Author: Nikhil Dhanji & Manoj Dhanji

Date: Feb 06 2021

Version: 1.0

Description:

This program provides an intuitive interface to assemble polynomial functions by creating terms (simple or complex) and linking them to create larger polynomials. A polynomial is of the form a0xn + a1xn-1+ .... A simple term consists of a coefficient (real number) and an exponent (real number). 2(x)2 and √x are examples of simple terms. A complex term on the other hand, consists of a trigonometric function that accepts an argument and is multiplied by a multiplier. Both the argument and the multiplier could be simple or complex. (x)2 sin2(cos(x)) is an example of a complex term.

Architecture:

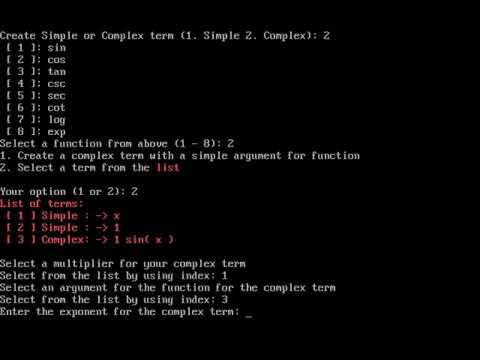
This program has been implemented using several user-defined types. The user-defined type, **simple** consists of a coefficient and an exponent; both real numbers. Likewise, **complex** contains a function (*trigonometric, ln or e*) that requires an argument; the **complex** also has a multiplier and an exponent (*a real number*). The multiplier and the argument can be a **simple** or a **complex**. Additionally, there is a user-defined type **term**. It holds a reference to either an object of **simple** or **complex** type. Since QBasic programming language does not support polymorphism, the **term** type uses a discriminator to indicate if it refers to **simple** or **complex** type.

The user builds **term** objects into a list. The **term** can contain a **simple** or a **complex** type. The complex type has index (pointers) that point back to multiplier or argument into the list. The user assembles a larger polynomial out of the **term** objects.

The user has an option to build a rational expression (p/q where p and q are polynomial expressions)

The program then prompts for the lower and upper values of the domain to plot the function. The user also has the choice of plotting the derivative of the function and or calculate the area under the curve.

Execution:

[](https://www.youtube.com/embed/Dz7A48X_hL8?feature=oembed)