

Czy monada może być osobliwym przypadkiem obiektu OOP?

(Abstrakt)

metoda: odczytanie monadologii przy użyciu pojęć i twierdzeń obiektowego paradygmatu programowania (OOP)

wynik: przy pewnych zastrzeżeniach monada może być rozumiana jako szczególny przypadek obiektu OOP

1. Paradygmat obiektowy (OOP):

- a. program jest zbiorem obiektów wymieniających ze sobą komunikaty,
- b. klasa to typ danych zdefiniowany przez użytkownika (swoisty wzorzec),
- c. obiekt to instancja klasy (swoisty indywidualny przypadek wzorca),
- d. obiekt złożony jest z atrybutów określających, jakie cechy ma obiekt oraz z operacji określających możliwe działania podejmowane przez obiekt,
- e. dziedziczenie to relacja między klasami polegająca na tym, że klasa podrzędna dziedziczy całą strukturę klasy nadrzędnej,
- f. polimorfizm to mechanizm umożliwiający działanie jednej operacji na różnych typach danych,
- g. hermetyzacja/enkapsulacja, czyli ochrona obiektu przed zmianą jego stanu przez inne obiekty (komunikacja przez interfejsy),
- h. abstrakcja, czyli działanie prowadzące do uproszczenia i uogólnienia rozważanego problemu.

2. Monadologia:

- a. istnieje jeden rodzaj substancji,
- b. istnieje nieskończona liczba substancji - *lex continua*,
- c. substancje są niematerialne,
- d. zmiany w substancji są niezależne od innych substancji (monada nie ma okien),
- e. ciało jest sposobem, w jaki jedna monada zjawia się drugiej.

3. Monada vs obiekt OOP:

lp.	monada	obiekt OOP
1.	Jest substancją prostą, czyli bez części.	Obiekty mogą być proste lub złożone.
2.	Jest elementem agregatu.	Może być elementem agregatu albo sam jest agregatem.
3.	Jest nierozciągła.	Jest przedmiotem logicznym (abstrahując od jego fizycznej implementacji), więc jako taki nie ma rozciągłości.
4.	Powstaje lub ginie za jednym zamachem.	Jest „stwarzany” przez specjalną metodę zwaną konstruktorem (niszczony przez metodę destruktor).
5.	Jest jestestwem, więc ma jakieś właściwości.	Ma atrybuty i operacje/metody.
6.	Jest różna od innej monady.	Jest różny od innego obiektu.
7.	Jest zmienna.	Może ulegać zmianie.

8.	Zasada jej zmiany jest wewnętrzna - to apetycja: przejście od jednej percepcji do drugiej.	Zmiana stanu obiektu może wynikać z działania samego obiektu lub być skutkiem komunikatu innego obiektu.
9.	Jej percepcje i zmiany nie są mechaniczne.	Jego zmienność może być <i>quasi</i> -mechaniczna.
10.	Jest źródłem własnych czynności (bezcieleśny automat).	Jest bezcieleśnym automatem.
11.	Rzeczywistość faktyczna jest zbiorem monad, których harmonia została wprzódy ustanowiona.	Program jest zbiorem obiektów, które komunikują się ze sobą, a struktura obiektów i całego programu została z góry ustalona.
12.	Bóg Monada stwarza monady.	Obiekty tworzone przez metodę zwaną konstruktorem.

dodatkowe:

A. Rozszerzenia:

- a. Zasada podstawiania Liskova (typ bazowy powinien być w pełni zastępowalny typem pochodnym) zastosowana do monadologii umożliwia zachowanie tezy o jednym rodzaju substancji z dopuszczeniem istnienia różnorodności klas, które musiałyby dziedziczyć po jednej klasie bazowej.
- b. Modyfikatory dostępu (w OOP od publicznych po prywatne) w przypadku monada zredukowane byłby do poziomu najniższego, czyli prywatnego.
- c. W programowaniu przekazywanie argumentu do funkcji przez wartość sprawia, że funkcja działa wyłącznie na kopii wartości, a zatem gdyby monady komunikowały się ze sobą poprzez przekazywanie wyłącznie kopii wartości, to zachowana byłaby teza, że żadna monada bezpośrednio nie oddziałuje na inną monadę, a jednocześnie możliwa byłaby harmonia „wprzódy nieustanowiona”.

B. Trudności:

- a. Absolut: jeżeli jest obiektem, to jego podstawą jest klasa, a jeżeli jest klasą, to bardziej fundamentalny jest system „komputerowy”, w którym klasa może zaistnieć,
- b. status ontologiczny klasy,
- c. Czysto teoretyczna interpretacja obiektowa może być uzgodniona z zasadą ciągłości i z nieskończoną liczbą monad. Obiektowa implementacja musi naruszyć zasadę ciągłości, a ilość obiektów zawsze jest skończona.

C. Rozjaśnienia:

- a. Monady może nie mają okien, ale muszą mieć interfejsy (struktura pośrednicząca między obiektem a jego otoczeniem), bo:
 - i. w ogóle o nich dyskutujemy, więc są poznawczo dostępne,
 - ii. monada ujawnia się monadzie poprzez ciało,
 - iii. ustanowienie harmonii między monadami zakłada jakąś formę ingerencji w monady.
- b. Relacja agregacji w OOP, będącej modelem relacji „ma”, pozwala zrozumieć, jak jest możliwe, że ciało jest agregatem monad.