

ENHANCING MATHEMATICAL UNDERSTANDING THROUGH PROGRAMMING: CREATING AN ANIMATED GUIDE TO TANGENT AND NORMAL LINES

Tanja Krunić¹

Abstract: The main focus of this paper is the creation of an animated tutorial to enhance understanding of the use of the first derivative to find the equation of tangent and normal lines to curves. The animated tutorial is created using Wolfram Mathematica. For given input data, such as the curve equation and the observed interval on the x-axis, the program generates an output in the form of animated steps for determining the equations of the tangent and normal lines of the curve. The tutorial is posted on the Wolfram cloud and is shareable online. Besides, the tutorial is bilingual: the user can choose between two languages, English and Serbian.

Keywords: Mathematics, education, tangent line, normal line, programming, Mathematica, cloud, animation

1. INTRODUCTION

Many students don't realize the connection between topics in mathematics and the real world, and as a consequence, they are not interested in those lectures at all. For some of them, mathematics is boring, and for others it is frustrating. Source [1] claims that there are two kinds of people in the world: those who have enjoyed math class in school, and the other 98% of the population, attributing this to the way mathematics is taught. The majority of students in the Higher Education Technical School of Professional Studies in Novi Sad have Mathematics as an obligatory subject. Among many other topics, students learn the application of the first derivative to determine equations of tangent and normal lines to curves. To make this topic more interesting for students, the professor of this subject created an interactive animated tutorial, or better said assistive teaching tool. For any input curve, the tool outputs the steps of determining the equations of tangent and normal lines, together with an appropriate graphical interpretation. The author of this paper has already created an interactive tutorial for learning the steps of determining the flow and plotting graphics of functions, presented in [2]. In contrast to the previous tutorial, the tutorial presented herein has a novelty – it is animated. Namely, to make it easier to follow the content, the steps are displayed on the screen one after the other with a delay of 2 seconds.

According to [3], the use of such specially created animation in teaching, to help students to better understand the material of specific lessons is considered as a kind of Computer Assisted Instruction (CAI). CAI refers to any kind of instructions presented via computer, but nowadays, CAI is mainly performed using web-based tutorials (WBT). To keep up with current trends, the animated tutorial presented herein is created on a cloud service, and as such, can be shared with all interested students. So this CAI is a WBT. To fulfill their purpose, WBTs are not supposed to be just a collection of texts. They rather should contain interactive tools that help students to learn actively [4]. Exactly this is the case with our assistive tool, as it allows the students to interact with it by choosing the input curve and then following the steps of determining the equations of the tangent and normal lines for a chosen point on the x-axis.

This paper is organized as follows: Section 2 gives a short theoretical overview of tangent and normal lines to curves and their application in practice. Section 3 is devoted to the programming language Mathematica and the Wolfram Cloud service. Features of the animated tutorial, as well as some code snippets used to create specific parts of the tutorial, are given in Section 4.

2. TANGENTS AND NORMAL LINES

Tangents and normal lines to curves have a wide range of applications in practice in many different scientific areas: in architecture (interior design, design curves in buildings, such as arches and domes), traffic engineering (road and bridge design), physics (force, motion, optics, astronomy, sound waves,

¹PhD, The Higher Education Technical School of Professional Studies in Novi Sad, Novi Sad, Serbia, Školska 1, email: krunic@vtsns.edu.rs

water waves, electromagnetic waves), illustrations and computer graphics, aerodynamics, medicine (MRI, CT Scans), cartography, artificial intelligence (neural networks) and robotics, [5] - [7]. So it is highly likely that students will encounter the practical application of tangents and normals in their future work. Hence the importance for them to understand the mathematical background of this topic.

The tangent to a curve at a specific point is a straight line that "touches" the curve only at that point. The slope of the tangent is equal to the derivative of the curve evaluated at the point where the tangent touches the curve, [8]. The equation of a line passing through a specific point (x_0, y_0) is given by

$$y - y_0 = k(x - x_0), \quad (1)$$

where $y_0 = f(x_0)$, and k presents the slope of the line. The slope of a tangent line touching the curve $y = f(x)$ at the point (x_0, y_0) , is calculated as

$$k = f'(x_0). \quad (2)$$

So, the equation of the tangent line to a curve at the point (x_0, y_0) is of the form

$$y - y_0 = f'(x_0)(x - x_0). \quad (3)$$

The normal to the curve at a specific point is the orthogonal line to the tangent at that point. If two lines are orthogonal to each other, then the product of their slopes equals to -1. This means that the slope of a normal to a curve at (x_0, y_0) is given by

$$k = -\frac{1}{f'(x_0)}, \quad (4)$$

if $f'(x_0) \neq 0$. In this case the equation of a normal line to a curve at the point (x_0, y_0) is given by

$$y - y_0 = -\frac{1}{f'(x_0)}(x - x_0). \quad (5)$$

Taking into account (3), in the case when $f'(x_0) = 0$, the equation of the tangent line becomes

$$y = y_0, \quad (6)$$

and the normal appears in the form of a vertical line

$$x = x_0. \quad (7)$$

A function can also have a vertical tangent at the point (x_0, y_0) if and only if

$$f'(x) \rightarrow -\infty \text{ or } f'(x) \rightarrow +\infty \text{ as } x \rightarrow x_0. \quad (8)$$

The vertical tangent is of the form

$$x = x_0, \quad (9)$$

and the normal line is given by

$$y = y_0, \quad (10)$$

in that case.

On the other hand, if the function is continuous at the point (x_0, y_0) , and the left and right derivatives differ, but are infinite, i.e.

$$f'(x) \rightarrow +\infty \text{ as } x \rightarrow x_0^- \text{ and } f'(x) \rightarrow -\infty \text{ as } x \rightarrow x_0^+ \quad (11)$$

or

$$f'(x) \rightarrow -\infty \text{ as } x \rightarrow x_0^- \text{ and } f'(x) \rightarrow +\infty \text{ as } x \rightarrow x_0^+, \quad (12)$$

the function has neither a tangent nor a normal at that point, but it does have a vertical cusp. Similarly, if the function is continuous at the point (x_0, y_0) , and the left and right derivatives differ, but are finite, the function has neither a tangent nor a normal at that point, but it does have a corner, [9].

3. MATHEMATICA PROGRAMMING LANGUAGE AND WOLFRAM CLOUD SERVICE

Wolfram Mathematica is used for creating the animated interactive tutorial for calculating the equations of tangent and normal lines of curves. It is a software system that includes extensive built-in libraries for a wide range of technical computing tasks. It supports machine learning, statistics, symbolic computation, data manipulation, network analysis, time series analysis, natural language processing (NLP), optimization, function plotting, data visualization, algorithm implementation, user interface creation, and integration with programs written in other programming languages. The programming language used in Mathematica is the Wolfram Language. The main advantage of the Wolfram Language is its seamless integration of symbolic and numerical computation, which are traditionally treated as separate domains. This enables solving a wide range of problems. Unlike other traditional programming languages, Mathematica is not strictly tied to a single programming paradigm; it supports simultaneously various paradigms (procedural, functional, object-oriented, rule-based, pattern-based, etc.). This is a big advantage since no programming paradigm is ideal for solving every problem. Besides, Mathematica enables processing, calculating, visualizing, and presenting data interactively, without the need to install add-ons. Also, it includes built-in, specialized functionality for a wide range of technical fields. So, there is no need to use several programs for solving a specific problem, [10].

The key advantage of Mathematica in choosing it for this project was the ability to use the free Mathematica online service, [11], which enables storing documents on the Wolfram cloud, [12]. Mathematica online service provides many features, among which is sharing and publishing documents.

4. THE ANIMATED INTERACTIVE TUTORIAL FOR TANGENT AND NORMAL LINES TO CURVES

Let us now present the main features of the animated tutorial on tangent and normal lines to curves. The tutorial is in the form of an interactive tool whose task is to calculate the equations of the tangent and normal lines of a curve. The tool has a pattern of the form

$$\text{tangentNormal}[\text{fun_}, x_0, \text{eps_}, \text{lang_}] . \quad (13)$$

As we can see, there are four required input data: *fun* - the function of the curve, *x0* - the point of tangency of the curve and the tangent, *eps* - half of the width of the interval on the x-axis where the graph of the function is plotted, and *lang* - the language of the tutorial. This tutorial provides an output in the form of instructions that solve this problem step by step. All instructions appear on the screen in separate lines, and with a delay of 2 seconds. According to [13], this is the optimal delay for short lines of text including 1-2 sentences. This delay is achieved by using the built-in function *Pause[2]* after every *Print[]* command. In the end, for illustration, an appropriate graph of the function together with the tangent and normal lines through the point x_0 is plotted on the interval $(x_0 - \epsilon, x_0 + \epsilon)$. The language parameter 'lang' is supposed to have one of the values 'eng' (for English) or 'srb' (for Serbian). Switching the tutorial from one language to another is enabled by choosing the proper language array for output depending on the value of the input parameter (Listing 1), and using the proper array of texts for the chosen language. Parts of the language arrays are shown in Listing 2.

```
If[lang=="srb",
text=textSrb,
If[lang=="eng",text=textEng,
Print[Style["You have selected the wrong language parameter. You can choose between 'srb' and 'eng'. \nОдабрали сте погрешан параметар за језик. Можете бирати између 'srb' и 'eng'.", FontSize -> 20, Spacings -> {0, 2}]]];
Goto[end]]];
```

Listing 1 - Code snippet that enables switching the language of the tutorial

<pre> textEng = { (*1*)"The task is to find the equations of the tangent and normal for the function f(x)=", (*2*)ToString[TraditionalForm[fun]], (*3*)StringJoin[" at the point ", ToString[TraditionalForm[Subscript[x, 0]]], "=", ToString[x0], "."], (*4*)StringJoin["The equation of the line at the point ", ToString[TraditionalForm[Subscript[x, 0]]], " is calculated by the formula y=", (*5*)"Here, k represents the coefficient of the line.", (*6*)StringJoin["The slope of the tangent at the point ", ToString[TraditionalForm[Subscript[x, 0]]], " is calculated by the formula k=", (*7*)"The derivative of the function is f'(x)=", } </pre>	<pre> textSrb={ (*1*)"Задатак је да пронађемо једначине тангенте и нормале за функцију f(x)=", (*2*)ToString[TraditionalForm[fun]], (*3*)StringJoin[" у тачци ", ToString[TraditionalForm[Subscript[x,0]]], "=",ToString[x 0],"."], (*4*)StringJoin["Једначина праве у тачци ",ToString[TraditionalForm[Subscript[x,0]]], " се рачуна по формули у=", (*5*)"При томе к представља коефицијент праве.", (*6*)StringJoin["Коефицијент правца тангенте у тачци ",ToString[TraditionalForm[Subscript[x,0]]], " се рачуна по формули k=", (*7*)"Извод функције је f'(x)=", } </pre>
--	--

Listing 2 – Text arrays for different languages

In the case when one tries to enter a value of the language parameter that is neither 'eng' nor 'srb' (i.e. lang='fr'), an appropriate error message appears on the screen, Figure 1.

```
In[2]:= result = tangentNormal[Exp[x - 2], 2, 3, "fr"]
```

You have selected the wrong language parameter. You can choose between 'srb' and 'eng'.

Одабрали сте погрешан параметар за језик. Можете бирати између 'srb' и 'eng'.

Figure 1 - The result on the screen upon selecting the wrong language input parameter

Figure 2 shows calculating the tangent and normal line through $x_0 = 2$ to the curve $f(x) = e^{x-2}$ on the interval $(-1,5)$. Note that in Mathematica the function e^{x-2} is presented as $Exp[x - 2]$. This tutorial also covers the case when $f'(x_0) = 0$. As shown in Section 2, in this case, the tangent and normal lines are of the form (6) and (7), respectively. Figure 2 (right) shows such an example for the curve $f(x) = x^2 + x + 1$ and $x_0 = -0.5$.

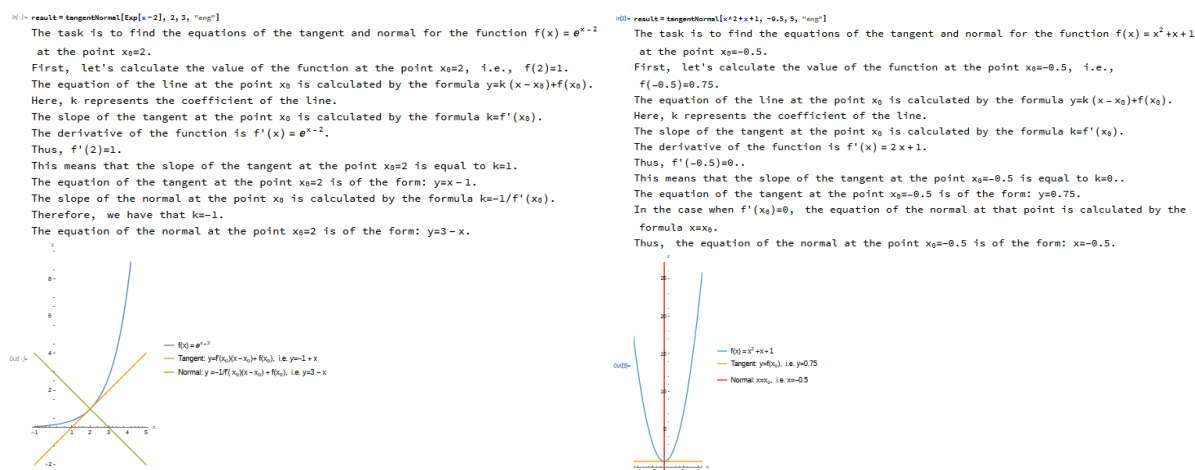


Figure 2 - Tangent and normal, $f(x) = e^{x-2}$, $x_0 = 2$ (left) and $f(x) = x^2 + x + 1$, $x_0 = -0.5$ (right)

The tutorial also covers the case of vertical tangents given by (8)-(10) in Section 2. Figure 3 shows an example of the existence of a vertical tangent at $x_0 = 0$ of the function $f(x) = \sqrt[3]{x}$. As we can see, we denoted the left derivative at the point x_0 with $f'_-(x_0)$ and the right derivative in the same point with $f'_+(x_0)$. Note that Mathematica does not natively support the cube root of negative numbers, requiring the third root to be expressed as a piecewise function to account for both positive and negative inputs. Hence the cube root is actually of the form

$$f[x_] := \text{Piecewise}[\{\{-(-x)^{(1/3)}, x < 0\}, \{x^{(1/3)}, x \geq 0\}\}] \quad (13)$$

```

in0:= f[x_] := Piecewise[{{(-x)^(1/3), x < 0}, {(x)^(1/3), x > 0}}]
in0:= result=tangentNormal[f[x], 0, 2, "eng"]

```

The task is to find the equations of the tangent and normal for the function $f(x) = \begin{cases} -\sqrt[3]{-x} & x < 0 \\ \sqrt[3]{x} & x \geq 0 \end{cases}$ at the point $x_0=0$.

First, let's calculate the value of the function at the point $x_0=0$, i.e., $f(0)=0$.

The equation of the line at the point x_0 is calculated by the formula $y=k(x-x_0)+f(x_0)$.

Here, k represents the coefficient of the line.

The slope of the tangent at the point x_0 is calculated by the formula $k=f'(x_0)$.

The derivative of the function is $f'(x) = \begin{cases} \frac{1}{3(-x)^{2/3}} & x < 0 \\ \frac{1}{3x^{2/3}} & x > 0 \\ \text{Indeterminate} & \text{True} \end{cases}$.

Thus, $f'(0)=\text{Indeterminate}$.

This means that the slope of the tangent at the point $x_0=0$ is equal to $k=\text{Indeterminate}$.

$f'(x_0)=\infty$

$f'(x_0)=\infty$

The equation of the tangent at the point $x_0=0$ is of the form: $x=0$

The equation of the normal at the point $x_0=0$ is of the form: $y=0$.

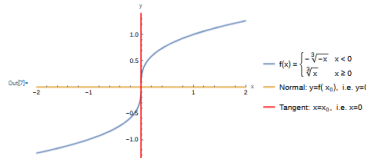


Figure 3 –A vertical tangent for $f(x) = \sqrt[3]{x}$ at $x_0 = 0$

To check which of the previously mentioned cases ($f'(x_0) = 0$, $f'(x_0) = \infty$, or the default case) is applicable, the *Which* control structure is used. Listing 3 shows the code of the default case. As can be seen, the code first calculates the slopes and equations of the tangent and normal, and then it displays the steps for their calculation on the screen. Then, the text for the legends of the curve, tangent, and normal is generated, followed by the graph itself. The code in the remaining two cases is similar, but in those cases, the equations of the tangent and normal are calculated differently (see Section 2), and the following tutorial text and legend labels are adjusted accordingly.

```

fPrime=D[fun,x];k1=fPrime/.x->x0;
Which[
  !NumberQ[k1],.....,
  k1==0, .....,
  True, tangentLine=k1*(x-x0)+(fun/.x->x0);
  Print[Style[StringJoin[text[[10]],ToString[TraditionalForm[Simplify[tangentLine]]],"."],FontSize->20,Spacings->
{0,2}]];
  Pause[2];
  k2=-1/k1;
  Print[Style[StringJoin[text[[11]],ToString[TraditionalForm[Subscript[x,0]]],"."],FontSize->20,Spacings->{0,2}]];
  Pause[2];
  Print[Style[StringJoin[text[[12]],ToString[TraditionalForm[k2]],"."],FontSize->20,Spacings->{0,2}]];
  Pause[2];
  normalLine=k2*(x-x0)+(fun/.x->x0);
  Print[Style[StringJoin[text[[13]],ToString[TraditionalForm[Simplify[normalLine]]],"."],FontSize->20,Spacings->{0,
2}]];
  legendT=StringJoin[text[[14]],ToString[TraditionalForm[Subscript[x,0]]],")(",ToString[TraditionalForm[(x-
Subscript[x,0])] ]),")+ f(",ToString[TraditionalForm[Subscript[x,0]]],")",text[[26]],"y=",ToString[Simplify[tangentLine]]];
  legendN=StringJoin[text[[15]],ToString[TraditionalForm[Subscript[x,0]]],")(",ToString[TraditionalForm[(x-
Subscript[x,0])] ]),")+ f(",ToString[TraditionalForm[Subscript[x,0]]],")",text[[26]],"y=",
ToString[TraditionalForm[Simplify[normalLine]]];
  gr=Plot[{fun,tangentLine,normalLine},{x,x0-eps,x0+eps},PlotLegends->{legendF,legendT,legendN},AxesLabel->
{"x","y"},AspectRatio->Automatic];
];
Return[gr];
Label[end];

```

Listing 3 – The default case of the *Which* control structure

A special case may occur if the input function is a line, i.e. $f(x) = ax + b$. In that case, the tangent is the line itself. The slope of the normal is then calculated as $k = -\frac{1}{a}$, where $a = f'(x_0)$. Very similar, in the case when the function is constant, i.e. $f(x) = c$, the tangent is the line $f(x) = c$, itself since then $f'(x_0) = 0$, and according to (7), the normal is the vertical line $x = x_0$. The output of

the tutorial for the case when the input function is $f(x) = 2x + 1$, at $x_0 = 1$ is shown in Figure 4 (left) and for $f(x) = 2$ at $x_0 = 1$ in Figure 4 (right).

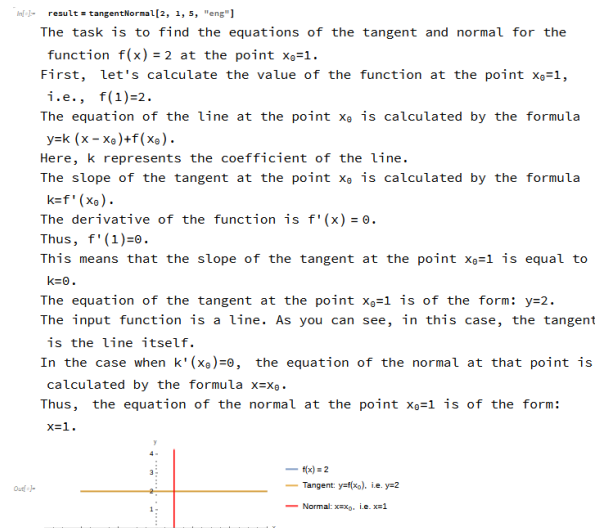
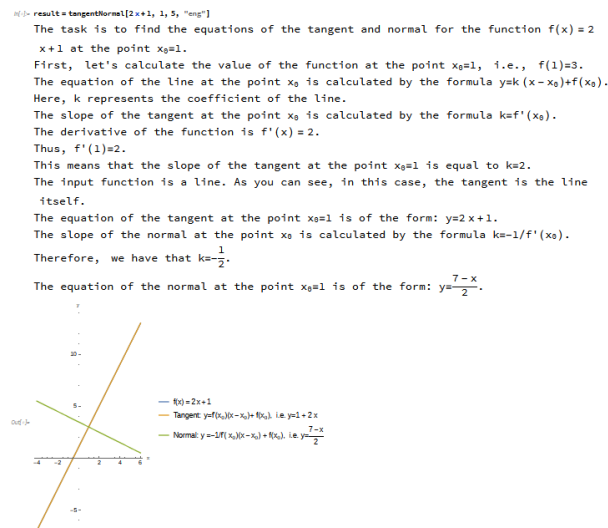


Figure 4 - The tutorial for $f(x) = 2x + 1$ (left) and $f(x) = 2$ (right) at $x_0 = 1$

In these two cases, a note with the information that in the case of a linear input function the tangent coincides with the linear function itself is displayed on the screen. This note is stored in the text array at index 16, i.e. in `text[[16]]`. A criterion for a linear function is $f'(x) = 0$ or $f''(x) = 0$, see Listing 4.

```
fSecond=D[fun,{x,2}];
If[fPrime==0||fSecond==0,Print[Style[text[[16]],FontSize->20,Spacings->{0,2}]]];
```

Listing 4 – The case of linear input functions

The tutorial also covers the case of non-smooth functions presented in Section 2, i.e. the cases when cusps and corners arise at the point (x_0, y_0) . Figure 5 (left) shows the existence of a vertical cusp at $x_0 = 0$ for the function $f(x) = \sqrt[3]{x^2}$, which is (for the reasons mentioned above) presented as a piecewise function of the form

$$f[x_]:= \text{Piecewise}[\{ \{(-x)^{(2/3)}, x < 0\}, \{x^{(2/3)}, x \geq 0\} \}] \quad (14)$$

The existence of a corner at $x_0 = 1$ for the function $f(x) = |x^3 - 1|$ is shown in Figure 5 (right). Note that the input function in this case is presented as

$$f[x_]:= \text{Piecewise}[\{ \{x^3 - 1, x \geq 1\}, \{-(x^3 - 1), x < 1\} \}]. \quad (15)$$

Listing 5 shows the code snippet for detecting if a function has a vertical cusp or a corner at point x_0 . This code snippet is placed at the top of the program. Firstly, the left and the right derivatives at the point x_0 are calculated. If the values differ, the program checks if the values are infinite (the function has a vertical cusp) or finite (the function has a corner) at x_0 . The appropriate information is displayed on the screen, followed by the graph of the function.

```
leftDerivative = Limit[(f[x] - f[x0]) / (x - x0), x -> x0, Direction -> "FromBelow"];
rightDerivative = Limit[(f[x] - f[x0]) / (x - x0), x -> x0, Direction -> "FromAbove"];
If[leftDerivative!=rightDerivative,
Print[Style[Row[{ "f'" <> Subscript["-", " "] [[1]] <> "(" <> ToString[TraditionalForm[Subscript[x,0]] <> ")",
"=", ToString[TraditionalForm[leftDerivative]]}],FontSize->20,Spacings->{0,2}]],
Print[Style[Row[{ "f'" <> Subscript["+", " "] [[1]] <> "(" <> ToString[TraditionalForm[Subscript[x,0]] <> ")",
"=", ToString[TraditionalForm[rightDerivative]]}],FontSize->20,Spacings->{0,2}]]];
```

```
If[Abs[leftDerivative]==Infinity, Print[Style[text[[27]],FontSize->20, Spacings->{0,2}]],
Print[Style[text[[32]],FontSize->20, Spacings->{0,2}]]];

gr=Plot[fun,{x,x0-eps,x0+eps}, PlotLegends->{legendF}, AxesLabel->{"x","y"},AspectRatio->Automatic];
Return[gr];
Goto[end]];
```

Listing 5 – Code for detecting vertical cusps and corners

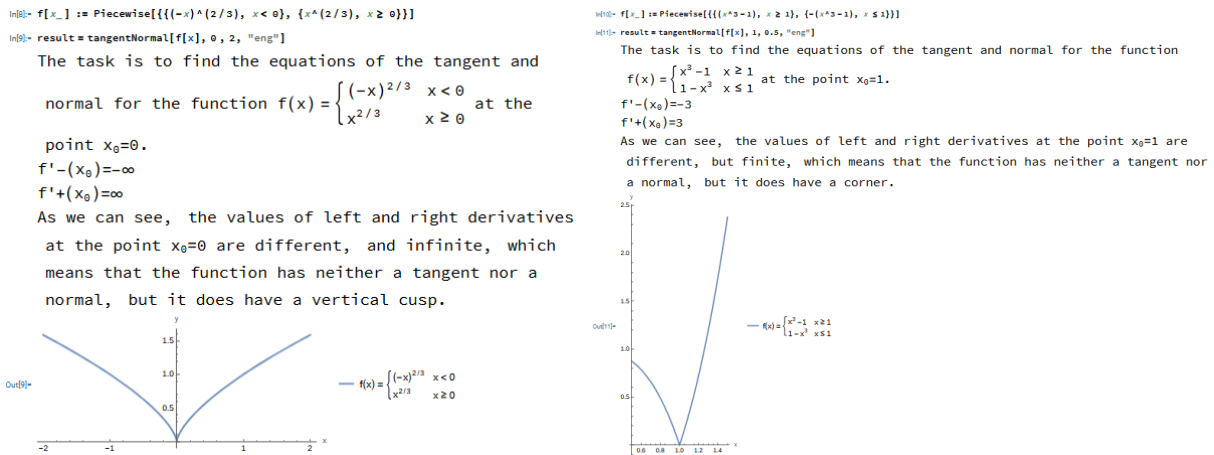


Figure 5 - A vertical cusp for $f(x) = \sqrt[3]{x^2}$, $x_0 = 0$ (left), a corner for $(x) = |x^3 - 1|$, $x_0 = 1$ (right)

Besides, some cases of incorrect user input were also handled. Listing 6 shows the code that handles three cases of incorrect user input: when the point x_0 , or the left or right edge of the observed interval does not belong to the domain of the function. In each of these cases, the appropriate information is displayed on the screen. The case when the left boundary of the observed interval (2,4) for the function $f(x) = 3x - 4 - \ln(x - 2)$ does not belong to the domain of the function, is shown in Figure 6.

```
domain = FunctionDomain[fun, x];
If[Simplify[domain /. x -> x0] == False, Print[Style[text[[17]], FontSize -> 20, Spacings -> {0, 2}]]; Goto[end]];
If[Simplify[domain /. x -> (x0-eps)] == False, Print[Style[text[[18]], FontSize -> 20, Spacings -> {0, 2}]]; Goto[end]];
If[Simplify[domain /. x -> (x0+eps)] == False, Print[Style[text[[19]], FontSize -> 20, Spacings -> {0, 2}]]; Goto[end]];
```

Listing 6 – Handling incorrect output

```
in[92]:= result = tangentNormal[3 x - 4 - Log[x - 2], 3, 1, "eng"]
```

The left boundary of the observed interval $x=2$ does not belong to the domain of the function.

Figure 6 – Incorrect input: $x = 2$ does not belong to the domain of $f(x) = 3x - 4 - \ln(x - 2)$

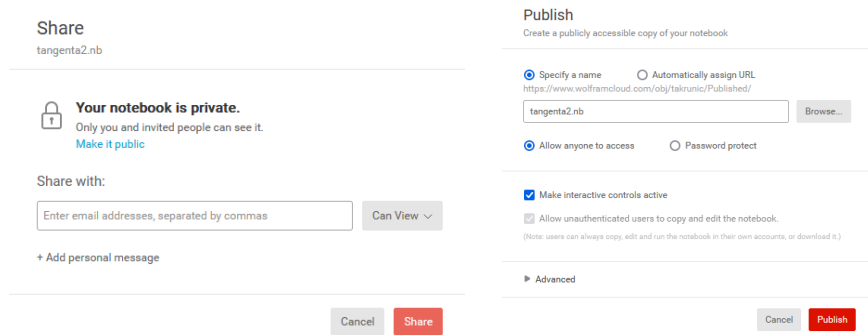


Figure 7 – Sharing (left) and publishing (right) notebooks online

As already mentioned, the Wolfram cloud service enables sharing documents online (Figure 7, left), so this tutorial can be shared with all students who are interested in this topic. Moreover, documents stored on this service can be published online (Figure 7, right) and become accessible to everyone. This tutorial has to undergo some additional testing and is intended to be published afterward.

5. CONCLUSION

The success of teaching mathematics largely depends on the teacher's ability to engage students with the subject. Exactly this was the motivation to create the interactive tutorial presented herein. The animated interactive tutorial covers many different cases that can occur when calculating the tangent and normal lines to curves at some point x_0 . All steps are explained and appear on the screen with a delay of 2 seconds, which helps the user to stay focused.

Many other topics of mathematics can be used for the creation of similar interactive learning tools. Thus, creating interactive tutorials with other mathematics topics will remain for further investigation by the author of this paper. As an interactive tutorial for plotting graphics of functions is already created by the author, new tutorials could include other topics like partial integration, operations on matrices, calculating inverse matrices, calculating determinants, and other topics covered by the Mathematics course held in The Higher Education Technical School of Professional Studies in Novi Sad. For the selection of topics covered by interactive animated tutorials, a questionnaire could be sent to students to figure out which topics students find difficult to master, so that an interactive tutorial on those topics would be useful to them. Moreover, when accrediting the subject of Mathematics at the aforementioned school, standard lessons were selected that are mostly covered in other higher education institutions of technical professions in Serbia, but also in other European countries. The collection of animated tutorials could further be integrated into any popular learning management system like Moodle or Canvas since they can be published online. Such a system could be used widely.

6. REFERENCES

- [1] *Why Math Class Is Boring—and What to Do About It*, <https://fs.blog/mathematicians-lament-2/>, December 2024
- [2] Krunić T.: *The development of a mathematical software for analyzing and drawing the graph of a function*, (original: Крунић, Т: *Развој математичког софтвера за испитивање тока и цртање графика функције*), The 1st conference of advanced technologies in the function of economic development, Vrnjacka Banja, Serbia, June 2022, ISBN 978-86-6211-138-8, https://napredneteh.vtsns.edu.rs/NTP_2022/Zbornik_radova_NTP_2022.html
- [3] Han HX. D., Toh T.L: *Use of animation to facilitate students in acquiring problem-solving: From Theory to Practice*, The Mathematics Enthusiast, 16(1), 377-387, 2019
- [4] Walsh, J., *Methods of instruction, Information Literacy Instruction, Selecting an Effective Model*, 3-55, 2011.
- [5] *Tangents in Everyday Life*, <https://www.geeksforgeeks.org/tangents-in-everyday-life/>, April 2024
- [6] *Tangent in Day-to-Day Life Situations*, <https://unacademy.com/content/jee/study-material/mathematics/tangent-in-day-to-day-life-situations/>, December 2024
- [7] Goel S.: *10 Common Applications Of Tangents and Normal in Real-life*, <https://numberdyslexia.com/applications-of-tangents-and-normal-in-real-life/>, August 2023
- [8] *Tangents and Normals*, <https://www.ncl.ac.uk/webtemplate/ask-assets/external/maths-resources/core-mathematics/calculus/tangents-and-normals.html>,
- [9] *Vertical tangents and cusps*, S.O.S. Mathematics, <https://www.drkhamsi.com/sosmath/calculus/diff/der09/der09.html>, December 2024
- [10] *Why Mathematica?*, <https://www.wolfram.com/mathematica/compare-mathematica/>, December 2024
- [11] *Mathematica online*, <https://www.wolfram.com/mathematica/online/>, March 2024
- [12] *Wolfram cloud*, <https://www.wolframcloud.com/>, November 2024
- [13] OpenAI: *Optimal delay for text animation in tutorials*. ChatGPT. <https://chatgpt.com/>, December 2024.