Ask me anything

Mentimeter

0 questions
0 upvotes

Should a Square inherit from Rectangle or vice versa? Why?

Mentimeter

square from rectangle, because it's a specialization?

square should inherit from rectangle because it is a special case of a rectangle

depending on the usage, usually shouldn't because they should inherit from shape

According to Subclassing, rectangle inherit from square, in is_a, a square inherit from rectangle

because rectangles may have further attributes and methods

5

What is the difference between subtyping and specialization?

Mentimeter

Subtyping in a way is a conceptual specialization

subtyping : substitutabilityspecialization : says that one thing is a special case of another

subtype can substitute a type, specialization is an extension of something else

subtyping can also be used to just ensure compatibility of multiple(!) different types, whereas is-a vertical chain of is-a relations

4

Can a dynamically-typed language be polymorphic, or only a statically-typed one?

Mentimeter

A dynamically typed language is always polymorphic, because the variables accept objects of various (all) types.

No

<---- i agree with this person

I think one is independent of other. At least if you consider ad-hoc polymorphism

4

How is an object like a closure?

Mentimeter

as long as attributes are private, they are hidden to the outside, just like in closures

both give a way to hide (make private) certain parts

It is a variable namespace that may overwrite variables with greater scope, thus why in cpp for example "this" exists. Also self in python is explicitly the closure.

Both are environnment that binds free names within it

closure gives you access to an outer function's scope from an inner function.

5

Should an “intersect” methods for a Shape class and subclasses be covariant in its arguments? Contravariant? Neither?

Mentimeter

What does intersect do ?

The issue is that you can create new shape by intersecting known shape

Probably contravariant, becaus if it was covariant, you could only intersect two shapes of the same type.

The output is a new shape so it should be invariant.

neither because the initial set is smaller than the final set

Neither, as the return type is not known and probably a complex shape

6

Can a dynamically typed language support overloading? Why or why not?

Mentimeter

yes.

It is independant again so yes

<-----I agree. That's normal

It could by having no strict types for the function signature and then do type checking of the parameters it got and call different methods

Yes, haskell does it for example (pattern matching)

no - if i overload methods i have to say what static type they have - but there is no static typing here

Python does it with type hints.

yes, even if it's dynamically typed, it still can have different types, that require the same overloaded method.

8

In what way does a class represent a family of types?

Mentimeter

Because a class is a generator of type

That a class can have subclasses which again represent a subtype of the type of the original (super) class

2

Last chance for questions