DR. B. R. AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY, JALANDHAR

3RD YEAR MINOR PROJECT REPORT

Department of Computer science and Engineering

Session: 2020-2024

ASYMPTOTIC GENIUS

- *Online Asymptotic Calculator and Plotter*

MENTORED BY :

Dr. Lalatendu Behera

Assistant Professor

[Department of CSE]

PRESENTED BY :

(**Group – 50)**

Cherry Kataria (20103045)

Harmandeep Singh (20103063)

Ridhima Parmar (20103123)

ACKNOWLEDGEMENT

It is with great pleasure that we express our heartfelt gratitude to all those who have contributed to the successful completion of our project on Asymptotic Notation. We, the team members of this project, would like to take this opportunity to thank all those who have provided us with their invaluable support and guidance throughout the project.

First and foremost, we would like to express our sincere thanks to our project mentor, Professor Lalatendu Behera, who has been a constant source of motivation, guidance, and inspiration. His valuable inputs, feedback, and support have been instrumental in shaping our project in the right direction.

We are also grateful to Dr. Rajneesh Rani, Head of Department of Computer Science & Engineering, for providing us with the necessary resources to carry out this project successfully.

We would also like to extend our sincere gratitude to all the faculty members of the Department of Computer Science & Engineering for their valuable suggestions and feedback at every stage of our project. Their expertise and insights have been critical in the successful completion of our project.

Thank you all for your invaluable contributions and support towards the successful completion of our project on Asymptotic Notation.

Sincerely,

[Group – 50]

DECLARATION

We, Group – 50, hereby declare that our Project titled- "**Asymptotic Notation**" being submitted by us in the Department of Computer Science and Engineering is a project work carried by us under the noble supervision of Dr. Lalatendu Behera and the project has not been copied from anywhere and has been made solely by us.

We will be solely responsible if some Plagiarism is found.

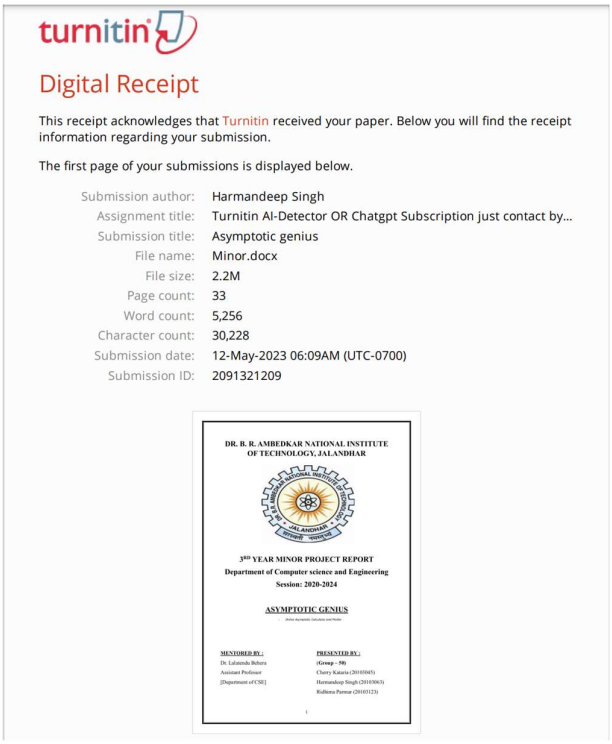
Thank You All.

Date: 11th May, 2023

[Group 50]

PLAIGIARISM CHECK

We, *Group 50, have checked plagiarism for our Project Report for our project Asymptotic Notation at* ***turnitin.*** *We are thankful to our mentor- Dr. Lalatendu Behera for guiding us at this. Below is the digital receipt. The Plagiarism is less than 10%.*



turnitin

Digital Receipt

This receipt acknowledges that Turnitin received your paper. Below you will find the receipt information regarding your submission.

The first page of your submissions is displayed below.

Submission author:	Harmandeep Singh
Assignment title:	Turnitin AI-Detector OR Chatgpt Subscription just contact by...
Submission title:	Asymptotic genius
File name:	Minor.docx
File size:	2.2M
Page count:	33
Word count:	5,256
Character count:	30,228
Submission date:	12-May-2023 06:09AM (UTC-0700)
Submission ID:	2091321209

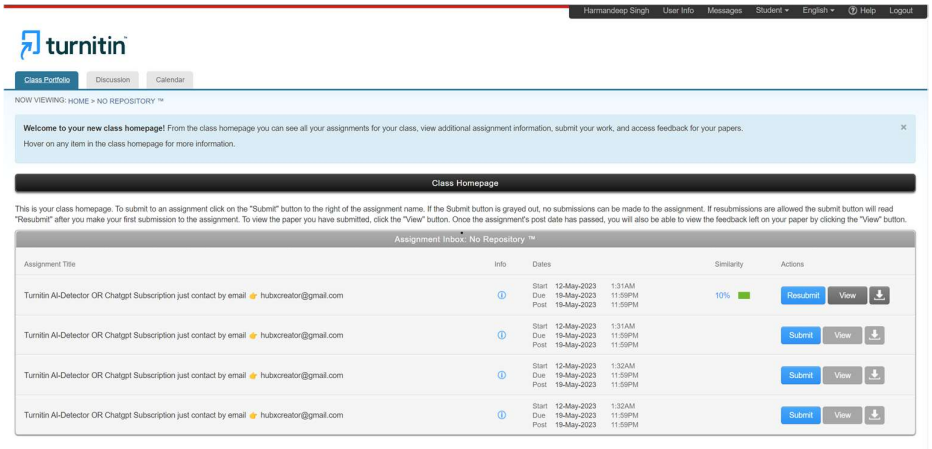
DR. B. R. AMBEDKAR NATIONAL INSTITUTE OF TECHNOLOGY, JALANDHAR

3RD YEAR MINOR PROJECT REPORT
Department of Computer science and Engineering
Session: 2020-2024

ASYMPTOTIC GENIUS

MENTORED BY :
Dr. Lalatendu Behera
Assistant Professor
[Department of CSE]

PRESENTED BY :
(Group – 50)
Cherry Kataria (20103045)
Harmandeep Singh (20103063)
Kulbeera Parmar (20103123)



Harmandeep Singh User Info Messages Student English Help Logout

turnitin

Class Portfolio Discussion Calendar

NOW VIEWING: HOME > NO REPOSITORY ™

Welcome to your new class homepage! From the class homepage you can see all your assignments for your class, view additional assignment information, submit your work, and access feedback for your papers.
Hover on any item in the class homepage for more information.

Class Homepage

This is your class homepage. To submit to an assignment click on the "Submit" button to the right of the assignment name. If the Submit button is grayed out, no submissions can be made to the assignment. If resubmissions are allowed the submit button will read "Resubmit" after you make your first submission to the assignment. To view the paper you have submitted, click the "View" button. Once the assignment's post date has passed, you will also be able to view the feedback left on your paper by clicking the "View" button.

Assignment Inbox: No Repository ™

Assignment Title	Info	Dates	Similarity	Actions
Turnitin AI-Detector OR Chatgpt Subscription just contact by email hubxcreator@gmail.com		Start 12-May-2023 1:31AM Due 19-May-2023 11:59PM Post 19-May-2023 11:59PM	10%	Resubmit View
Turnitin AI-Detector OR Chatgpt Subscription just contact by email hubxcreator@gmail.com		Start 12-May-2023 1:31AM Due 19-May-2023 11:59PM Post 19-May-2023 11:59PM		Submit View
Turnitin AI-Detector OR Chatgpt Subscription just contact by email hubxcreator@gmail.com		Start 12-May-2023 1:32AM Due 19-May-2023 11:59PM Post 19-May-2023 11:59PM		Submit View
Turnitin AI-Detector OR Chatgpt Subscription just contact by email hubxcreator@gmail.com		Start 12-May-2023 1:32AM Due 19-May-2023 11:59PM Post 19-May-2023 11:59PM		Submit View

Table Of Contents

ACKNOWLEDGEMENT……..……..…(2)

DECLARATION.…….(3)

PLAIGIARISM CHECK REPORT……(4)

1. INTRODUCTION- BUILDING UP THE FOUNDATION

1.1. Background of the problem……..……..……..…..(6)
1.2. Problem Statement ……..……..……..…..……...(7)
1.3. Necessity…….……..……(8)
1.4 Objective…….……(10)

1.5. Feasibility: Technical And Non-Technical(Brief)…..……(12)

2. PROPOSED SOLUTION- ICE BREAKING

2.1. Identifying Stakeholders…...…….....…….……...(13)

2.2. Detailed Solution…….…….…….....(14)

3. TECHNICAL ANALYSIS

3.1. UML Diagrams..……..……..……(15)

3.2. Tech Stack Analysis…….…..……...(21)

4. ECONOMIC ANALYSIS…….…..……..…(25)

5. RESULT AND DISCUSSION

5.1. App Usage Instruction…..……..…(26)

5.2. Risk Analysis……..…….....(29)

5.3. Deployment and Testing Status…….……..(30)

6. SOCIAL AND ENVIRONMENTAL IMPACT…...…….……..…(31)

7. CONCLUSION…….……(32)

8. REFERENCES…….……..……..……(33)

INTRODUCTION-Building Up the Foundation

1.1 Background

Our project on asymptotic notation uses HTML, CSS, and JavaScript to create a web application that helps users understand and visualize asymptotic notation concepts. The goal of this project is to provide an easy user interface that allows users to enter algebraic expressions and visualize the growth rate of a function using different asymptotic symbols such as Big O, Big Omega, Big Theta.

To achieve this goal, we used our knowledge of algebraic expressions and algorithmic analysis to create a backend function that calculates the growth rate of the input function. The results are displayed in interactive graphs that visualize the growth rate using different asymptotic symbols.

The aim of this project is to provide a suitable and intuitive tool for students and professionals who need to analyze and compare the efficiency of algorithms. Using HTML, CSS, and JavaScript, we aim to make our apps easily accessible to anyone with an internet connection and a browser. Our project is based on the mathematical concepts of asymptotic notation, and we hope to use this project to make these concepts available to a wider audience.

1.2 Problem Statement

The major Problems that motivated the solution are as follows:

1. **Traditional methods of teaching asymptotic notation can be difficult:**
   Traditional methods of teaching asymptotic notation, such as textbooks and lectures, can be dry and difficult to understand. The abstract nature of this notation can make the concepts difficult for students and professionals to understand and difficult to use effectively.

2. **Analyzing algorithms and comparing performance using asymptotic notation is time-consuming and error-prone:**
   Analyzing algorithms and comparing their performance using asymptotic notation can be time-consuming and error-prone, especially for those unfamiliar with the concept. Manual calculations are tedious and error-prone even for experienced programmers.

3. **Create a user-friendly and intuitive interface:**
   To address these issues, this project aims to create a web application that provides a user-friendly and intuitive interface for understanding and visualizing asymptotic notation concepts. This software makes it easier for students and professionals to understand the concepts by providing a more attractive and interactive platform.

4. **Let users enter algebraic expressions and see the growth rate of the function:**
   This web application allows users to enter algebraic expressions and instantly see the growth rate of the function using various asymptotic symbols. This makes it easier for users to better understand the concepts and analyze and compare the efficiency of algorithms.

5. **Making the concept of asymptotic symbol more accessible:**
   Additionally, this project makes the concept of asymptotic notation more accessible by using HTML, CSS, and JavaScript to create a web application that is easily accessible from anywhere with an Internet connection. This makes the concept accessible and understandable to a wider audience.

6. **Providing an interactive and attractive platform:**
   The goal of this project is to make the concept more accessible and understandable to a wider audience by providing an interactive and engaging platform for understanding asymptotic notation. This project helps users to understand:

1.3 Necessity

The necessity of your Asymptotic Notation website project:

I. Accessibility

Creating a web application for understanding asymptotic notation will make the concept easier for more people to understand. Students and professionals who struggle with traditional teaching methods can benefit from the user-friendly and intuitive interface.

II. Performance

Analyzing the efficiency of algorithms is important for developing high-performance software. By providing a platform where users can enter algebraic expressions and instantly see growth rates of functions using various asymptotic symbols, web applications can help users analyze and compare the performance of algorithms more efficiently.

III. Error prevention

Performing calculations by hand can be tedious and error-prone, especially for those unfamiliar with the concept of asymptotic notation. Web applications avoid errors, save time, and reduce the risk of mistakes by using various asymptotic symbols to calculate growth rates of functions.

IV Engagement

Traditional methods of teaching asymptotic notation can be dry and difficult to understand. Web applications help users learn and retain information more efficiently by providing an interactive and engaging platform for understanding concepts.

V. Innovation

The aim of this project is to provide a new and innovative way to understand and visualize the concept of asymptotic notation. This project uses HTML, CSS and JavaScript to create web applications that provide a more engaging and interactive platform than traditional educational methods.

VI. Convenience

Creating a web application that can be accessed from anywhere with an internet connection makes it easier to learn the concept of asymptotic notation. Students and professionals can access the platform at home, work or school without the need for textbooks or lectures.

VII. Future-Proofing

A future project will prove this concept by providing a web application for asymptotic notation understanding. As technology continues to evolve, web applications can be updated and improved to keep up with the latest developments and ensure the availability of concepts.

1.4 Objective

I. Education

The main goal of the web application is to educate users about the concept of asymptotic notation and its importance in computer science and software development.

II. Simplification

This web application aims to simplify the process of understanding asymptotic symbols by providing a user-friendly interface that allows users to enter algebraic expressions and instantly see the growth rates of functions using different asymptotic symbols.

III. Comparison

This web application helps users compare and analyze the performance of algorithms by providing a platform where users can enter multiple formulas and compare their growth rates using different asymptotic symbols.

IV. Visualization

This web application provides a platform for users to visualize the concept of asymptotic notation through interactive charts and graphs.

V. Customization

This web application allows users to select different asymptotic symbols to display growth rates of functions and customize their experience by adjusting graph visualization settings.

VI. User Engagement

This web application engages users by providing an interactive and user-friendly interface that makes learning about asymptotic symbols more fun and accessible.

VII. Accessibility

This web application makes the concept of asymptotic notation more accessible to a wide audience, including students, software developers, and experts in related fields.

VIII. Future development

This web application provides a foundation for future development, including potential collaboration and further research in the areas of asymptotic notation and algorithmic analysis.

1.5 Feasibility: Technical and Non- Technical

❖ **Technical feasibility**

Technical feasibility analysis considers the technical aspects of web application development, such as development tools, frameworks, hosting, and data management systems. Feasibility points are:

1. **Development tools:**
   The development tools needed to build websites, such as HTML, CSS, and JavaScript, are widely available and commonly used. These tools are easy to learn and use and make the development process relatively simple.

2. **Frameworks and libraries:**
   There are many open source frameworks and libraries available for web development, such as React and Angular, which can simplify the development process and provide additional functionality.

❖ **Non-technical feasibility:**

A non-technical feasibility analysis considers the non-technical aspects of web application development, such as the financial, social and environmental impacts of the project. Feasibility points are:

1. **Financial feasibility:**
   Website development costs vary depending on the scope and size of the project and the expertise and experience of the development team. However, open source technology and cloud-based hosting can keep costs down and make your project economically viable.

2. **Social feasibility:**
   A website's social impact depends on its ability to educate and engage users with the concept of asymptotic notation. The website's user-friendly interface and interactive features make the concept more accessible and attractive, helping to make the project feasible.
3. **Environmental feasibility:**
   Developing and using websites has minimal environmental impact, as no physical infrastructure or operations are involved. However, you can minimize your website's carbon footprint by using green hosting and development practices.

PROPOSED SOLUTION – ICE BREAKING

"Identifying a Problem is an easy task, finding a solution and making your own way toward it – is the Real Challenge."

2.1 Identifying Stakeholders

1. **User:**
   End users are the main stakeholders of your website. These are the people who use the website to learn about asymptotic notation. In order to make your website user-friendly and informative, it is important to consider the needs and wants of your users during the website development process.

2. **Developer:**
   The development team is responsible for building the website. They use their technical skills to create website features and functionality. It is important to ensure that your development team has the skills and resources necessary to create a quality website.

3. **Field expert:**
   Domain experts are people who have deep knowledge of asymptotic notation. We provide valuable insight and feedback throughout the website development process and ensure that website content is accurate and informative.

4. **Regulatory authority:**
   A regulator is a government agency or organization responsible for ensuring that a website complies with relevant laws and regulations. To avoid legal or financial consequences, it is important to ensure that your website meets all legal requirements.

2.2 Detailed Solution

The Asymptotic Notation website aims to provide an easy-to-use platform for users on asymptotic notation. Websites are designed and developed according to the needs and requirements of our stakeholders. Here's what we plan for each beneficiary:

1. **End User:**
Our goal is to create websites that are easy for end users to use and understand. Provides clear and concise explanations and examples of asymptotic notation to help users better understand the concepts. Also, make sure your website is visually appealing, responsive, and easily accessible from a variety of devices.

2. **Developer:**
For developers, we are going to develop the functionality of the website using HTML, CSS and JavaScript. We ensure that our website is optimized for performance and speed and is easy for users to navigate and use. Also, make sure your website code is clean and well-structured, which will make it easier to maintain and update in the future.

3. **Project manager:**
We plan to provide project managers with regular updates on the website's development progress, including milestones, timelines, and budgets. We also work closely with the project manager to ensure that the website meets the project objectives and any issues or concerns are quickly addressed.

4. **Field expert:**
We aim to ensure that the website content is accurate and informative for domain experts. We work closely with subject matter experts to ensure website content is up-to-date and relevant. It also provides a feedback mechanism for domain experts to suggest changes or improvements in website content.

5. **Sponsor:**
We plan to provide our backers with regular updates on the website's development progress, including milestones, timelines, and budgets. We also work closely with our sponsors to ensure that the website meets their expectations and that any issues or concerns are addressed quickly. Also, make sure the design and functionality of the website matches the brand and image of the sponsor.

TECHNICAL ANALYSIS

3.1 UML Diagrams

❖ Introduction:

UML (Unified Modelling Language) diagrams are graphical representations used in software engineering and systems analysis to visualize, specify, construct, and document the structure and behaviour of a system. UML diagrams provide a standardized way to communicate and understand the different aspects of a system's architecture.

- Communication:

These provide a common language for stakeholders, including developers, designers, clients, and end-users, to discuss and understand the system. They facilitate effective communication by using standardized symbols and notations.

- Visualization:

They offer a graphical representation of complex systems, making it easier to visualize and comprehend the system's structure and behavior. They help stakeholders gain a high-level overview and understand the relationships and interactions between different components.

- Analysis and Design:

They assist in analyzing and designing systems. They enable stakeholders to identify potential issues, design flaws, and missing functionalities early in the development process.

- Documentation:

They serve as documentation for the system. They capture the design decisions, architecture, and requirements in a visual format, making it easier to maintain and share knowledge about the system over time. UML diagrams also provide a reference for future enhancements or modifications.

- Blueprint for Implementation:

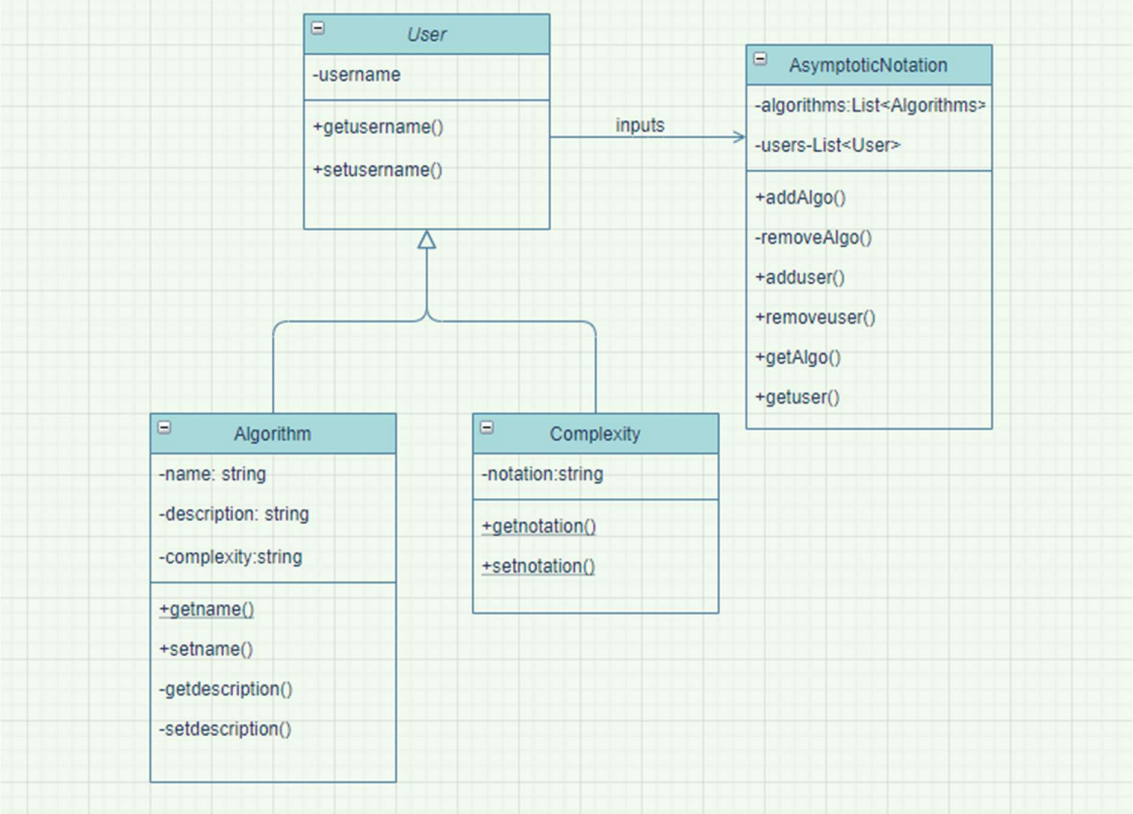
They can serve as a blueprint for developers during the implementation phase. They provide a detailed representation of classes, relationships, methods, and interactions, guiding developers in writing code that aligns with the intended system design.

- Testing and Validation:

UML diagrams can aid in testing and validation activities. They help identify test scenarios, define test cases, and ensure that the system meets the specified requirements. UML diagrams also assist in verifying that the system behaves as expected during different scenarios and interactions.

1. Class Diagram:

Class diagrams are a useful tool for visualizing the structure of object-oriented systems. They enable stakeholders to understand the relationships between classes, the behavior of the system, and the interactions between objects. Class diagrams can also help in designing and documenting the system's architecture, facilitating communication between team members, and guiding the implementation phase.



Description:

In this class diagram, we have four main classes: Asymptotic Notation, Algorithm, User, and Complexity.

The Asymptotic Notation class represents the website itself and contains two private attributes: algorithms (a list of Algorithm objects) and users (a list of User objects). It provides methods to add or remove algorithms and users, as well as retrieve the lists of algorithms and users.

The Algorithm class represents an algorithm and has private attributes such as name, description, and complexity. It provides getter and setter methods for these attributes.

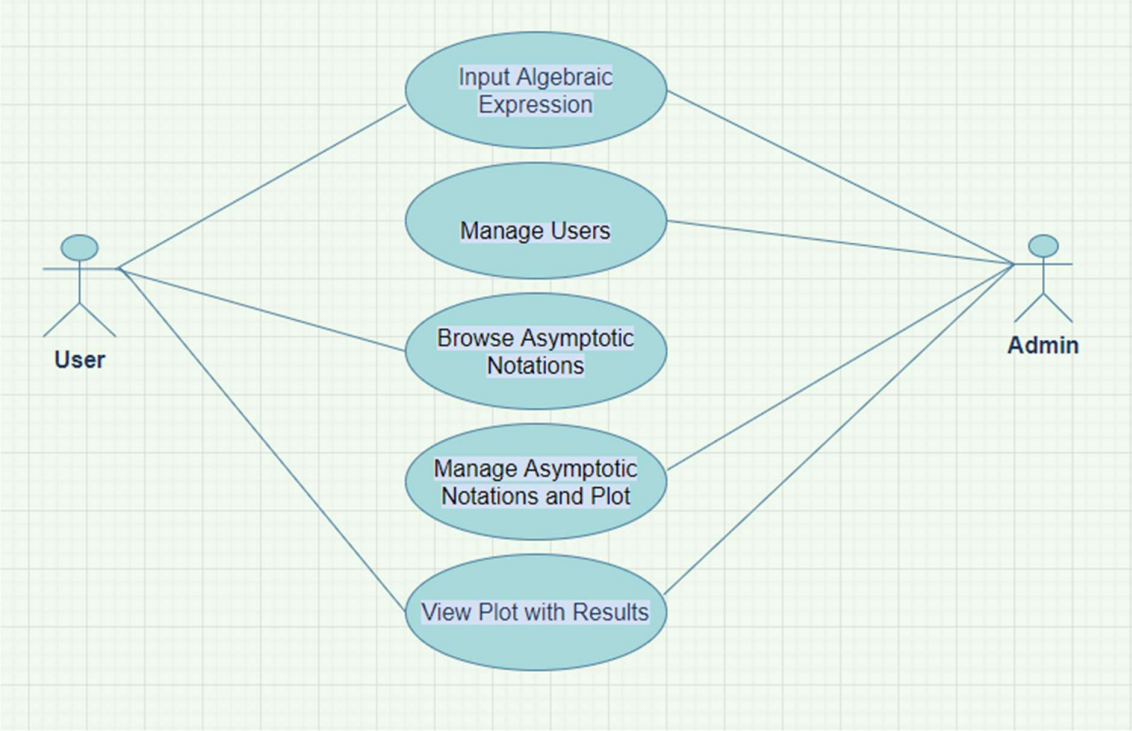
The User class represents a user of the website and has private attribute: username. It also provides getter and setter methods for the attribute.

The Complexity class represents the complexity of an algorithm and has a single private attribute notation (e.g., "O(n^2)"). It provides a getter and setter method for the notation.

However, the actual implementation can vary based on specific requirements and design choices.

2.Use Case:

A use case diagram is a type of UML diagram that illustrates the interactions between a system (or a software application) and its external actors. It represents the functionalities or behaviours of the system from the perspective of the users or actors involved .Use case diagrams also facilitate communication between project stakeholders, guide the development process, and serve as a basis for deriving test cases and validating the system's functionality.



Description:

In this use case diagram, we have two main actors: **User** and **Admin**.

The User represents a regular user of the website and has the following use cases:

- Search: The user can search for notations based on their names or descriptions.
- Input the algebraic expression: the user is responsible for the input of the function.
- View plot with results: The user can view details of a specific notation, including its name, graph and complexity.
- Browse Asymptotic Notation: The user can browse/submit notations on the website, including its name, description, and complexity.

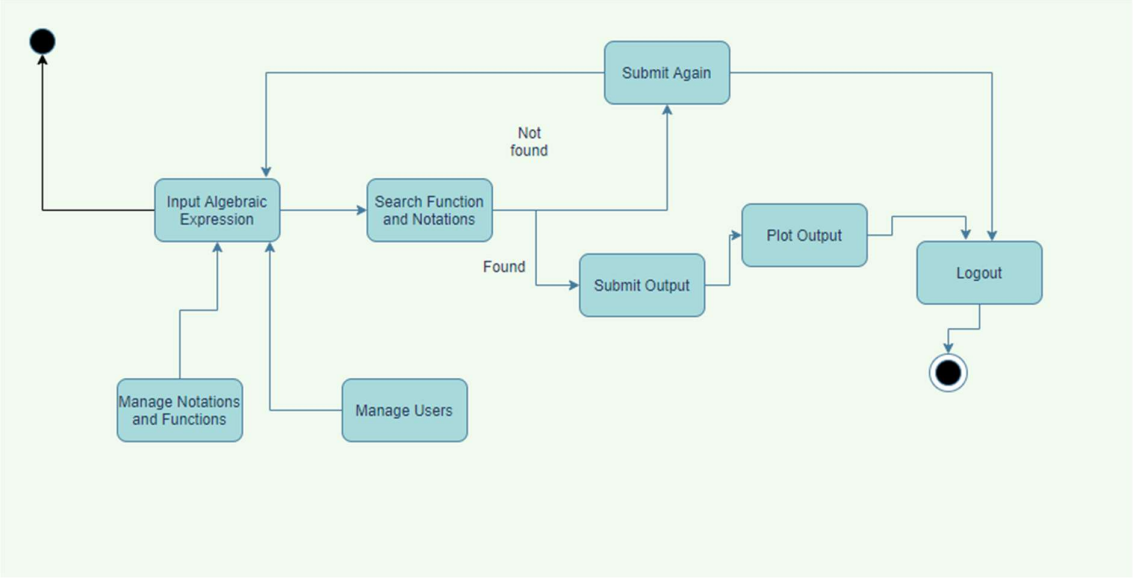
The admin represents an administrator of the website and has the following use cases:

- Manage Asymptotic Notations and plot: The admin can manage (add, update, or remove) algorithms from the website, including their names, descriptions, and complexities.
- Manage Users: The admin can manage (add, update, or remove) users of the website, including their usernames and emails.
- Search and View Algorithms.

The arrows in the diagram represent the communication or interaction between the actors and the system. For example, a user can perform a search, view algorithm details, or submit a new algorithm. The admin can manage notations and users within the system.

3.Activity Diagram:

Activity diagrams help stakeholders understand the sequence of activities, decision points, and control flows within a system. They can be used to model business processes, software algorithms, or complex behaviours. Activity diagrams facilitate communication, analysis, and design by providing a visual representation of the system's behaviour and the order in which activities are performed. They are also useful for identifying potential bottlenecks, exceptions, and alternative paths within a process.



Description:

In this activity diagram, we have several activities represented by rectangular nodes, and the arrows show the flow of control between them.

The flow of control starts with the users interacting with the website. They input the algebraic expression. They can perform activities such as writing the algebraic expression, viewing asymptotic notations details, or submitting a new algorithm for the same. On the other hand, the admin can manage various functions, notations and users. However, the actual implementation may require more detailed steps or decision points based on the website's functionality and requirements.

4.Sequence Diagram:

A sequence diagram is a type of UML diagram that illustrates the interactions and message exchanges between objects or components within a system over a specific period of time. It depicts the dynamic behaviour of the system, showing the order in which messages are sent and received between objects. Sequence diagrams are widely used in software development to model and visualize the flow of communication and collaboration between different elements of a system. They can also serve as a basis for generating code or designing test cases.

Description:

The user logins onto the website and inputs the algebraic expressions. Thus, initiates a search for algorithms on the website.

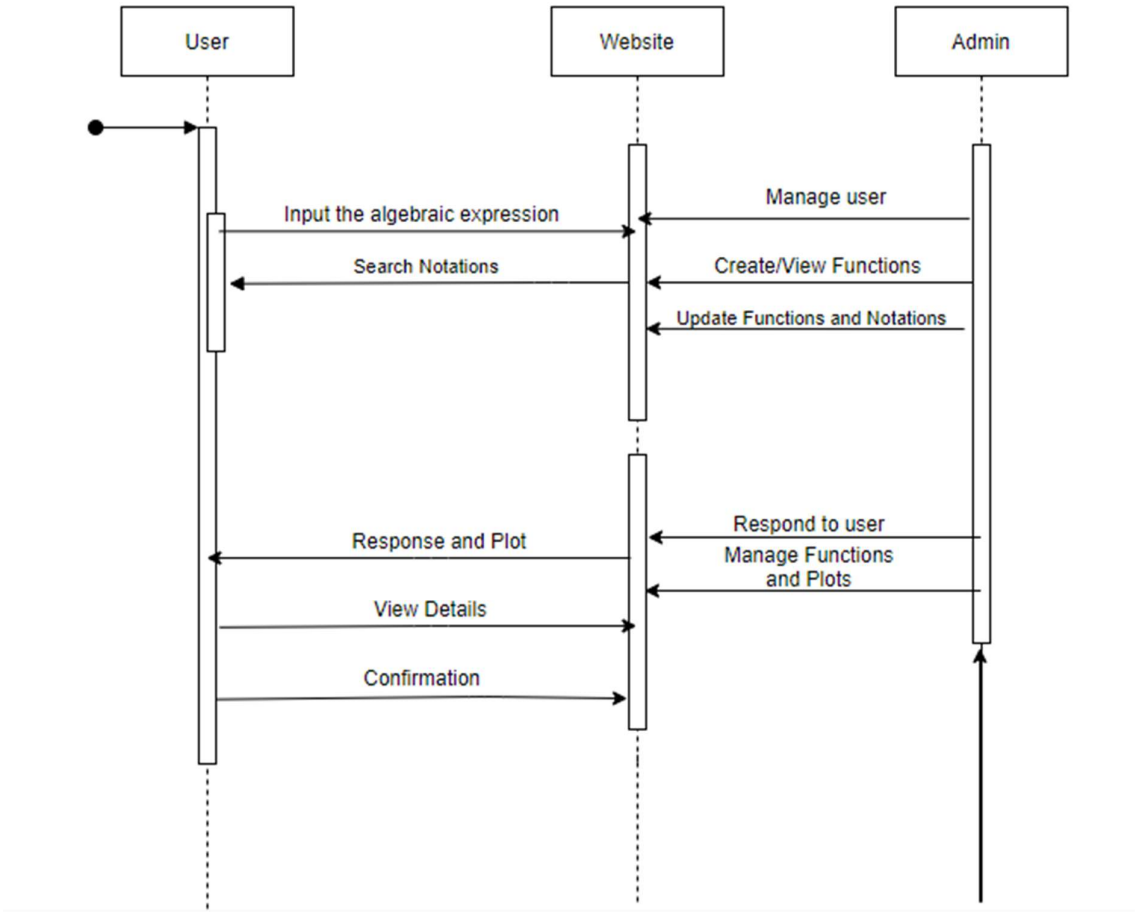
The Asymptotic Notation Website receives the request and searches for appropriate functions.

The admin manages the users of the website. It acknowledges the user and its input. The admin is responsible for creating and managing the algorithms. It also updates the existing list according to user inputs.

After the website has processed the input it displays the result e.g., Big Oh of the expression along with its graph.

Other functions are handled by the admin on the website.

The user views the details of the output and after confirming logs out of the application safely.


User
Website
Admin
Input the algebraic expression
Search Notations
Manage user
Create/View Functions
Update Functions and Notations
Response and Plot
View Details
Confirmation
Respond to user
Manage Functions
and Plots

.

3.2. Tech Stack Analysis

Choosing a tech stack for a software project involves selecting the set of technologies, frameworks, and tools that will be used to develop the application. The choice of a tech stack depends on several factors, including the project requirements, scalability needs, team expertise, budget, and time constraints.

In order to achieve required solutions, we have used appropriate Tech Stacks. These technologies have been chosen on the basis of the following few criteria:-

- Scalability
- Ease of Usage and Ease of Learning
- Time Required to build
- Efficiency
- Security

On the basis of the above mentioned criteria the Technologies used are:-

HTML

HTML provides a standardized way to define the structure and content of a webpage.HTML is supported by all major web browsers and platforms. It ensures that web pages can be rendered consistently across different devices and operating systems, making them accessible to a wide range of users.HTML supports responsive web design techniques, allowing developers to create mobile-friendly and adaptive layouts.HTML has been widely adopted and is supported by a large community of developers, designers, and browser vendors. This means there are abundant resources, tutorials, and frameworks available, making it easier for developers to learn, collaborate, and find solutions to common challenges.

Usage: Our Website of Asymptotic Notations uses HTML as the backbone language providing the essential structure, semantics, and compatibility necessary for creating the web page that is accessible, searchable, and compatible across different platforms and devices. The input box, headings, elements are fundamentally written in HTML.

CSS

CSS (Cascading Style Sheets) is a styling language used in web development to describe the appearance and layout of HTML elements on a webpage. CSS works by applying rules and styles to HTML elements, allowing developers to control various aspects of their presentation.

Usage:

Our website integrates CSS for creating visually appealing and interactive web page. It enhanced the user experience, enabled responsive design, and provided a high degree of customization and control over the appearance of HTML elements. The Graph Styling, fonts, colors, etc. were all designed through CSS. By separating the styling concerns from the structure and content of a webpage, there was easier maintenance, reusability, and consistency across multiple pages.

JAVASCRIPT

JavaScript is a powerful and flexible language that empowers developers to create dynamic and interactive web experiences. Its versatility extends beyond the web, with the ability to build server-side applications, mobile apps, games, and more. The extensive ecosystem of libraries, frameworks, and tools makes JavaScript a popular choice for a wide range of development tasks.

Usage: The website uses javascript for backend processing to enhance the interactivity and functionality of web page. The regular expressions are combined with appropriate js libraries and functions to give the right asymptotic notations. We used Chart.js library to create charts for visual representations of notations, thus making complex information more understandable and engaging for users.

- **REGEX – JAVASCRIPT MODULE**

  Regular expressions is a good approach when we are in need to match character combinations of a string. In javascript, we have used exec(), test() , match() functions of Regular expression module to get our desired algebraic expressions.

  This is how we have made our regular expressions:

```
// a*n!
const regex2 = /[\+\-\*/\s][\d\*]*n![\+\-\*/\s]/;
```

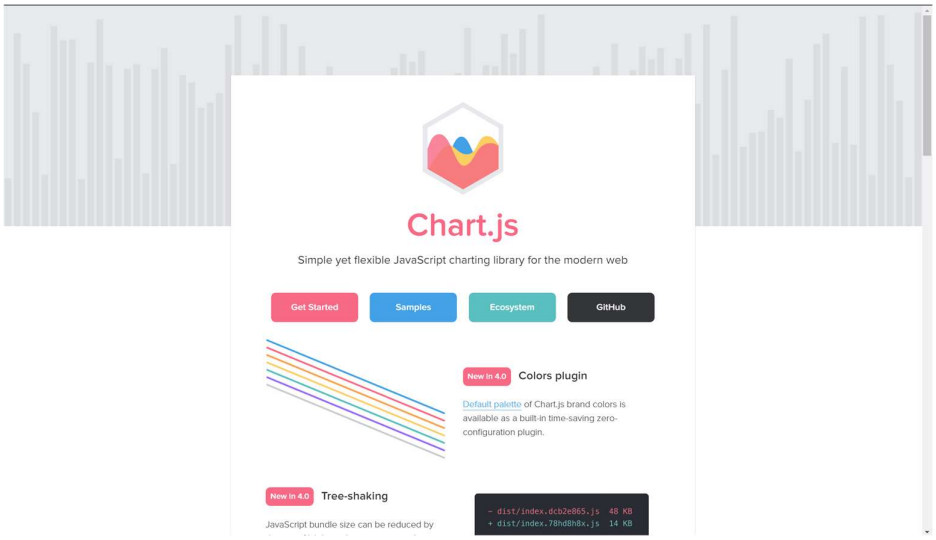
- **EXPR-EVAL JAVASCRIPT MODULE**

  It is a third-party module which is needed to be added dynamically in our code for evaluating Mathematical expressions after parsing the string. It provides support to inbuit mathematical operators. The evaluate() function is used for evaluating the strin. It requires parameter to be passed to it for evaluation along with its value.
  e.g.

```
let fnValue = expr.evaluate({ n: i });
let gnValue = exp.evaluate({ n: i });
```

- **CHARTJS – JAVASCRIPT LIBRARY**

  It is frequently used library in javascript which provide us with features that help in plotting and customizing the charts. This library provides interactive designs and simplified plots for complex problems. We can set the values according to our will for x and y axes. It enables us to make line plots, scatter plots, bar plots, histograms etc. It is easier to use than SVG which is more dynamic but difficult to use. For beginners, chartjs library is recommended always.



Visual Studio Code:

Visual Studio Code (VS Code) is a popular integrated development environment (IDE) developed by Microsoft. It is widely used for various programming languages and platforms, including web development, mobile app development, and cloud development.

We chose it as an IDE due to the following reasons:

- It gives a wide range of language support and helped integrate HTML, CSS and Javascript.
- It is lightweight and gave fast results.
- It provides cross platform support.
- It provides extensive customization options through its vast collection of extensions.

ECONOMIC ANALYSIS

We aim at providing a cost-effective yet easy to use and well-equipped website that can efficiently analyse algebraic functions and provide asymptotic notations. We have made a user-friendly website for inputting functions and viewing their asymptotic notations.

- With regards to the various development stacks used, they are freely available & hence what we require is- an Internet connectivity and a laptop costing; to ZERO.
- Students and Professionals can use it regardless of time, space and money.
- Scientists can do algorithm comparison and analysis easily without paying a single penny.
- We ensured usability across different devices and screen sizes thus scalability.
- We continuously update and improve the algorithm and website.

Our entire Website and platforms have been made with freely available YET Secure technologies and hence ZERO COST at all.

RESULT AND DISCUSSION

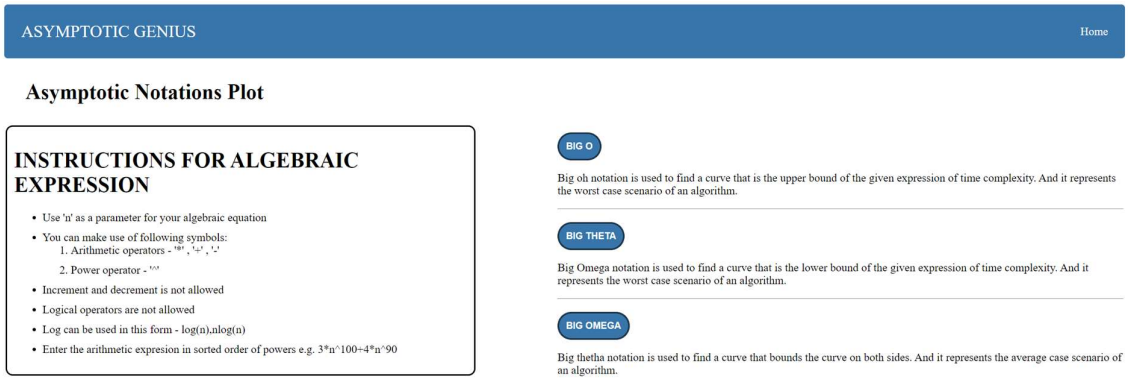
4.1 Website Usage Instructions

We have created a website “Asymptotic Genius” that will help you in getting the time complexity of the algebraic function along with its graph representation. We have introduced some new fields which will be giving the c and n0 values for the entered expression. We would also like to accept your suggestions and hope you find the web interface simple and easy to use.

Functionalities:

1. Instructions for Algebraic expressions:

We have made it simple for you to find the instructions to enter the algebraic expression as we have written them in the block that can be seen on the web page.



2. Navigate through web pages:

User can navigate through different pages for finding Big Oh, Big Omega and Big Theta. The Home button can help you to get back to the first page where you can start again with your choice for getting the plot.

3. Instant result:

Once you enter your algebraic expression you get an instant plot of the order of growth for your expression. The values of c and n0 are printed accordingly. The time Complexity is also shown along with the graph.

BIG O PLOT

INSTRUCTIONS FOR ALGEBRAIC EXPRESSION

- Use 'n' as a parameter for your algebraic equation
- You can make use of following symbols:
  1. Arithmetic operators - '*' , '+' , '-'
  2. Power operator - '^'
- Increment and decrement is not allowed
- Logical operators are not allowed
- Log can be used in this form - log(n),nlog(n)
- Enter the arithmetic expresion in sorted order of powers e.g. 3*n^100+4*n^90

Enter the Function: 3n^3+ n
C value: C Value
n0 value: N-Not Value

PLOT

Result:

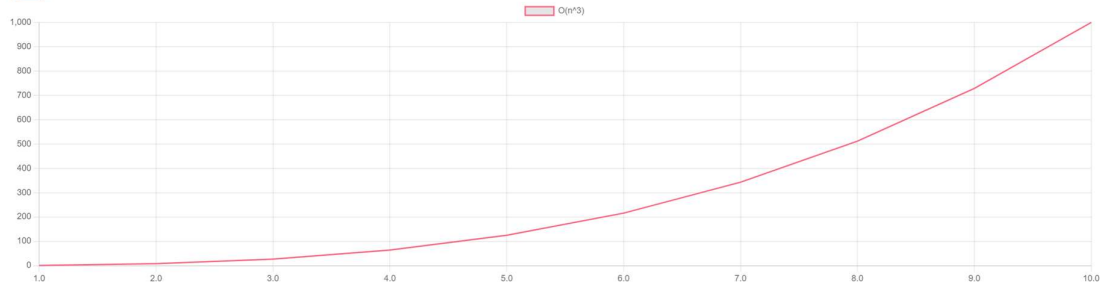
BIG O PLOT

INSTRUCTIONS FOR ALGEBRAIC EXPRESSION

- Use 'n' as a parameter for your algebraic equation
- You can make use of following symbols:
  1. Arithmetic operators - '*' , '+' , '-'
  2. Power operator - '^'
- Increment and decrement is not allowed
- Logical operators are not allowed
- Log can be used in this form - log(n),nlog(n)
- Enter the arithmetic expresion in sorted order of powers e.g. 3*n^100+4*n^90

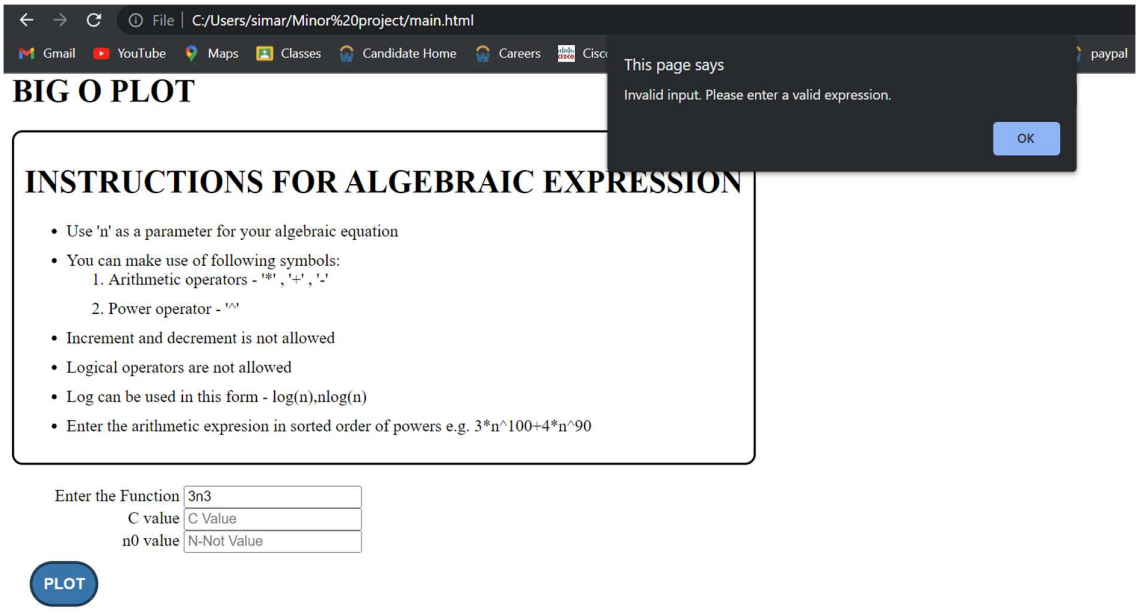
Enter the Function: 3*n^3 + n
C value: 3
n0 value: 1

PLOT



4. Alert Message:

An Alert is generated when user tries to input an invalid expression that does not follow the instructions or some hidden case is there.

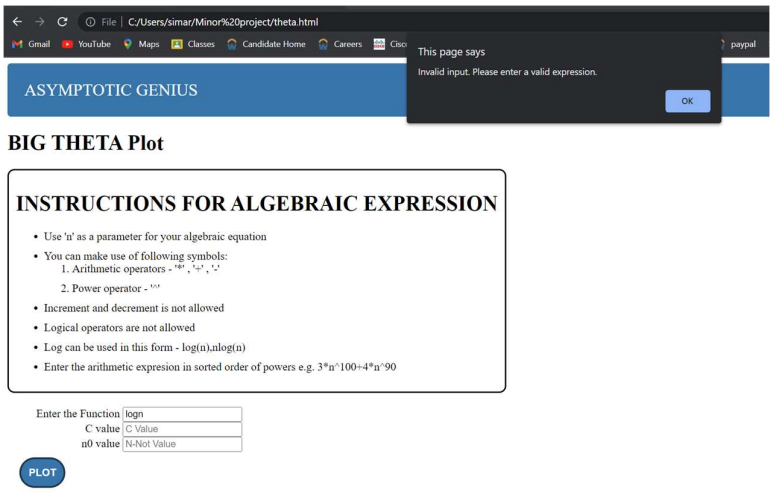

File | C:/Users/simar/Minor%20project/main.html
Gmail
YouTube
Maps
Classes
Candidate Home
Careers
paypal
This page says
Invalid input. Please enter a valid expression.
OK
BIG O PLOT
INSTRUCTIONS FOR ALGEBRAIC EXPRESSION
Use 'n' as a parameter for your algebraic equation
You can make use of following symbols:
1. Arithmetic operators - '*' , '+' , '-'
2. Power operator - '^'
Increment and decrement is not allowed
Logical operators are not allowed
Log can be used in this form - log(n),nlog(n)
Enter the arithmetic expresion in sorted order of powers e.g. 3*n^100+4*n^90
Enter the Function
3n3
C value
C Value
n0 value
N-Not Value
PLOT

4.2 Risk Analysis

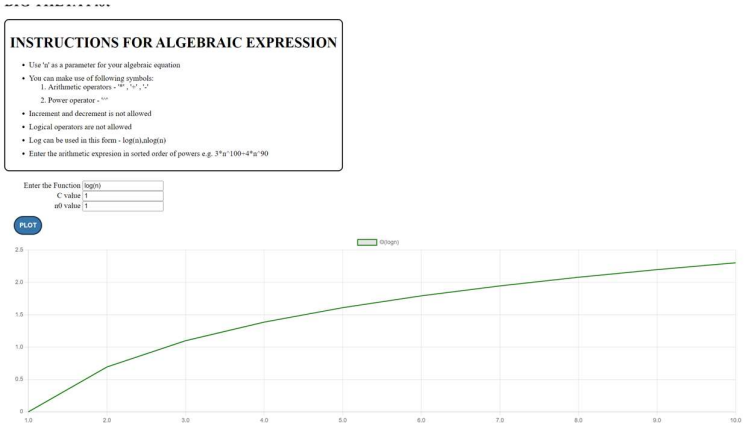
There are always risks involved when you are developing and designing something from scratch. We have also highlighted the risks involved in our project – Asymptotic Genius.

1. Risk of Scaling while Plotting the function(X and Y axes scaling error)
2. Algebraic expressions that are valid gave alert message (e.g.$log(log(logn))$ etc)
3. Risk of unaccepting the parameter that is not n.
4. Risk of getting wrong c and n_0 values (e.g. $logn$ etc)

The major concern is error 1 as some complex algebraic function leads to failing of correct plot due to scaling issue. The other errors are dealt with as we have narrow down our approach to some simple expressions. We need to add some more regular expressions that will cover the hidden cases like above. As of error 4, We have set the c and n0 values to 1 in case they approaches to infinity or are undeterministic.
Hence, We have minimized the risk level by various discussions with our mentor. But, still we are adding more regular expressions that cover most algebraic expressions. Our Website is till in testing phase.



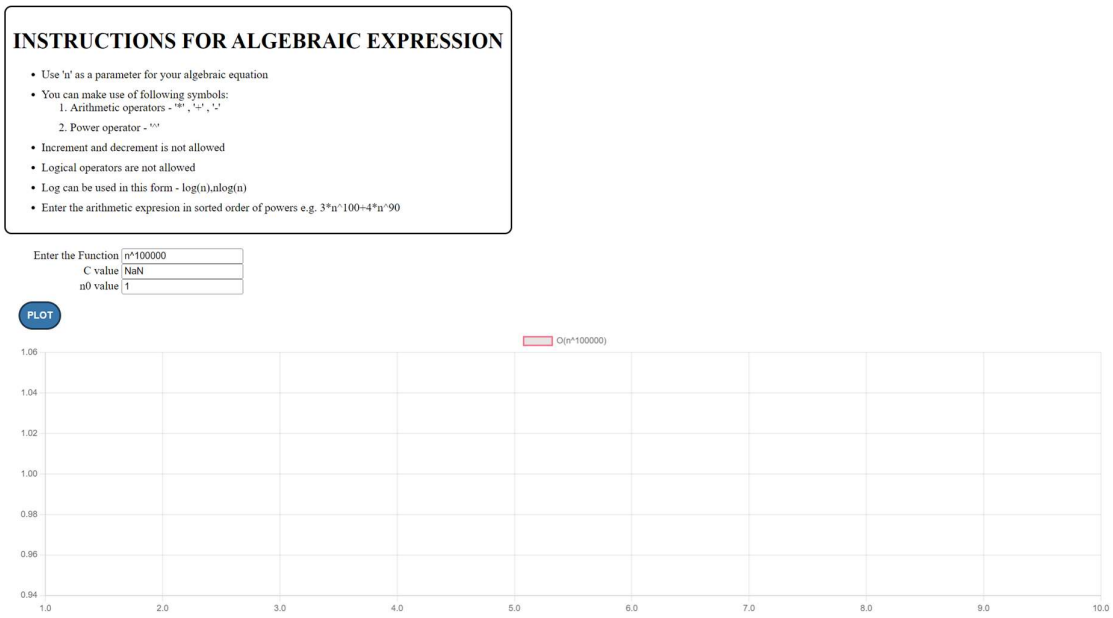
e.g. It shouldn't give us error but it is saying invalid input.



4.3 Deployment and Testing Status

Our website is still in testing phase. We are testing it for various algebraic expressions like $3 * n^{10000}, log(logn)$ etc and some more hidden cases that are leading to errors. We were also facing problems in plotting the graph along with c and n0 values as it is leading to scaling problems.
We had a talk with our mentor Dr. Lalatendu Behera, Assistant Professor, Department of Computer Science and he guided us to simply mention the results of c and n0 in the output fields for the user to use and plot the order of growth according to c and n0 values set to 1 each. Thus, Now, this issue is resolved and now we are looking into regular expressions. Many, general cases and Once, it is solved our website will be ready for deployment.

Below shows testing of high powers expression and it is failing to plot.



SOCIAL AND ENVIRONMENTAL IMPACT

- Making up of a website has reduced the overload of high amount of paper-work involved in determining asymptotic notations like analysing appropriate algorithms, tasks, time complexities etc manually.
- Immediate results to all students, professionals and scientists using the website and their storage for future retrieval.
- The website prioritizes user experience by providing intuitive navigation, clear content, and responsive design contribute to positive social impact.
- While websites can connect people globally, the digital divide remains a challenge. Our website crosses this digital and minimizes the gap between those with access to technology and those without.
- We have ensured the privacy and security of our users.

By prioritizing accessibility, user experience, sustainability, and ethical practices, the website can have a positive impact on society and contribute to a more environmentally conscious digital landscape.

CONCLUSION

Our Project- Asymptotic Genius is complete and fully functional. Throughout the report, various research methods, data analysis techniques, and credible sources were employed to gather and present relevant information. The objectives have been achieved to a great extent. The trial has also been successful.

Conclusively, we have achieved the following:

- To provide an easy to use, economically feasible website that can efficiently analyze algebraic functions and provide asymptotic notations.
- This is in-line with the pathway to Digitization where everything can be done with a button-click.
- We have learned a lot about various technologies, asymptotic notations along with team-work and collaboration.

While this project has strived to provide a comprehensive understanding of asymptotic notations, it is important to acknowledge the limitations encountered during the process. These limitations may include restricted access to data, time constraints and complexities, or the scope of the study of functions. Such limitations should be taken into account when interpreting the findings and drawing conclusions.

It is hoped that it will stimulate meaningful discussions, inspire new ideas, and contribute to the academic or practical advancement of the field.

REFERENCES

- Horowitz and Sahani, “Fundamentals of data Structures”, Galgotia Publication Pvt. Ltd., New Delhi.
- R. Kruse etal, “Data Structures and Program Design in C”, Pearson Education Asia, Delhi-2002
- M. Tenenbaum, “Data Structures using C & C++”, Prentice-Hall of India Pvt. Ltd., New Delhi.
- Web Development Tools: W3Schools, Javatpoint, Tutorials Point.
- *Regular Expression match in javascript*. (n.d.). Stack Overflow. https://stackoverflow.com/questions/9901836/regular-expression-match-in-javascript
- *Chart.js*. (n.d.). Chart.js. https://www.chartjs.org/docs/latest/
- Wikipedia For Basic Introduction of Technology and Diagrams
- Images from simplilearn, Wikipedia.

Thank You. Your suggestions are most Welcomed.

----------END OF REPORT----------