produce a more cluttered display.

There is also the decision whether or not to expand polynomials. Expanding $(x+2)^2$ may be trivial, but expanding $(6x^3+2x)^{300}$ would require an enormous amount of memory and time, not to mention far too much display space on the screen to be impractical. The factored form is much more compact. If integers are raised to a power, the limitations of the computer's number system may hinder expansion. This again shows why it is important to choose a number representation system that allows for arbitrarily large values. It may be necessary to expand simple polynomials of a very high order into large polynomials of a very high order in order to simplify further, but a lone value should probably be left as it is for simplicity in display. Expanding might be appropriate when there are several polynomials, and terms will cancel out when expanded, but no further simplification can be done otherwise.

The systems might also want to apply trigonometric or other identities to expressions for simplification of functions. For example, $\sec^2(x^2+1)$ and $\tan^2(x^2+1)+1$ are equivalent if the trig identity $\tan^2(x)+1=\sec^2(x)$ is applied. There are many other ways to manipulate functions, especially trigonometric ones, that could hinder further simplification. Sometimes the only way to simplify further is to apply these identities, which makes knowing when to use identities difficult. The natural logarithm function also has identities, which, besides being used to simplify expressions with the natural logarithm function, can also be used to simplify some other expressions. The system must know when to apply these identities and when to leave the functions as they are.

Simplifying is not only used when an expression is first entered in by the user, but, in particular, differentiating an equation will produce an expression that will need to be simplified for it to look like what the user expects to see. It is important that the computer algebra system be able to represent everything that may happen when expressions are simplified and expanded, but it must also decide whether or not to simplify certain operations depending on the circumstances.

### 5.3.2 The Steps of Simplification - Our Approach

In order to break up the complex task of simplification and reduction to a canonical form, we created a number of small algorithms that performed very simple, specific operations on syntax trees. By calling these simple procedures in order, and repeatedly, we are able to simplify many equivalent representations into a single deterministic form. In the following sections, we will describe the steps taken. Some of the steps seem to move away from simplification instead of towards it - these are intermediate steps that make later simplification easier.

### 5.3.3 Transforming Negatives

In this step, all negative operators (unary negative and subtraction) are transformed to terminal constants with negative values. For example, $x$ becomes $1 * x$ and $a - b$ becomes $a + (-1 * b)$. The trees of these expressions are as follows:

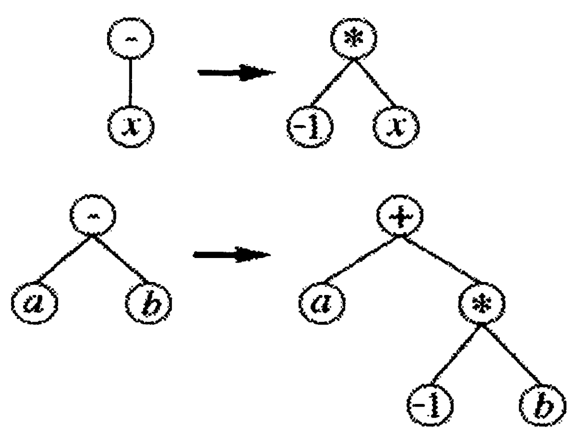


Figure 5.9

This step, while extremely simple, has a number of advantages in terms of defining a canonical form and simplifying later operations. For the canonical benefit, the above pairs of expressions, and others like them can be determined to be equivalent, obviously. The real benefit of this operation is that it significantly reduces the syntactic complexity of the expression trees. Two elements, negation and subtraction, are removed from the set of operators that have to be dealt with in later stages.

Subtraction is replaced by addition, a commutative operator, which allows greater flexibility in ordering. A subtraction node must have exactly two children, and their order cannot be reversed. An addition node, on the other hand, can have any number of children, and they can appear in any order.

### 5.3.4 Leveling Operators

When the expressions $a * b * c$ and $a + b + c$ are parsed by our calculator, the following syntax trees result:

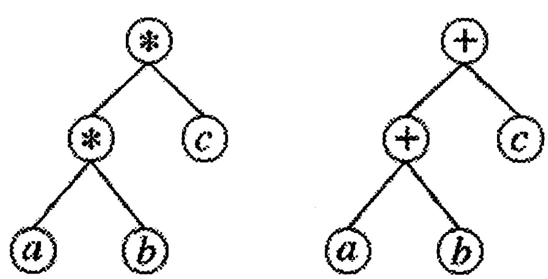


Figure 5.10

This is due to the fact that our parser assumes that all operators, with the exception of negation, are binary - that is, they have two operands. Since the parser is designed to read expressions written in infix notation, this is a valid assumption. But there's no reason that commutative operators such as addition and multiplication have to be binary in another notation. For example, $a + b + c$ could be written as $(+ a\ b\ c)$ in prefix notation. The same is true of our syntax tree. For example, the expressions $a + b + c + d$ and $(a + b) + (c + d)$ would be parsed as follows:

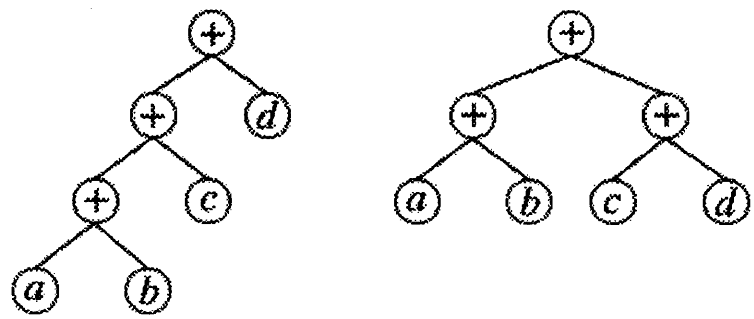

Figure 5.11

However, after the simplification step of leveling operators, both expressions are represented as:

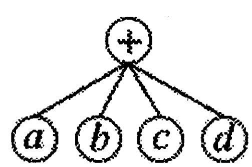

Figure 5.12

This operation trims unnecessary complexity from the syntax trees and resolves problems of associativity in canonical form.

### 5.3.5 Simplifying Rational Expressions

Recall the various syntax trees for the expression $\frac{ab}{cd}$ illustrated in section 5.2.8. This simplification step will transform a syntax tree so that a division node cannot be the immediate child of either a division node or a multiplication node. The end result is that any expression formed of multiplicative operators (multiply and divide) will be transformed so that there is a single division node at the top of the tree, with only multiplication operators below it. This simplification takes three specific cases into account in order to form a general procedure for other cases. The first case is the event when a division node ($D_1$) has another division node ($D_2$) as its numerator. In order to simplify this situation, the numerator of $D_1$ must become the numerator of $D_2$, and the denominator of $D_1$ must become the product of the denominators of $D_1$ and $D_2$. This transformation can be seen in the following diagram:

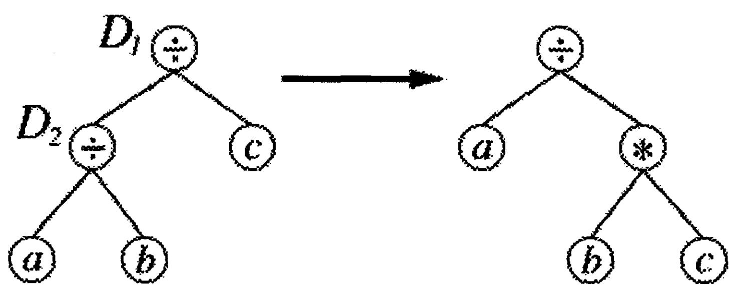


Figure 5.13

The second case is very similar to the first. In this case, the second division node ($D_2$) occurs in the denominator of the first ($D_1$). The numerator of $D_1$ becomes the product of the numerator of $D_1$ and the denominator of $D_2$. The denominator of $D_1$ becomes the numerator of $D_2$, as seen in the following illustration:

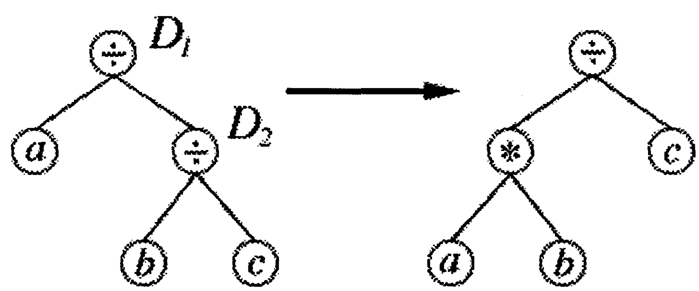


Figure 5.14

The final case is when a child of a multiplication node (M) is a division node (D). It doesn't matter how many children the multiplication node has, or how many of those children are division nodes. Only the first division node is considered in this simplification. This situation is a little more complicated than the previous two since the operation of the top node must be changed and its children moved, rather than just reshuffling some node links. To simplify this case, M is replaced by a division node whose numerator is the product of the numerator of D and the children of M (with the exception of D itself), and whose denominator is the denominator of D. This transformation is shown below:

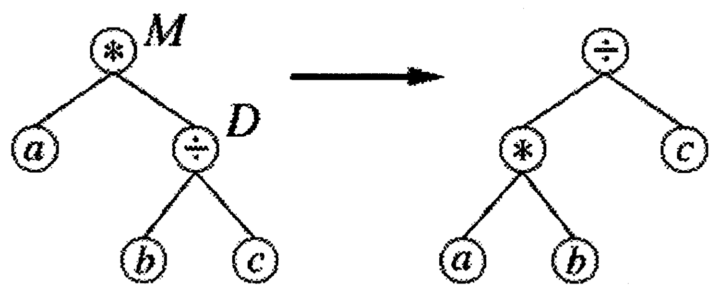


Figure 5.15

From repeated application of these three cases, more complicated expressions can be reduced:

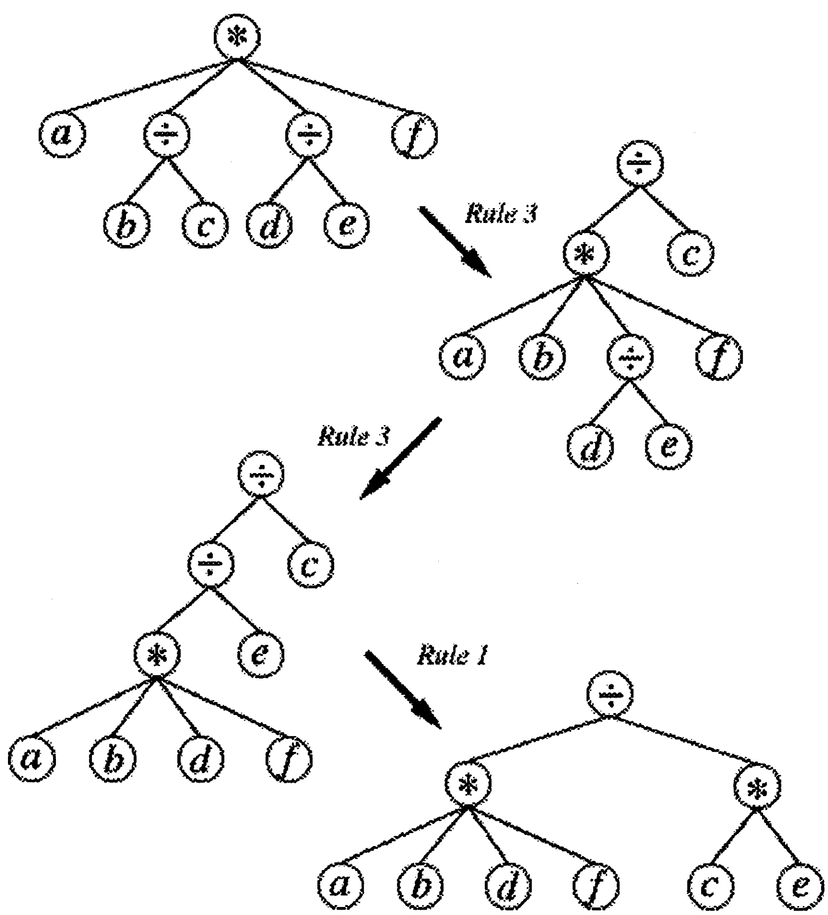


Figure 5.16

### 5.3.6 Collecting Like Terms

The first step involved in collecting like terms is to explicitly represent any coefficients or exponents a term may possess. For example, $x + 2x$ becomes $1x + 2x$ and $x * x^2$ becomes $x^1 * x^2$. The main reason for doing this transformation is to make all the children of an addition or multiplication node share a common form - that is, all the children are either multiplication nodes or power nodes, respectively. In order to collect like terms below a multiplication node, one compares the base of each child power node ($P_i$) with the bases of the remaining children ($P_{i+n}$). In the event that two bases are equal, the exponent of $P_i$ becomes the sum of the exponents of $P_i$ and $P_{i+n}$, and $P_{i+n}$ is removed:

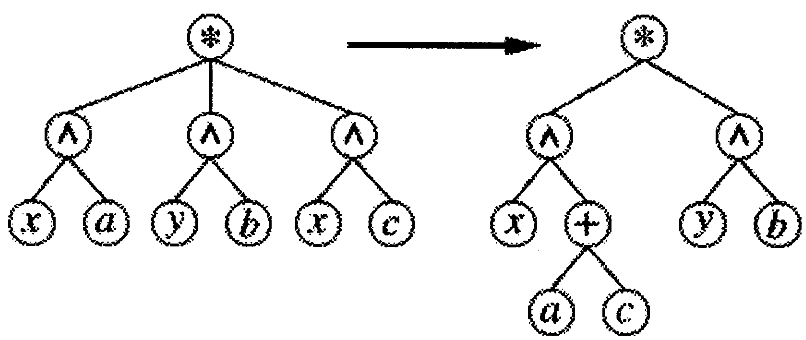


Figure 5.17

A similar operation would take place for collecting like terms under an addition node, but we have not actually implemented it in our calculator.

### 5.3.7 Folding Constants

Once terms have been collected together, unnecessary constants can be collected or removed. A constant, in this sense, is a real number in the expression, such as the three in $3 * e^{kx}$. The $k$ is a mathematical constant, but for purposes of symbolic manipulation, it is treated as a variable. When multiple constants occur below an addition or multiplication node, they can be combined (added or multiplied as the case may be) into a single constant. When both children of a power node are constants, it can optionally be replaced with a single number, although it is not always wise to do so. The number $10^{9000}$ is usually expressed as such because a 901 digit number is unwieldy. In our calculator, we fold a constant power term if the result is less than 1000, an arbitrary choice. Furthermore, a power term with a base of zero can be folded to zero, unless the exponent is also zero. In that case, our calculator simply leaves $0^0$ alone since it has no provisions for indeterminate forms. Power nodes with a base of one can be reduced to one, and power nodes with exponents of one or zero can be reduced to the base alone or one, respectively, with the previously mentioned exception of $0^0$. If a multiplication node contains a one, that child can be eliminated; if it contains a zero, the whole multiplication node can be replaced with zero. Also, if a multiplication or addition node is left with a single child in the course of these reductions, the node can be eliminated and replaced with its sole child.

### 5.3.8 Canonical Order

All of these simplifications are fine and wonderful, but what's the use if they can't even determine that $a + b$ and $b + a$ are equivalent? That's why it's important to define a canonical ordering of terms, as discussed in section 5.2.2 and 5.3.1. In order to arrange our syntax trees in canonical order, all the children of a commutative node are sorted with a simple ordering function. The children are sorted first by their node type. In our calculator, there is a different node type for each operator, and one for each of the following: variables, functions, and constants. After node type, the children are sorted lexicographically. This ordering scheme doesn't always order expressions the way one would expect to see it written, but it works well with syntax trees and is consistent - which is the important part.

### 5.3.9 Full Simplification

Sometimes a single iteration of the simplification steps is not enough to reduce an equation as much as it should be. To compensate for this, we keep iterating through these simplifications until the syntax tree ceases to change.

## 5.4 Advanced Operations

### 5.4.1 Introduction

After canonically representing an equation in memory, the Computer Algebra system can demonstrate its true power. The advanced operations that a system is capable of performing are what separate one system from another. Advanced operations include factorization, differentiation, integration, and finding the limit of a function.

### 5.4.2 Differentiation

Mathematical operations that are defined in terms of algorithmic processes are rather painlessly integrated into Computer Algebra systems. Assuming that an appropriate representation is chosen for describing an equation, any algorithmic manipulation can be fairly easily translated into a Computer Algebra system. Differentiation is one such operation that is defined algorithmically in a very general way and is therefore particularly well suited to a computational definition.

Differentiation essentially consists of four basic rules (Davenport 165):

$$(a \pm b)' = a' \pm b'$$
$$(ab)' = a'b + ab'$$
$$(\frac{a}{b})' = \frac{a'b - ab'}{b^2}$$
$$f(g(x))' = f'(g(x))g'(x)$$

The algorithm must only know two additional pieces of information. First, the algorithm must be informed that $(x^{\frac{p}{q}})' = (\frac{p}{q}x^{1-\frac{p}{q}})$, which enables the computation of the derivative of any function that does not contain references to other functions. Second, to be a completely flexible at differentiation, the algorithm must be aware of the derivatives of functions (e.g. sin, cos, ln, etc.). The differentiation of functions can easily be accomplished by storing a table of derivatives.

The ability of a computer to perform differentiation is thus demystified. Because we, as human problem solvers, compute derivatives in a very algorithmic way, it is easy for a computer to emulate such behavior. Artificial Intelligence is the attempt at algorithmically modeling a human's ability to think. However, when there is no obvious algorithm that exists, modeling such behavior becomes extremely difficult (and, at this point, all attempts are nothing more than an approximation). The same is true in Computer Algebra systems: some mathematical computations are not clearly performed algorithmically.

### 5.4.3 Integration

Integration is an example of an operation which, at first, appears to have no algorithmic definition. The only general rule that appears to be useable is that $\int f + g = \int f + \int g$. However, even this rule turns out to be unusable in certain cases. Consider attempting to find $\int x^x + (\log x)x^x$; breaking it up at its addition will not yield a solution. The two respective parts have no integral, yet the integral of the combination yields $x^x$ (Davenport 167).

Integration appears to be a compendium of different techniques such as integration by parts, integration by substitution, and simply consulting a table of known integrals. Which integration problems require which technique can not be generally defined. The first attempts at computer integration, then, took an obvious brute force approach. That is, try all possible known techniques until an answer is found.

Ultimately though, a full theory of integration in terms of an algorithmic process that computers can perform was developed. This theory is beyond the scope of this paper, but a summary of the theory is developed in Davenport, Siret, and Tournier pp. 167-186.

### 5.4.3 Differentiation - Our Implementation

The only advanced operation we chose to implement is differentiation in one variable. In order to differentiate a syntax tree, one must take a top-down approach. The differentiation procedure starts with the root node and tries to differentiate it based on what type of node it is. For example, if the node is an addition node, the derivative of the node is an addition node whose children are the derivatives of each of the original node's children. Differentiation is an inherently recursive procedure, and syntax trees are well suited to recursive evaluation. Below is a list of how each type of node is differentiated:

- *Addition Node*: As mentioned above, the derivative of an addition node is an addition node whose children are the derivatives of each of the original node's children.
- *Multiplication Node*: If a multiplication node has $n$ children, then by the product rule, the derivative of a multiplication node is an addition node with $n$ children. Each child of the addition node is a multiplication node, also with $n$ children. In the $i^{th}$ multiplication node, the $i^{th}$ child is the derivative of the $i^{th}$ child of the original node, and the other children are the same as the other children in the original node. For example, the derivative of $x * y * z * w$ is $x' * y * z * w + x * y' * z * w + x * y * z' * w + x * y * z * w'$.
- *Division Node*: The derivative of a division node is simply expressed by the quotient rule. The only difference is that the subtraction is replaced by addition and the second term is multiplied by -1 (in keeping with the idea of eliminating subtraction operations).
- *Power Node*: Though the power rule is one of the first methods of differentiation we learned, it wasn't very practical for our calculator. Instead we used the more general form $\frac{d(b(x)^{e(x)})}{dx} = b(x)^{e(x)} \frac{d(e(x) * \ln(b(x)))}{dx}$. It makes a mess of simple things like constant bases or powers, but if the resulting tree is simplified, everything is cleaned up.
- *Function Node*: Our program will only try to differentiate functions of one variable, although it will symbolically manipulate functions of an arbitrary number of variables.

In fact, the only functions it knows how to differentiate at the moment are *ln*, because it occurs so much in the differentiation of power nodes, and *sin* and *cos*. If the program doesn't know how to differentiate a function, it will simply encase the function and its arguments in a *Deriv(a,b)* function, where *a* is the unknown function, and *b* is the variable with respect to which the function is differentiated.

- *Variable Node:* If the variable is same as the independent variable for which we are differentiating, then the derivative node is the constant 1. Otherwise the variable is considered a symbolic constant and the derivative is the constant 0.
- *Constant Node*: The derivative is always the constant 0.

## 6. Conclusions

As a result of our research, we believe we have learned two important lessons. First, in studying the history of the Calculus, we became acutely aware of our own lack of understanding for the subject. Second, the software that automates algebraic manipulations requires in itself a complex set of theory, and we have gained an appreciation for this tribute to the accomplishment of modern technology. Despite having been trained and performed well in Calculus exams and courses, it is clear that our appreciation for the subject matter was minimal at best. We believe that studying the history of the subject has enhanced our understanding, yet there clearly remain numerous gaps in our knowledge.

As students, we have been trained not to understand the origins and reasoning behind the science we study, but merely to be adept at applying memorized techniques to problems. Though the application of science is undoubtedly a useful skill, we wonder whether we ought to be brought through the development of the subjects we study. Calculus is merely one branch of science, is it possible to truly appreciate subjects such as Physics without studying the experimental processes that ultimately converted popular thought to heliocentricity and universal gravitation? What, after all, do we really understand about the knowledge we purport to have attained if we do not understand the process that led to its development? Though studying the history of a subject is not a necessity in understanding its development, it seems that merely learning science as an act of route memorization is inherently contradictory to the process of science. After all, the process of science requires that one not accept theory without evidence, that one should attempt falsification within reason, and demands that one not take a theory as truth because it is argued from authority.

This paper was an attempt to remedy our own lack of understanding of the origins of one science we had studied. Though there still remain topics in the history of Calculus that we were unable to research or fully comprehend, some of the holes in our understanding have been filled in. In addition, the development of our own Computer Algebra system taught us the strength in learning through experimentation. Facing the technical issues head-on brought us a far deeper appreciation than any amount of reading could have conveyed. The major lesson learned is that our best approach to understanding is not to simply research known methods, but to simultaneously use and understand those techniques.

**Appendix A**

## Bibliography


Aho, Sethi, and Ullman, "Compilers: Principles, Techniques, and Tools", Addison-Wesley, Massachusetts, 1986.

Albrecht, Buchberger, Collins, and Loos, "Computer Algebra: Symolic and Algebraic Computation" 2nd Edition, Springer-Verlag, New York, 1983.

Ball, W. W. Rouse, "A Short Account of the History of Mathematics 4th Edition", 1908. Adapted for the Web at *www.maths.tcd.ie/pub/HistMath/People/RBallHist.html*.

Boyer, Carl B., "The History of the Calculus and its Conceptual Development", Dover Publications, Inc. New York, 1949.

Boyer, Carl B. and Uta Merzbach, "A History of Mathematics 2nd Edition", Wiley, New York, 1989.

Bradley and Smith, "Calculus", Prentice Hall, Englewood Cliffs, New Jersey, 1995.

Davenport, Siret, and Tournier, "Computer Algebra: Systems and Algorithms for Algebraic Computations", St. Edmundsbury Press Limit, Bury St Edmunds Suffolk, 1988.

Dubbey, J. M., "Development of Modern Mathematics", Butterworth & Co Ltd, New York,1975.

Edwards, C. H., "The Historical Development of the Calculus", Springer-Verlag, New York, 1979.

Hooper, Alfred, "Makers of Mathematics", Random House, New York, 1958.

Fischer and LeBlanc, "Crafting a Compiler", Benjamin/Cummings Publishing, Menlo Park, California, 1998.

Goldstine, Herman H., "A History of the Calculus of Variations from the 17th through 19th Century", Spring-Verlag, New York, 1980.

Gittleman, Arthur, "History of Mathematics", Charles E. Merrill Publishing Company, Columbus, Ohio, 1975.

Weisstein, Eric W. "Eric's Treasure Trove of Mathematics" published on the Web at *www.astro.virginia.edu/~eww6n/math/*.

**Appendix B**

## Web Address

As was mentioned in the paper, we developed our own Symbolic Calculator. A Web accessible version of the calculator can be found at

**http://www.math.wpi.edu/IQP/BVCalcHist/** .

Along with the Calculator is an HTML version of this paper.