

### *Umysł – software czy hardware?*

Jednym z powodów tego, że problemy filozoficzne zyskują lub tracą na aktualności, jest nasza wiara w szansę ich ostatecznego rozstrzygnięcia. Czasami - zazwyczaj wtedy, gdy filozofowie otrzymają do swej dyspozycji jakieś nowe narzędzie – dominować zaczyna przekonanie, że rozwiązanie któregoś z tradycyjnych problemów jest już tylko kwestią czasu. Narzędzia, które w ten optymistyczny sposób nastrajają filozofów, mogą być tak złożone jak całkiem nowe dziedziny wiedzy lub tak proste jak pojedyncza analogia. Jedną z takich popularnych analogii chciałbym się w tym artykule zająć – jest nią analogia pomiędzy dychotomią umysł/ciało a dychotomią *software/hardware* (w dalszych częściach tekstu nazywaną dychotomią S/H). Moim celem jest ocena trafności tej analogii, co pozwoli nam sformułować odpowiedź na tytułowe pytanie.

Na początek chciałbym zająć się dwoma zagadnieniami terminologicznymi. Jest to o tyle istotne, że są to ustalenia, nie pozbawione pewnych implikacji filozoficznych. Po pierwsze – niektórzy czytelnicy mogą mieć wrażenie, że zestawienie dychotomii S/H z dychotomią umysł/ciało nie jest najszcześniejsze. Być może powinniśmy mówić raczej o analogii pomiędzy dychotomią S/H a dychotomią umysł/mózg? Sądzę, że są przynajmniej dwa powody, dla których to pierwsze z tych zestawień jest lepsze. Po pierwsze, *ciało* wydaje się być mniej więcej na tym samym stopniu ogólności co *hardware*, w odróżnieniu od *mózgu*, który pasowałby raczej na korelat *jednostki centralnej* lub *procesora*. Po drugie, *ciało* jest wobec *mózgu* kategorią nadrzędną i moglibyśmy przyjąć, że posługiwanie się taką ogólniejszą kategorią, stanowi po prostu wyraz pewnej ostrożności, do której nie należy przywiązywać większej wagi. Warto jednak zastanowić się przez chwilę, czym spowodowana miałaby być ta ostrożność. Powodem jest to, że nawet jeśli nie mamy wątpliwości co do tego, że umysł posiada pewien fizyczny nośnik, to nie mamy żadnej jasnej odpowiedzi na pytanie, co tym nośnikiem jest. W grę wchodzi tu przynajmniej trzy równoprawne odpowiedzi – „mózg”, „układ nerwowy” i „ciało”. Trudność, na którą się tu natknęliśmy polega na tym, że chcąc rozważyć dwa aspekty pewnego obiektu (programowy i sprzętowy lub umysłowy i cielesny, zależnie od tego czym się akurat zajmujemy), nie zakreśliliśmy granic pomiędzy tym obiektem a jego otoczeniem. Czy nośnik mojego umysłu zawiera moje ręce i nogi jako swoje części czy też może jest tylko do nich podłączony? A jeśli przyjmę drugą opcję – czy cały mózg uznać mam za nośnik mojego umysłu? Może powinienem zdecydować się jedynie na jakiś jego ośrodek a

pozostałe ośrodki nazwać otoczeniem? Czy porównanie z dychotomią S/H pomoże nam w udzieleniu odpowiedzi na te pytania? Nie, ale pomoże nam ono w pokazaniu, dlaczego popularna strategia „na przeczekanie” nic nie da.

Strategia ta polega na odłożeniu wszystkich tych pytań do czasu, gdy poznamy dokładnie budowę mózgu i sposób, w jaki przetwarza on informacje pochodzące z podłączonych do niego organów. Okazuje się, że mając podobną wiedzę o budowie komputera, jestem mniej więcej tak samo bezradny wobec analogicznych pytań dotyczących jego granic. Które z elementów całościowego systemu, dzięki któremu piszę te słowa a Ty Czytelniku je czytasz, nazwać powinienem komputerem, a które podłączonymi do niego urządzeniami peryferyjnymi? Są takie, co do których wątpliwości raczej nie mam – drukarki, monitora<sup>1</sup> czy klawiatury nie nazwałbym częścią komputera. Nie mam też wątpliwości, że procesor i płyta główna to części komputera. Ale do której z grup miałbym włączyć kartę graficzną czy dysk twardy? Wymienny dysk USB wydaje mi się typowym urządzeniem peryferyjnym ale co by było, gdybym włożył go do środka obudowy i uruchamiał z niego system operacyjny? A jeśli moje intuicje zmieniłyby się w tej kwestii, to czy powinienem, aby pozostać konsekwentnym, nazwać częścią komputera płytę CD, z której uruchomiłem system operacyjny? Być może niektórzy czytelnicy uznają, że osiągam w tych uwagach poziom szczegółowości, który nie przystoi filozofowi, zależy mi jednak na pokazaniu, że znajomość budowy urządzenia nie pomaga mi w udzieleniu odpowiedzi na pytania, które są zwierciadlanym odbiciem pytań o granice nośników umysłu. Co więcej – jest wręcz przeciwnie, czym więcej wiem o budowie tych urządzeń tym mniej jasne stają się pewne podziały – w tym przypadku na komputer i jego otoczenie a jak zobaczymy dalej, również na *hardware* i *software*.

Przyjęcie dualizmu psychofizycznego owocuje zdublowaniem każdego problemu, nie powinniśmy się zatem dziwić, że obok niepewności co do granic nośnika umysłu, doskwiera nam też niepewność co do granic samego umysłu. Niepewność ta wiąże się z drugą z zapowiadanych kwestii terminologicznych, omówimy więc obie te sprawy za jednym zamachem. Od początku tego artykułu posługuję się anglojęzycznymi określeniami *software* i *hardware*, co może niektórych razić – czy nie mógłbym równie dobrze posługiwać się polskimi odpowiednikami – *oprogramowanie* i *sprzęt*? Powodem, dla którego zdecydowałem się na makaronizmy, jest dwuznaczność słowa *software*, którą nie cechuje się polski odpowiednik – *oprogramowanie*. Istnieje węższe i szersze znaczenie słowa *software* – w znaczeniu szerszym, terminem tym oznacza się nie tylko oprogramowanie ale i wszelkie dane, które komputer, za pomocą owego oprogramowania przetwarza. W węższym znaczeniu, *software* to na przykład edytor tekstu, którym się teraz

<sup>1</sup> Podejrzewam, że intuicje użytkowników notebooków i niektórych komputerów firmy Apple nie są w tym przypadku tak jednoznaczne, jak moje. Rolę odgrywa tu niemożność fizycznego odłączenia monitora od tego, co ja nazwałbym komputerem.

posługuję. W znaczeniu szerszym, *software* to również pisany przeze mnie artykuł. Polski termin *oprogramowanie* odpowiada jedynie znaczeniu węższemu. Dwuznaczność ta ma jednak pewne znaczenie filozoficzne. Zajmujący się umysłem filozofowie, już od czasów Locke'a, radzić sobie musieli z jej odpowiednikiem. Czy umysłem nazwać należy jedynie zbiór operacji na treściach psychicznych czy też również same te treści? Nigdzie dwuznaczność ta nie daje o sobie znać w większym stopniu niż w napięciu, jakie powstaje pomiędzy hume'owskim obrazem umysłu jako ciągu idei a wyróżnionym statusem operacji umysłu, o których autor ten niezwykle często rozprawia. Czy operacje są po prostu jakimś specjalnym rodzajem idei czy może raczej prawami, zgodnie z którymi idee następują po sobie w ciągu. Do niektórych ze wzmiankowanych tu problemów jeszcze powrócimy.

Jak dotąd wygląda na to, że mamy do czynienia z udaną analogią – podobieństwa między dychotomiami zaczynają się mnożyć. Warto jednak uzmysłwić sobie główny powód, dla którego analogia pomiędzy dychotomią S/H a dychotomią umysł/ciało wydaje się wielu ludziom pouczająca. Najważniejszą intuicją stojącą za podziałem na *software* i *hardware* jest rozróżnienie na fizyczne i niefizyczne aspekty jakiegoś urządzenia. To oczywiście nie wszystko – istotne jest tu założenie, że pomiędzy owymi niefizycznymi i fizycznymi aspektami, zachodzą związki przyczynowe. Nasza analogia wydaje się płodna, ponieważ tradycyjne pytania dotyczące stosunku umysłu do ciała mają swój odpowiednik w pytaniach o stosunek *software* do *hardware*. Zamiast pytać o to, w jaki sposób pewien rodzaj myśli, którym jest moja chęć poruszenia ręką, wywołuje owo poruszenie, mogę pytać, w jaki sposób operacje logiczne mogą wywołać ruch membrany w głośniku, z którego płynie zapisana cyfrowo muzyka albo zbombardować ekran monitora elektronami. Te ostatnie pytania wydają się mieć nad pytaniem tradycyjnym olbrzymią przewagę – znamy budowę komputerów i to my je oprogramowaliśmy – powinniśmy więc znać poprawne odpowiedzi. Najprostszym sposobem na rozstrzygnięcie tych problemów jest próba odrzucenia dualizmu, poprzez sprowadzenie *software* do *hardware*. Tęsknota za taką redukcją nie jest oczywiście niczym w filozofii nowym – wielu naturalistycznie nastawionych filozofów chciałoby zredukować umysły do ciał. Do jakiego fizycznego aspektu świata moglibyśmy spróbować zredukować kategorię *software*? Naturalnym kandydatem wydaje się być struktura geometryczna nośników – jest to jakaś pociecha, choć struktury są abstraktami, przez co i tak wydają się przebywać w świecie naturalistów jedynie na specjalnej przepustce, udzielonej im przez akceptujące własności geometryczne nauki ścisłe. Oczywiście nie mówimy tu o uznaniu, że każda struktura geometryczna przedmiotu to jakiś rodzaj *software*. Z całą pewnością, nie każdą strukturę bylibyśmy skłonni tak nazwać. Kształt klucza nie jest oprogramowaniem uruchamianym w zamku. Nie jest nim, też kształt zamka. Zapisany na karcie magnetycznej kod otwierający zamek

elektroniczny to już jednak, jak się wydaje *software*. A czy rząd przycisków w windzie nazwalibyśmy jej oprogramowaniem? Jakie warunki musi spełniać dana struktura aby uznać ją za *software*?

Zanim odpowiemy sobie na to pytanie proponuję rozważenie dwóch pytań pomocniczych, które pozwolą nam lepiej pojąć relacje pomiędzy *software* a strukturami ich nośników. Po pierwsze – zastanówmy się, czy ten sam *software* zapisany być może na nośnikach o zupełnie różnej strukturze. Podejrzewam, że odpowiedź na to pytanie nie nastęczy trudności nikomu. Doskonale wiemy, że to samo oprogramowanie, czy też te same dane, możemy zapisać na nośnikach zupełnie do siebie niepodobnych. Człowiek, który nie wiedziałby nic o komputerach, miałby spore trudności z zaklasyfikowaniem przedmiotów tak różnych jak kasetta magnetofonowa, płyta CD, dysk twardy czy przenośny dysk USB do tej samej kategorii. Próżno byłoby tu szukać jakiegoś zbioru wspólnych własności – materiałem na nośnik może być bowiem dowolny przedmiot. W tym właśnie leży urok cyfrowego zapisu, że w odróżnieniu od zapisu analogowego, dane można przenosić z nośnika na nośnik mając pewność, że wykonaliśmy kopię doskonałą. Skąd czerpiemy tę pewność? Wykonując kopię zapisu analogowego staramy się odtworzyć własności oryginalnego nośnika w nośniku-kopii, co często jest niewykonalne albo niezwykle pracochłonne. W przypadku zapisu cyfrowego, odtwarzamy jedynie ciąg zer i jedynek, które reprezentowane mogą być w nośniku przez dowolne dwie własności (byle odróżnialne od siebie).

Zostało nam jednak jeszcze drugie z zapowiadanych pytań pomocniczych; oto ono – czy dwa nośniki o dokładnie tej samej strukturze mogą być nośnikami różnego *software*? Podobnie jak w przypadku pytania poprzedniego, odpowiedź nasuwa się natychmiast, z tą różnicą, że tym razem jest to odpowiedź negatywna. Obawiam się jednak, że wbrew pozorom, nie jest to wcale aż tak oczywiste. Czy nie mogłoby bowiem być tak, że dokładnie ten sam zestaw instrukcji, zapisany w ten sam sposób na takich samych nośnikach, byłby oprogramowaniem dwóch różnie zbudowanych komputerów, przez co wywoływałby zupełnie różne efekty? To zupełnie tak, jak z dwoma identycznymi pod każdym względem kartkami papieru, na których znajduje się identyczny pod względem geometrycznym ciąg symboli – czy nie mogłyby one być nośnikami dwóch zupełnie różnych komunikatów? Oczywiście mogłyby - wszystko zależy od tego, jakiego zestawu reguł użyjemy do ich zdekodowania. Zaczyna wyglądać więc na to, że próba zredukowania *software* do własności fizycznych napotyka na poważną trudność. Jeżeli analogia pomiędzy dychotomią S/H a dychotomią umysł/ciało jest poprawna, to akcje zwolenników naturalizacji umysłu właśnie zaczęły spadać. Zastanówmy się więc nad jej poprawnością.

Porównanie z komunikatem na kartce nie pojawiło się w poprzednim akapicie przypadkowo. Aby

unaocznic sobie pewną istotną własność *software*, proponuję zwrócić się w stronę historii elektroniki. W roku 1976 pojawiła się na rynku amerykańskim nowość – pierwsza programowalna konsola do gier wideo. Z punktu widzenia użytkownika nie różniła się ona zbyt od poprzedzających ją konsol, które programowalne nie były. W obu przypadkach użytkownik brał do ręki pewien przedmiot (kasetkę, będącą obudowaną w plastik płytką elektroniczną) i umieszczał go w konsoli celem uruchomienia konkretnej gry. Różnica pomiędzy omawianym urządzeniem a jego poprzednikami polegała na tym, że poprzednie konsole posiadały pewną skończoną ilość stanów, w których mogły się znaleźć a kasetka, którą wciskał w nie użytkownik służyła do wyboru jednego z nich. Polegało to na tym, że rzeczona kasetka zawierała po prostu taki a nie inny układ ścieżek elektronicznych, które zawierały wybrane ścieżki w gnieździe konsoli. Wybór gier dokonywał się w ten sposób, że jedna kasetka zawierała jedne ścieżki a druga inne. Powszechnie uznaje się, że nie ma tu jeszcze mowy o *software*. Nasze kasetki funkcjonowały po prostu jak dość nietypowe przyciski. Co ciekawe, konstrukcja kasetki konsoli programowalnej jest dość podobna. Również mamy tu do czynienia ze zwieraniem obwodów. Różnica polega jednak na tym, że urządzenie zamiast ustawić się w jeden z przewidzianych przez konstruktorów stanów, ustawia się w sekwencję stanów, której konstruktorzy nie przewidywali – możliwych sekwencji jest po prostu zbyt wiele.<sup>2</sup> Zadaniem konstruktorów było jedynie ustanowienie zasad, zgodnie z którymi twórcy *software* układać mogą poprawne, czyli wywołujące zamierzone efekty, ciągi instrukcji. Oczywiście, wraz z pojawieniem się instrukcji pojawić się musi coś, co instrukcje te interpretuje – procesor. Słowo „interpretacja” może być jednak w tym kontekście mylące - ta przejawiana przez procesor zdolność do interpretacji sprowadza się po prostu do szczegółów jego budowy. Budowa ta jest po prostu fizyczną realizacją zbioru reguł pewnego języka, w tym samym sensie jak prosta bramka logiczna negacji jest fizyczną realizacją tej operacji logicznej.

Zauważmy jednak, że nasz obraz niepostrzeżenie się skomplikował. Zaczęliśmy od przeciwstawienia *software* i *hardware*. Po drodze pojawiły się jeszcze: nośniki, ich struktura, procesory i struktura procesorów. W jaki sposób powinniśmy uporządkować tę wielość i jak ma się do niej dychozomia umysł/ciało? Cóż – procesor jest rodzajem *hardware* i możemy łatwo wkomponować go w naszą analogię – jest on po prostu odpowiednikiem mózgu, o czym wspomnieliśmy już na początku. Ale dalej sytuacja nieco się komplikuje - jeśli coś miałooby w tym rozszerzonym obrazie odpowiadać umysłowi, to byłoby to raczej struktura procesora a nie wykonywane instrukcje. Co możemy o tej strukturze powiedzieć? Czy możemy uznać, że jest to po

---

<sup>2</sup> Można nawet powiedzieć, że jest ich nieskończenie wiele – jedynym ograniczeniem jest tu wielkość nośnika i zdolność urządzenia do odczytywania oprogramowania z nośników o dowolnej pojemności. Ponieważ ten drugi warunek nie jest w większości urządzeń spełniony, ograniczam się do mówienia o bardzo wielu możliwych sekwencjach.

prostu struktura geometryczna? Cóż, nie jest wcale wykluczone, by dwa procesory różniące się strukturą uruchamiały dokładnie ten sam *software* w taki sam sposób. To, czego poszukujemy to raczej „struktura logiczna” niż geometryczna. Dla objaśnienia różnicy posłużmy się takim oto przykładem: weźmy dwa, niezbyt podobne do siebie przedmioty codziennego użytku – włącznik światła i drzwi wejściowe. Są to przedmioty, które nie posiadają podobnej a tym bardziej wspólnej struktury geometrycznej. Posiadają one jednak tę samą strukturę logiczną – oba te przedmioty możemy bowiem uznać, za fizyczną realizację operacji negacji. Użycie włącznika zmienia pewien stan wyjściowy na inny. Jeżeli stanem wyjściowym było zgaszone światło to efektem jest zapalenie światła i odwrotnie. Gdy dwukrotnie użyję włącznika to stan wyjściowy pozostanie bez zmian. Gdy użyję drzwi, to jeśli stanem wyjściowym było moje przebywanie na zewnątrz domu, znajdę się wewnątrz domu i odwrotnie. Gdy użyję drzwi dwukrotnie to wyląduje w tym samym miejscu, z którego wystartowałem. Jak widać, struktura logiczna dwóch inaczej zbudowanych przedmiotów może być ta sama. Możliwa tożsamość struktur logicznych rozmaicie zbudowanych układów wynika też z czegoś innego – skoro alternatywę mogę zdefiniować za pomocą negacji i koniunkcji to mogę równie dobrze zbudować dwa urządzenia, które działały będą jak bramka logiczna alternatywy ale różniły się będą pod względem układu swych części.

Czy mogłoby jednakże być tak, że dwa tak samo zbudowane urządzenia są nośnikami dwóch różnych struktur logicznych? Trudno sobie taką sytuację wyobrazić, jedyna możliwość, jaka przychodzi mi do głowy sprowadza się do takiej oto wątpliwości:

Załóżmy, że dostaliśmy do ręki urządzenie, którego nie znamy i przyglądamy się jego budowie. Widzimy fragment, który wydaje się być bramką logiczną koniunkcji. Czy możemy być jednakże pewni, że przed naszymi oczami nie znajduje się bramka logiczna alternatywy? Aby je od siebie odróżnić, musimy wiedzieć, jaki stan lub własność oglądanego przez nas urządzenia odpowiada prawdzie a jaki fałszowi. Odpowiedniość ta jest jednak przyjętą przez konstruktorów konwencją, której nie wyczytamy z samej konstrukcji *hardware*<sup>3</sup>. Wygląda więc na to, że podobnie jak to się okazało w przypadku *software*, struktury logicznej *hardware* nie uda nam się sprowadzić do fizycznych własności jej nośnika (w tym przypadku procesora).

A jednak, jest to przedwczesne złożenie broni. Sformułujmy problem raz jeszcze - nie znając konwencji przyporządkowującej prawdę i fałsz poszczególnym stanom, czy własnościom procesora, wiemy o tych stanach tylko tyle, że są od siebie różne. Ta minimalna wiedza, nie

---

<sup>3</sup> W praktyce nie mielibyśmy zapewne większych problemów z tymi ustaleniami, wystarczyłoby znaleźć choćby bramkę logiczną implikacji, która nie ma swojego „lustrzanego” odpowiednika w tradycyjnym zestawie spójników. Nie może to jednak oddalić filozoficznej wątpliwości, ponieważ nic nie stoi na przeszkodzie, by jakieś urządzenie miało wbudowaną bramkę opartą na spójniku, który zwracałby prawdę tylko i wyłącznie wtedy, gdy jego poprzednik jest prawdziwy a następnik fałszywy.

pozwala nam na odróżnienie niektórych z operacji, które uznajemy zazwyczaj za odmienne. Ich jasne rozróżnienie było dla nas ważne, ponieważ uważaliśmy strukturę logiczną procesora za zestaw reguł, dzięki którym interpretowany jest *software*. Ale struktura ta wydaje się być zależna od konwencji, która nie jest częścią fizycznego wyposażenia procesora. Wygląda więc na to, że nie mogę zdekodować *software* nie tylko zanim nie poznam budowy *hardware* ale i zanim nie poznam pewnej dodatkowej konwencji. Takie sformułowanie problemu kryje już w sobie załączek jego rozwiązania – skoro konwencja ta nie jest wpisana w *hardware* (a tym bardziej w *software*, który musi być za pomocą tego *hardware* zdekodowany) to znaczy, że nie wpływa ona w żaden sposób na działanie urządzenia, ponieważ, mówiąc obrazowo, ono jej nie zna. Aby uzmysłowić sobie, co odgrywa tu rzeczywistą rolę, wróćmy na chwilę do naszego przykładu, w którym oglądaliśmy nieznaną nam urządzenie. Załóżmy, że dostajemy następujące polecenie – zaprogramuj urządzenie tak, aby naciśnięcie dwóch przycisków wywołało efekt w postaci zapalenia się lampki. Rozważmy znów nasze dwie bramki – alternatywy i koniunkcji. Czy różnice pomiędzy nimi nadal wydają się nieistotne? Okazuje się, że nie. Aby wykonać zadanie, muszę znaleźć taką bramkę, która w wyniku wystąpienia dwóch stanów, które wytworzone są przez naciśnięcie przycisków wywoła taki stan, który wywołuje zapalenie się lampki. Dla przykładu – jeżeli naciśnięcie przycisków wywołuje zwarcie, którego efektem jest popłynięcie prądu a nasza bramka jest tak skonstruowana, że tylko w tym przypadku przepuszcza prąd do żarówki, wywołując jej zaświecenie, to nie mamy innego wyjścia, jak ją właśnie wybrać. Konkurencyjna bramka, która wydawała się nam jeszcze przed chwilą tak podobna, działa bowiem w ten sposób, że zapala lampkę również wtedy, gdy naciśnięty jest tylko jeden z przycisków. Nie ma tu odwołania się do żadnej konwencji – wybór zdeterminowany jest przez konstrukcję urządzenia.

Zauważmy jednak, że w poprzednim akapicie przeszliśmy niepostrzeżenie od mówienia o konstrukcji procesora do mówienia o konstrukcji całego systemu złożonego z urządzeń wejścia (klawisze) i wyjścia (lampka). Na początku pisałem o tym, że nie ma jasności co do tego, co uznać za właściwy nośnik *software*, tak samo jak nie ma jasności co do tego, co uznać za właściwy nośnik umysłu. Z tego powodu wstrzymałem się przed mówieniem o dychotomii umysł/mózg na rzecz dychotomii umysł/ciało. To rozszerzenie granic pozwala nam uznać umysł za odpowiednik struktury logicznej całego urządzenia, na którym uruchamiany jest *software*. Jednocześnie nadal możemy przyjmować, że struktura ta zależy od własności fizycznych owego urządzenia – dwa identycznie zbudowane urządzenia nie mogą być nośnikami różnych reguł interpretacji pewnego ciągu instrukcji, którym jest *software*.

Takie ujęcie pozwala nam na sformułowanie dość zaskakującej konkluzji – umysł to raczej

*hardware* niż *software*<sup>4</sup>. Nienaturalność tego rozstrzygnięcia bierze się stąd, że o regułach interpretacji jakiegoś języka myślimy jako o sformułowanych w jakimś języku. Prowadzi to do regresu w nieskończoność, na co zwracał uwagę już Wittgenstein ale nauczyliśmy się akceptować ten paradoks jako coś nieuchronnego. Okazuje się jednak, że pogodziliśmy się z tym przedwcześnie – analiza dychotomii *S/H* pokazuje nam, że zasady interpretacji pewnych wyrażeń językowych mogą być po prostu zasadami budowy urządzenia, które służy do interpretacji tych wyrażeń.

---

<sup>4</sup> Zwolennikom umysłów jako *software* pozostaje jednak coś na otarcie łez – jest to pewna struktura, a zatem abstrakt.