Przegląd Filozoficzny – Nowa Seria
R. 22: 2013, Nr 2 (86), ISSN 1230–1493
DOI: 10.2478/pfns-2013-0070

Bogdan Dziobkowski

Nonfaktualizm a kognitywistyka

***Słowa kluczowe*:** *kognitywistyka, S. Kripke, nonfaktualizm, obliczeniowa teoria umysłu, sceptyczny paradoks Wittgensteina*

Od połowy dwudziestego wieku filozoficzna refleksja teoriopoznawcza jest wypierana przez badania naukowe. W tekście „Epistemologia znaturalizowana" Quine tak o tym pisze:

> Epistemologia, czy też jakiś jej odpowiednik, staje się po prostu działem psychologii, a więc nauki. Bada ona naturalne zjawisko, mianowicie fizyczny podmiot ludzki. Podmiotowi temu dane jest pewne eksperymentalnie kontrolowane wejście – na przykład pewien układ promieniowania świetlnego o określonej częstotliwości – a po odpowiednim czasie podmiot daje jako wyjście opis trójwymiarowego świata zewnętrznego i jego historii. (Quine 1986: 118)

Według Quine'a powinniśmy porzucić marzenie o filozoficznym ugruntowaniu wiedzy i skupić się na opisie mechanizmów poznawczych przetwarzających dane, które „fizyczny podmiot ludzki" otrzymuje na wejściu (pobudzenia powierzchni sensorycznych), w informacje, które znajdujemy na jego wyjściu (teorie opisujące świat).

Zarysowany przez Quine'a program badawczy realizowany jest obecnie przede wszystkim w obrębie kognitywistyki, przy czym podmiot postrzega się zwykle jako pewien rodzaj komputera, a procesy poznawcze jako stany obliczeniowe.

Celem niniejszego tekstu jest wykazanie, że preferowane przez kognitywistów obliczeniowe koncepcje umysłu generują regres w nieskończoność i tym samym nie mają mocy eksplanacyjnej. Do uzasadnienia tej tezy wykorzystam nonfaktualną interpretację filozofii późnego Wittgensteina, przedstawioną przez

Saula Kripkego w książce *Wittgenstein o regułach i języku prywatnym*. Oprę się przede wszystkim na krytyce tzw. analizy dyspozycjonalnej, jako jednego z możliwych sposobów bezpośredniego rozwiązania sceptycznego paradoksu Wittgensteina.

1. Nonfaktualizm

W uwadze 185 *Dociekań filozoficznych* Wittgenstein przedstawia następującą sytuację. Osoba *A* uczy osobę *B* rozwijać ciąg liczb według określonej reguły, na przykład „+2", przy czym wszystkie objaśnienia i ćwiczenia dotyczą liczb mniejszych niż 1000. Po przejściu przez ucznia serii testów, osoba *A* uznaje, że rozumie on zasadę ciągu. Następnie każe uczniowi rozwijać ten ciąg począwszy od liczby 1000. Wykonując to polecenie uczeń pisze: 1000, 1004, 1008, 1012, itd. Czy jego działanie jest zgodne z opanowaną wcześniej regułą „+2"? Wydaje się, że nie. Powiemy, że w odpowiedzi na polecenie *A* powinien on napisać: 1000, 1002, 1004, 1006, itd. Ale jakie fakty podważają pierwszą i uzasadniają drugą z tych odpowiedzi? Skoro uczeń opanował regułę „+2" ćwicząc na liczbach mniejszych niż 1000, skąd ma wiedzieć, co robić, gdy nauczyciel poleci mu przekroczyć tę granicę? Na pierwszy rzut oka problem ten nie wydaje się specjalnie skomplikowany. Mówimy, że uczeń powinien po prostu postępować zgodnie z opanowaną wcześniej regułą. Czym jest jednak ta reguła? Jeżeli utożsamimy ją z tabelą zawierającą skończoną liczbę przykładów rozwinięcia ciągu, wyjście poza te przykłady będzie zawsze krokiem w nieznane. Tabela nie powie nam, co robić w przypadkach, które nie zostały w niej ujęte. Przy takim rozumieniu reguły „+2", wszystko, co zrobi uczeń po przekroczeniu liczby 1000, będzie z nią zgodne. Dlatego mówiąc o regule, zwykle mamy na myśli nie tabelę, tylko pewien mechanizm, który jest w stanie wygenerować określoną odpowiedź w dowolnym nowym przypadku. Jednak według Wittgensteina, takie ujęcie również nie rozwiązuje naszego problemu. Przyjmijmy, że ten mechanizm ma formę wzoru. By zrozumieć, jak on pracuje, niezbędna jest interpretacja. Ale interpretacja ta jest kolejną regułą, która, aby być zrozumianą, wymaga nowej interpretacji itd. A więc niezależnie od tego, czy regułę utożsamimy z tabelą, czy z algorytmem, „reguła nie może wyznaczać sposobu działania, gdyż każdy sposób działania daje się z nią uzgodnić" (Wittgenstein 1972: § 201).

Analogiczny przykład znajdujemy w książce Kripkego *Wittgenstein o regułach i języku prywatnym*. Jan, podobnie jak każdy z nas, posłużył się symbolem „+" skończoną ilość razy. Przyjmijmy więc, że przed chwilą t_n nigdy nie wykonywał on operacji na liczbach większych niż 56. Po tej chwili pytamy go, ile jest „68 + 57", a on bez namysłu odpowiada: „125". Wydaje się, że jest to odpowiedź poprawna zarówno w sensie arytmetycznym, jak też metajęzykowym. Wprawdzie Jan przed chwilą t_n nie używał symbolu „+" w kontekście liczb większych od 56, ale łączył z nim funkcję, która daje określony wynik dla dowolnie dużych argumentów. W omawianym przypadku funkcja ta wygenerowała właściwą odpowiedź. Skąd jednak wiemy, jaka to była funkcja? Nazwijmy „kwodawaniem" funkcję, która dla argumentów mniejszych niż 57 daje wartości takie same, jak funkcja dodawania, a w pozostałych przypadkach jej wartością jest zawsze 5. Jakie fakty z historii Jana pozwolą rozstrzygnąć, czy przed chwilą t_n dodawał, czy kwodował? Sceptyk utrzymuje, że nie istnieją takie fakty i tym samym zdanie:

> Jan, używając przed chwilą t_n znaku „+", miał na myśli dodawanie

nie ma warunków prawdziwości.

2. Maszyna J

Na gruncie obliczeniowych koncepcji umysłu podmiot jest rozumiany jako pewnego rodzaju komputer. Zastąpmy więc Jana maszyną J. Na jej wejściu wprowadzamy ciąg symboli: „2", „+", „4", na wyjściu otrzymujemy „6". Jaką funkcję realizuje maszyna J: plus czy kwus? Inaczej mówiąc, jaka jest kompetencja Jana?

Działanie każdej maszyny możemy opisywać za pomocą dwóch rodzajów zdań. Po pierwsze, są to zdania jednostkowe typu:

(1) Taka-a-taka maszyna w chwili t_n robi to-a-to,

gdzie za „to-a-to" wstawiamy opis jakiegoś konkretnego, dostępnego empirycznie faktu. Po drugie, są to zdania ogólne:

(2) Taka-a-taka maszyna realizuje funkcję F.

W przypadku opisanej wyżej maszyny J, zdania typu pierwszego to np.:

(1.1) Gdy na wejściu maszyny J w chwili t_1 wprowadziliśmy ciąg symboli: „2", „+", „4", „=", na wyjściu otrzymaliśmy „6";
(1.2) Gdy na wejściu maszyny J w chwili t_2 wprowadziliśmy ciąg symboli: „17", „+", „14", „=", na wyjściu otrzymaliśmy „31";
(1.3) Gdy na wejściu maszyny J w chwili t_3 wprowadziliśmy ciąg symboli: „35", „+", „29", „=", na wyjściu otrzymaliśmy „64".

Natomiast zdania typu drugiego są następujące:

(2.1) Maszyna J dodaje;
(2.2) Maszyna J kwoduje.

Wyjaśnienia w kognitywistyce mają charakter zdań typu (2). Analizując więc wartość tych wyjaśnień, należy przede wszystkim rozstrzygnąć, jakie fakty są warunkami prawdziwości dla tego typu zdań.

Ogólnie rzecz biorąc, zdania typu (2) możemy rozumieć dwojako: realistycznie bądź instrumentalnie. W pierwszym przypadku warunkami prawdziwości tych zdań są fakty, które znajdują się „wewnątrz" maszyny. Takim faktem może być fizyczna budowa bądź realizowany program. W drugim przypadku zdania typu 2 mówią o interpretacji działań maszyny przez jej twórcę lub użytkownika. Dotyczą więc faktów, które są „na zewnątrz" maszyny.

Kognitywiści przyjmują pierwszą z tych możliwości. Utrzymują, że mówiąc: „Maszyna J dodaje", opisujemy jakiś fakt, który możemy wykryć przyglądając się maszynie „samej w sobie". Inaczej mówiąc, nie musimy zaglądać do umysłu twórcy maszyny, by rozstrzygnąć, czy realizowana przez nią funkcja jest dodawaniem, czy kwodawaniem.

3. Nagie fakty, czyli analiza dyspozycjonalna

Termin „maszyna" jest wieloznaczny. Możemy przez niego rozumieć, po pierwsze, pewien konkretny przedmiot fizyczny złożony ze skończonej ilości części (sprzęt), po drugie, byt abstrakcyjny (program), i wreszcie po trzecie, połączenie jednego i drugiego.

Jeżeli przyjmiemy pierwszą z tych możliwości, zdania typu (2) należy interpretować jako opisy dyspozycjonalnych własności pewnych konkretnych fizycznych struktur. Zgodnie z tym ujęciem, mówiąc, że maszyna J dodaje, mamy na myśli:

(3.1) Gdybyśmy wprowadzili do maszyny J ciąg symboli: „68", „+", „57", na wyjściu otrzymalibyśmy „125".

Podobnie, mówiąc, że maszyna J kwoduje, rozumiemy przez to:

(3.2) Gdybyśmy wprowadzili do maszyny J ciąg symboli: „68", „+", „57", na wyjściu otrzymalibyśmy „5".

Oba zdania opisują pewną możliwość (potencję) tkwiącą w maszynie. Chociaż dotychczas maszyna J nigdy nie otrzymała na wejściu liczb większych niż 56, to ma dyspozycję do określonego działania, gdyby ktoś takie liczby wprowadził. Dyspozycja ta jest redukowalna do obecnego stanu fizycznego maszyny. Zdania (3.1) i (3.2) należy więc rozumieć jako opis jej budowy. Ich warunkiem prawdziwości jest fakt, że maszyna jest taką, a nie inną fizyczną strukturą. Przemawia za tym następująca intuicja. Zarówno to, co maszyna robiła przed chwilą t_n, czyli gdy wpisywaliśmy np. na klawiaturze ciąg znaków: „4", „+", „5", „=" i otrzymywaliśmy na wyświetlaczu „9", jak i to, co będzie robiła po tej chwili, czyli w momencie, gdy wciśniemy następującą sekwencję klawiszy: „6", „8", „+", „5", „7", „=", nie jest przypadkowe. Rezultat otrzymywany na wyjściu w każdym wypadku (przeszłym i przyszłym) jest całkowicie zdeterminowany przez procesy fizyczne zachodzące wewnątrz (i ewentualnie na zewnątrz) maszyny. Krótko mówiąc, tak jak z fizycznej budowy szyby możemy wyprowadzić wniosek, że się potłucze, gdy z odpowiednią siłą rzucimy w nią kamieniem, tak z aktualnej fizycznej budowy maszyny możemy wyczytać, co maszyna zrobi, gdy na wejściu otrzyma określone dane.

Zwolennik analizy dyspozycjonalnej staje jednak przed szeregiem trudności. Po pierwsze, na gruncie tego ujęcia nie da się wyjaśnić tzw. problemu błędów. Przyjmijmy, że rzeczywiście znając aktualną fizyczną budowę maszyny możemy wyprowadzić wniosek, jak zachowa się ona w przyszłości, gdy na wejściu zostaną wprowadzone określone liczby[1]. Dokładnie badamy maszynę J i wychodzi nam, że gdy na wejściu wpiszemy „68 + 57", na wyjściu otrzymamy „5". Bierze się to z pewnych cech jej fizycznej struktury. Czy w ten sposób sfalsyfikowaliśmy zdanie, że maszyna dodaje, i jednocześnie uzasadniliśmy, że kwoduje? Oczywiście, że nie. Odkryte przez nas we wnętrzu maszyny nagie fizyczne fakty w żaden sposób nie rozstrzygają, jaką funkcję maszyna realizuje, bo są one zgodne zarówno z twierdzeniem, że maszyna dodaje, tylko ma jakiś defekt, który po przekroczeniu pewnego pułapu liczb sprawi, że będzie podawała błędne wyniki, jak też z twierdzeniem, że jej fizyczna struktura poprawnie realizuje funkcję kwodawania. Nagie fizyczne fakty są warunkami prawdziwości zdań (3.1) i (3.2), ale nic nam nie mówią o wartości logicznej zdań (2.1) i (2.2). Dzieje się tak dlatego, że w zadaniach typu 2 zawarty jest element oceny tego, co robi maszyna, i tym samym nie da się ich zredukować do skończonej ilości jednostkowych faktów. Aby rozstrzygnąć, czy faktyczny sposób działania maszyny jest poprawną realizacją funkcji F, czy błędną realizacją

[1] Tak jak mechanik, badając stan techniczny samochodu, może stwierdzić, że gdy przekroczymy pewną prędkość, w samochodzie odpadnie koło.

funkcji *G*, musimy spojrzeć na nią z zewnętrznej perspektywy. Potrzebujemy „instancji zewnętrznej", którą w tym wypadku są intencje twórcy lub użytkownika maszyny. To one decydują, czy dane działanie jest poprawne, czy może mamy do czynienia z usterką.

Po drugie, każde urządzenie, więc również maszyna J, o której tu mowa, jest skończone. Na przykład kalkulator może wykonywać obliczenia tylko na liczbach o określonej wielkości. Przyjmijmy, że wartości funkcji kwodawania i dodawania zaczynają się różnić dopiero w przypadku liczb, których, ze względu na fizyczne ograniczenia, takie jak wielkość pamięci czy moc obliczeniowa, nie jesteśmy w stanie wprowadzić do naszej maszyny. W takim przypadku, wszystkie działania maszyny oraz jej całkowity fizyczny opis będą do pogodzenia zarówno ze zdaniem, że maszyna dodaje, jak i ze zdaniem, że kwoduje.

Po trzecie, własności dyspozycjonalne maszyny, o których mówią zdania (3.1) i (3.2), mogą wynikać z bardzo wielu różnych własności fizycznych. Nie da się więc w prosty sposób zredukować dyspozycji maszyny do jej elementarnej fizycznej struktury. Każda taka redukcja jest wyborem dokonanym przez podmiot interpretujący działania maszyny.

Po czwarte, dyspozycjonalne własności maszyny, które stanowią warunki prawdziwości dla zdań typu (3), nie zależą jedynie od nagich fizycznych faktów, które znajdujemy „wewnątrz" maszyny, ale również od zewnętrznych warunków, w których maszyna działa. W tekście „Wiele twarzy realizmu" Hilary Putnam podzielił dyspozycje na ścisłe i warunkowe. Te pierwsze przysługują przedmiotom bez względu na okoliczności. Na przykład ciało o niezerowej masie spoczynkowej ma ścisłą dyspozycję do poruszania się wolniej niż światło. Jest fizycznie niemożliwe, by było inaczej. Jednak zdecydowana większość dyspozycji ma charakter warunkowy. Cukier rozpuszcza się w wodzie, tylko gdy zostaną spełnione pewne zewnętrze warunki.

> Faktem jest, że możemy powiedzieć tylko tyle, iż w *normalnych* warunkach cukier umieszczony w wodzie rozpuści się. Nie ma żadnych podstaw, by sądzić, iż wszystkiego rodzaju nienormalne warunki (w tym dziwaczne stany kwantowe, lokalne fluktuacje czasoprzestrzenne itd.), przy których cukier umieszczony w wodzie nie rozpuściłby się, dadzą się zbiorczo ująć za pomocą zamkniętej formuły języka fundamentalnej fizyki. (Putnam 1998: 334–335)

Problem ten dotyczy też zdań opisujących dyspozycje maszyny J. Ponieważ własności dyspozycjonalne zależą nie tylko od stanów fizycznych maszyny, ale również od zewnętrznych okoliczności, nie da się zredukować zdań typu (3) do skończonej liczby zdań będących formułami języka fizyki.

Wydaje się, że wskazane wyżej trudności można przezwyciężyć, wprowadzając do analizy dyspozycjonalnej warunek *ceteris paribus*. W tym wypadku, mówiąc, że maszyna dodaje, mamy na myśli:

(4.1) Gdybyśmy wprowadzili do wyidealizowanej maszyny ciąg symboli: „68", „+", „57", na wyjściu otrzymalibyśmy „125",

a mówiąc, że kwoduje:

(4.2) Gdybyśmy wprowadzili do wyidealizowanej maszyny ciąg symboli: „68", „+", „57", na wyjściu otrzymalibyśmy „5".

W obu wypadkach przez wyidealizowaną maszynę rozumiemy urządzenie, które nie może się popsuć, może wykonywać operacje na dowolnie dużych liczbach, na jego działanie nie mają wpływu czynniki zewnętrzne itd.

Ze względu na cele, które przyświecały zwolennikowi analizy dyspozycjonalnej, jest to jednak rozwiązanie desperackie. Przypomnijmy, że chciał on rozstrzygnąć, jaką funkcję realizuje określona maszyna. Warunków prawdziwości zdań typu (2) szukał we własnościach dyspozycjonalnych maszyny. Te ostatnie chciał natomiast zredukować do jej fizycznej budowy. Wprowadzając do swoich analiz warunek *ceteris paribus*, rezygnuje z realizacji tego celu. Nie możemy ustalić, co robi konkretna maszyna J, odwołując się do własności dyspozycjonalnych jakiejś różnej od J wyidealizowanej maszyny.

Co więcej, samo pojęcie wyidealizowanej maszyny zakłada wyjście poza jej strukturę fizyczną i odwołanie do intencji twórcy. Wyidealizowanie nie jest bowiem czymś zastanym w świecie; jest zawsze czyimś dziełem. Twierdzenie, że maszyna nie może się popsuć, wymaga wcześniejszego ustalenia kryterium poprawności jej działania i tym samym zidentyfikowania faktów, które będą traktowane jako usterki. Teoretycznie rzecz biorąc, taka charakterystyka może mieć dwie formy. Możemy podać nieskończoną tabelę stanów maszyny bądź wskazać regułę definiującą poprawność. Pierwszy przypadek jest praktycznie niewykonalny, drugi – powoduje, że zaczynamy kręcić się w koło. Analiza dyspozycjonalna miała określić, jaką funkcję realizuje maszyna. Okazuje się jednak, że by ją przeprowadzić, musimy już wiedzieć, jaka to jest funkcja.

Widzimy, że żadna forma analizy dyspozycjonalnej nie pozwala odpowiedzieć na pytanie, jaką funkcję realizuje konkretna fizyczna maszyna, która na pytanie „2 + 2" odpowiada „4", na pytanie „24 + 21" – „45", na pytanie „67 + 58" – „125".

> (...) konkretna fizyczna maszyna, rozpatrywana jako obiekt bez jakiegokolwiek odniesienia do projektanta, może (w przybliżeniu) podpadać pod dowolną liczbę programów rozszerzających (w przybliżeniu, pozwalając na pewne „usterki") jej skończone możliwości działania. Jeśli owa fizyczna maszyna nie została zaprojektowana, ale, by tak rzec, „spadła z nieba", nie może istnieć żaden stan rzeczy określający, pod który program maszyna „naprawdę" podpada, ani, co za tym idzie, żadna „najprostsza hipoteza" dotycząca tego nieistniejącego faktu. (Kripke 2007: 68–69)

4. Nad-fakty

Wydaje się, że kłopoty związane z analizą dyspozycjonalną nie dotyczą większości kognitywistów. Przez maszynę zwykle rozumieją oni nie pewien określony fizyczny przedmiot, tylko program. Procesy poznawcze opisują w kategoriach stanów obliczeniowych, a nie stanów fizycznych.

> Gdybyśmy chcieli streścić program badawczy kognitywizmu, powiedzielibyśmy: Myślenie jest procesem przetwarzania informacji, zaś przetwarzanie informacji jest niczym innym jak manipulacją symbolami. Czynią to komputery, dlatego najlepszym sposobem badania myślenia (używa się tu raczej słowa „poznanie") jest badanie obliczeniowych programów manipulowania symbolami, niezależnie od tego, czy zachodzą one w komputerach, czy w mózgach. Dlatego, zgodnie z tym poglądem, zadaniem dla nauk o poznawaniu jest opisywanie mózgu nie na poziomie komórek nerwowych, ani nie na poziomie świadomych stanów psychicznych, lecz na poziomie, na którym działa on jak system przetwarzania informacji. (Searle 1995: 39)

Przy tym ujęciu warunkiem prawdziwości zdań typu (2) będzie fakt, że maszyna realizuje taki, a nie inny program. Powiemy:

(5.1) Zdanie „Maszyna J dodaje" jest prawdziwe wtw, gdy maszyna J realizuje program dodawania;
(5.2) Zdanie „Maszyna J kwoduje" jest prawdziwe wtw, gdy maszyna J realizuje program kwodawania.

By jednak odpowiedź tę uznać za zadowalającą, musimy przede wszystkim rozstrzygnąć, na czym polega fakt, że maszyna realizuje określony program. O czym mówi np. wyrażenie: „Maszyna J realizuje program dodawania"? Czy jest to jedynie właściwa dla kognitywistyki forma opisu surowych fizycznych faktów zachodzących w maszynie, czy też odnosimy się w ten sposób do pewnego nieredukowalnego do sprzętu, abstrakcyjnego obiektu sterującego pracą komputera: swoistego nad-faktu? Pierwsze, niesubstancjalne rozumienie programu w gruncie rzeczy cofa nas do analizy dyspozycjonalnej. Wprawdzie opisujemy maszynę w kategoriach obliczeniowych, a nie fizycznych, ale kategoriom tym po stronie świata w sensie dosłownym nic nie odpowiada. Podobnie, opisując stół w kategoriach geometrycznych, nie musimy zakładać, że w świecie poza strukturami fizycznymi istnieją prostokąty. Przy tej interpretacji, ostatecznych warunków prawdziwości dla zdań typu (2) należałoby szukać wśród faktów fizycznych. W części 3 starałem się pokazać, że nie jest to stanowisko wiarygodne.

Przyjrzyjmy się więc drugiej możliwości. Zgodnie z nią, program ma charakter substancjalny.

> Właśnie to bowiem mamy ochotę powiedzieć – wyobrażamy tu sobie maszynę matematyczną, która, napędzana przez same reguły, posłuszna jest tylko prawom matematycznym, nie zaś fizycznym. (Wittgenstein 2000: 206)

Jest on nieredukowalnym do fizycznych stanów maszyny nad-faktem.

Wydaje się, że ujęcie to jest dużo bardziej płodne niż analiza dyspozycjonalna. Unika się tu nieprzezwyciężalnych w przypadku fizycznego rozumienia maszyny problemów błędu i skończoności. Program rozumiany jako byt abstrakcyjny (maszyna Turinga) nie może się zepsuć i może wykonywać operacje na dowolnie dużych argumentach. Jednak nawet gdy przyjmiemy, że maszyną sterują substancjalnie rozumiane reguły, że program jest abstrakcyjnym bytem, który „napędza" działanie tranzystorów bądź neuronów, nie znajdziemy w ten sposób warunków prawdziwości zdań typu (2). Skoro wszystko, co robi konkretna fizyczna maszyna J, da się pogodzić z wieloma abstrakcyjnymi programami, skąd wiemy, który z nich w danej chwili „napędza" maszynę? Problem, czy maszyna dodaje, czy kwoduje, pozostaje wciąż nierozstrzygnięty.

5. Interpretacje

Wyżej starałem się pokazać, że ani wśród surowych faktów fizycznych (sprzęt), ani wśród postulowanych bytów abstrakcyjnych (program) nie znajdziemy warunków prawdziwości dla zdań mówiących, że konkretna maszyna realizuje konkretną funkcję. Nie zagraża to jednak w żaden sposób sensowności tego typu zdań. Możemy powiedzieć np. o kalkulatorze, że dodaje, pomimo że wewnątrz kalkulatora nie znajdziemy niczego, co rozstrzygnęłoby, jaka jest wartość logiczna tego zdania. Dzieje się tak dlatego, że zdania typu (2) nie mają charakteru realistycznego, tylko instrumentalny. Są one zawsze jedną z wielu możliwych (na gruncie konkretnego zbioru faktów) interpretacji tego, co robi dane urządzenie. W interpretacji tej zawiera się nieusuwalny element normatywny. Rozstrzyga on, kiedy maszyna działa poprawnie, a kiedy popełnia błędy. Nawet istota wszechwiedząca nie będzie w stanie ustalić, czy maszyna, która „spadła z nieba", dodaje, kwoduje, czy realizuje którąś z nieskończenie wielu innych funkcji. Nie będzie w stanie tego rozstrzygnąć, bo nie jest to coś, co znajdujemy w świecie, tylko coś, co do świata wnosimy.

Konkluzja ta podważa fundament, na którym wznosi się gmach kognitywistyki. Zwolennicy tego programu badawczego zwykle przyjmują obliczeniowe teorie umysłu. Podmiot rozumieją jako przetwarzającą informacje maszynę. Stosuje się więc do niego wszystko to, co wyżej powiedziałem o maszynie J. Generuje to regres.

Ustaliliśmy, że aby rozstrzygnąć, jaką funkcję realizuje maszyna J, nie wystarczy zajrzeć do jej wnętrza. To, czy maszyna dodaje, czy kwoduje, jest kwestią interpretacji. Interpretacja to stan umysłu twórcy lub użytkownika maszyny, czyli – według kognitywisty – stan obliczeniowy pewnej innej maszyny, nazwijmy ją K. By jednak rozstrzygnąć, jaki to jest stan, czyli co robi („ma na myśli”) maszyna K przypisując maszynie J realizację określonej funkcji, niezbędna jest kolejna interpretacja. I tak w nieskończoność.

6. Wnioski

W kognitywistyce wciąż dominuje metafora komputerowa. Poznający podmiot jest przedstawiany jako przetwarzająca informacje maszyna. Jego stany charakteryzowane są w kategoriach obliczeniowych. Jeżeli jednak zgodzimy się, że zdania mówiące, iż dana maszyna realizuje taką-a-taką funkcję, nie są opisem faktów, tylko ich interpretacją, ujęcie to prowadzi do regresu. Tego typu regres nie jest problemem w przypadku inżyniera. Konstruując maszynę symulującą sposób, w jaki Jan używa symbolu „+”, nie musi on rozstrzygać, którą z potencjalnie nieskończonej ilości funkcji ten symbol reprezentuje. Kognitywista zwykle ma jednak większe ambicje. Stara się on zająć miejsce tradycyjnie rozumianego epistemologa. Przy tego typu aspiracjach wskazany wyżej regres staje się poważnym problemem. Powoduje, że obliczeniowa teoria umysłu, choć może być użyteczna praktycznie, eksplanacyjnie jest jałowa.

Bibliografia

Kripke S. (2007), *Wittgenstein o regułach i języku prywatnym*, przeł. K. Posłajko i L. Wroński, Warszawa: Fundacja Aletheia.

Putnam H. (1998), „Wiele twarzy realizmu", w: tenże, *Wiele twarzy realizmu i inne eseje*, przeł. A. Grobler, Warszawa: Wydawnictwo Naukowe PWN, s. 323–429.

Quine W.V.O. (1986), „Epistemologia znaturalizowana", w: tenże, *Granice wiedzy i inne eseje filozoficzne*, przeł. B. Stanosz, Warszawa: Państwowy Instytut Wydawniczy, s. 53–70.

Searle J. (1995), *Umysł, mózg i nauka*, przeł. J. Bobryk, Warszawa: Wydawnictwo Naukowe PWN.

Wittgenstein L. (1972), *Dociekania filozoficzne*, przeł. B. Wolniewicz, Warszawa: PWN.

Wittgenstein L. (2000), *Uwagi o podstawach matematyki*, przeł. M. Poręba, Warszawa: Wydawnictwo KR.

Streszczenie

Celem tekstu jest wykazanie, że preferowane w obrębie kognitywistyki obliczeniowe teorie umysłu generują regres w nieskończoność i tym samym nie mają mocy eksplanacyjnej. Uzasadnienie tej tezy opiera się na przedstawionej przez Saula Kripkego w książce *Wittgenstein o regułach i języku prywatnym* nonfaktualnej interpretacji filozofii późnego Wittgensteina.