Honors Precalculus 2024-2025

Project: Approximating Values of Trigonometric Functions

You and your teammates will work together to design and write a program that uses polynomials to approximate values of trigonometric functions. Deliverables (via Canvas):

- **(1p)** a **design document (spreadsheet)** describing the functions that will be written for your program; be as specific as possible and include descriptions of parameters, return values, and the task performed by each function. Template provided.
- **(1p)** a **write-up** explaining the math that will be implemented in your program. You should describe how to find the polynomial approximation to the sine function and how to relate trig functions to each other using symmetries in the unit circle. Your audience is another Honors Precalculus student who maybe missed the day when we introduced the project and learned about polynomial approximations.
- **(0.7p)** At least two example calculations: use input values that will exhibit different, somewhat interesting behavior and use a flowchart to show how your program and its functions would handle these inputs. Example provided.
- **(1p)** a **Python program** to accomplish these tasks; see overview, requirements and sample runs below.
- **(0.3p)** an individual **evaluation** of the process and team contributions; google form will be provided.

Collaboration and AI assistance – Lakeside Yellow

This is a team project. The intent is for your team to develop all the mathematics and code on your own. While you can discuss math or coding issues with other groups, you should not be sending each other pieces of code; viewing another student's code; or dictating code to each other.

You cannot use an AI assistant to write any of the code for you. However, you are welcome to use AI assistance to learn general concepts. For example, "show me how to write a for loop in Python" is ok; "write a function to approximate sine using polynomials" is not allowed.

If you are not sure whether something is allowed or not, or if you've unintentionally found too much AI assistance, check in with your teacher!

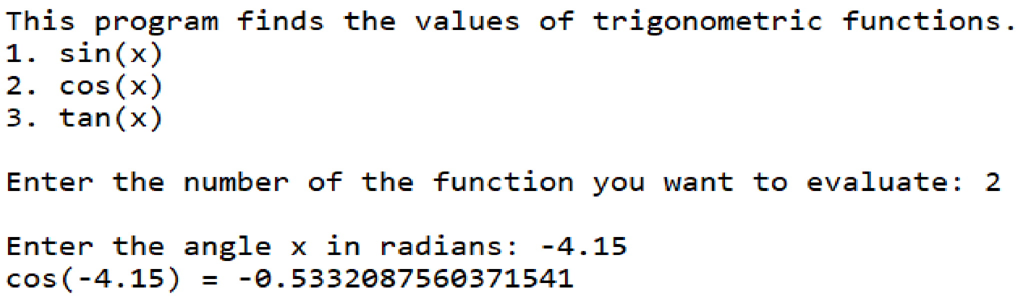
Program Overview

A ***transcendental function*** is a function which is not an algebraic function. In other words, a function which "transcends," i.e., cannot be expressed in terms of algebra. Examples of transcendental functions include the exponential function, the trigonometric functions, and the inverse functions of both. You will design and then write a program to approximate values of trigonometric functions. However, you cannot just use np.sine(). You are only allowed to use arithmetic operations to find these values. Specifically, the polynomial approximation technique you just learned about can serve this purpose.

Requirements and example:

- A core function that uses a polynomial approximation to find **the value of sine only for angles between 0 and π/4 radians.**
  - The core function must use a loop to generate the values from each term of the polynomial. There should be a parameter inside the function that lets you set the degree of the polynomial you want to use. You want to set it to higher than 11 to get good accuracy, but anything more than 19 might be overkill.
  - This function should do nothing more than find a value of sine to high precision within the specified, limited range.
- You must write other support functions that manage the rest of the details, some of which include finding values for inputs that are outside the allowed range of the core function **or** finding outputs of other trig functions (cos and tan). These other functions should **not** calculate anything using a polynomial approximation. That only happens in the core function. Each of these support functions should have one and only one clear purpose, but they can and should call other functions and/or the core function.
- The main program will be pretty simple – the "calculations" section of your code should just be coordinating calls to the functions - no actual processing should be happening in this section. All the action should happen in the functions.
- You may import the *numpy* library, but you CANNOT use any of its functions to compute trig functions (and you also can't use the Python *math* library's versions of these functions).
- You will need to define *π* for your program. This should appear just below your import statements, but before your functions: *PI = np.pi*

Here is a typical run of the program. You may use this format or not, so long as the user can clearly understand what is being requested and displayed.



```
This program finds the values of trigonometric functions.
1. sin(x)
2. cos(x)
3. tan(x)

Enter the number of the function you want to evaluate: 2

Enter the angle x in radians: -4.15
cos(-4.15) = -0.5332087560371541
```

Hints:

- At this point you should know how to relate a value of any of these three functions at any angle to a value of sine between 0 and π/4. That said, there are a variety of details you will need to manage. *Make sure you include two or three example calculations in your write-up.*
- Debug extensively! You can check any output values using your calculator or desmos.
- If you modularize your program well, you can distribute the writing of the functions to different team members.