To do list:

04/08/2021

Sine function put it in the sine wave in form of vector

Taking a sine of a function and then you save that

Google for a sample file of a sine wave sample data

What does “a dereferenced null pointer value of a polymorphic type” mean

Questions from Object Oriented Programming:

1. Time 4:43 - runtime: State of the derived object
2. Why would one make constructors private?
3. Time 7:13 why is the virtual destructor intializer by default  
   virtual ~Logger ( ) = default;  
   why not ~Logger ( );
4. Slides on timer 8:28 and 8:33 - unable to understand what the code is doing
5. Timer 10:00 - “not all derived types are subtype” did not understand that
6. Timer 12:00 - what does requiring more and promising less mean?
7. Timer 16:21 - I did not understand the concept of object slicing that has been expalined in the video. From what I have learned object slicing is when you intialize and object of base class with a derived class object. Now when you call this base class object it preserves the base class and gives the base class value rather than being over written by the derived class during copy.
8. Timer 17:30 - he calls size of as illogical as its gona tell us how is the animal not that how big is the lizard but..the animal is pointing to the lizard so logical it should give the size of lizard.
9. Timer 22:23 - assignment operators protected, I thought it was so that no one can directly access the data member but instead he says, “to prevent assignment from pointer to object.” Unable to understand.
10. Timer 23:57 - they say make virtual function non public but how would the compile override it if they are not public?
11. Timer 30:17 - why is asset not forcing the value of 1?
12. Timer 36:53 - The inner most scope has already define foo(int) so why did it give false? Why is the int foo hidden?
13. Slide at timer 40:52 Cannot understand the code written

Questions from function and class templates

1. Timer 31:47 - is this an example of composition where list class has a iterator class?
2. Timer 41:47 - are we using template to assign return type too by using the line int i =f<int>( );
3. Cannot understand dependent and non dependent names.