

ROLA UCZNIA W PROCESIE PRZEKAZYWANIA WIEDZY

Natalia Marcinowska, Ryszard Szubartowski, Marta Kwaśnik
Stowarzyszenie Talent, Gdynia
natalia.marcinowska@talent.edu.pl; www.talent.edu.pl

Abstract. In this paper we show the example of using the holistic method in programming education. We try to explain how this method works on the example of educational project National Academy of Programming, which was taken by Talent Association in 2015. In this approach the progress of educational process is correlated with the activity and involvement of the student. The most important results of using this method in programming education is not only growth of the knowledge and abilities in digital capacities, but also shaping the social attitudes, priced and desired on contemporary labor market.

1. Wprowadzenie

Rewolucja technologiczna, którą rozpoczęło pojawienie się pierwszych komputerów, na trwałe zmieniła sposób funkcjonowania życia społecznego w zakresie komunikowania oraz dostępu do informacji. Udogodnienia, jakie wiązały się z rozwojem nowych technologii sprawiły, że transformację musiało lub wciąż musi przechodzić wiele podmiotów życia społecznego, chcących sprostać nowym wymaganiom stawianym przez współczesność.

Umiejętność programowania, o której istnieniu nie można było mówić jeszcze na początku XX wieku, domaga się podjęcia rzeczowej dyskusji na temat skutecznych metod jej przekazywania. Warto zaznaczyć, że w obecnej erze komputerów i nowych technologii, nabywanie kompetencji cyfrowych przez najmłodsze pokolenia ma miejsce jeszcze przed rozpoczęciem formalnej edukacji. Jednak według wskaźnika dla gospodarki i społeczeństwa cyfrowego (DESI) opracowanego przez Komisję Europejską i opublikowanego w lutym 2016 roku wynika, że Polska znajduje się na czwartym miejscu od końca na 28 państw członkowskich pod względem liczby mieszkańców posiadających przynajmniej podstawowe umiejętności cyfrowe [8]. W obliczu tak mało satysfakcjonującego wyniku, konieczne wydaje się opracowanie skutecznej metody, dzięki której możliwe będzie dostosowanie umiejętności

jednostki do potrzeb i wymagań dyktowanych przez rynek pracy, który w społeczeństwie informacyjnym opiera się w głównej mierze na przetwarzaniu, analizie i selekcji informacji [1].

Kierunki rozwoju edukacji programistycznej oraz kształcenia informatycznego w świetle nowej podstawy programowej zależą od metodologicznych założeń, jakie przyjmą jej twórcy, przy uwzględnieniu nie tylko zapotrzebowania rynku i oczekiwań wielkich korporacji, ale przede wszystkim w zgodzie z psychicznym, emocjonalnym oraz moralnym rozwojem osoby.

W 2015 roku Stowarzyszenie Talent podjęło się realizacji ogólnopolskiego projektu edukacyjnego, którego celem był rozwój nauki programowania wśród dzieci i młodzieży za pośrednictwem metody angażującej w proces przyswajania wiedzy nie tylko nauczyciela, ale przede wszystkim samego ucznia. Niniejszy artykuł stanowi opis tej metody wraz z najważniejszymi założeniami psychopedagogicznymi, antropologicznymi i etycznymi, które leżą u jej podstaw.

2. Ogólnopolska Akademia Programowania

Ogólnopolska Akademia Programowania, realizowana na zlecenie Ministerstwa Administracji i Cyfryzacji w ramach zadania publicznego „Nowatorskie inicjatywy na rzecz rozwoju umiejętności programowania oraz budowania kapitału społecznego z wykorzystaniem technologii informacyjno-komunikacyjnych” stanowi przykład metody nauczania, opartej na rozwoju myślenia komputacyjnego [*computational thinking*], którego istotę stanowi wykorzystanie zasad algorytmiki i narzędzi informatycznych do twórczego i samodzielnego rozwiązywania realnych problemów, pojawiających się w różnych dziedzinach życia [6]. Idei myślenia komputacyjnego towarzyszy przeświadczenie, że „algorytmika jest czymś więcej niż działem informatyki. Tkwi ona w centrum wszystkich działów informatyki i z całą uczciwością można o niej powiedzieć, że jest ważna dla większości nauk matematyczno-przyrodniczych, ekonomii i techniki. Sama natura algorytmiki czyni ją jednak szczególnie odpowiednią do stosowania w tych dyscyplinach, które czerpią korzyści z posługiwania się komputerami.” [3]

Punktem wyjścia w opracowywaniu nowoczesnych sposobów przekazywania wiedzy jest koncepcja jednostki, rozumianej jako zintegrowana jedność psychofizyczna, aktywnie włączona w proces uczenia się. Zgodnie z takim podejściem, uczeń zgłębiający tajniki programowania nie jest już tylko biernym odbiorcą gotowych rozwiązań przekazywanych mu przez nauczyciela, lecz w procesie dydaktycznym występuje jako twórca własnych narzędzi, a tym samym w relacji nauczyciel-uczeń pełni rolę nadrzędną. Coraz częściej podkreśla się również znaczenie emocji ucznia w procesie przyswajania wiedzy, traktując je jako filtr decydujący

o tym, które informacje zostają przechowywane w pamięci, a które z niej eliminowane [2].

Wspomniane założenia zostały wzięte pod uwagę przez członków Stowarzyszenia Talent podczas tworzenia struktury organizacyjnej i podstawy dydaktycznej projektu. Ogólnopolska Akademia Programowania polegała na utworzeniu w całym kraju niemal dwustu stref, w których w okresie od września do listopada 2015 odbywały się zajęcia z programowania dla uczniów wszystkich poziomów edukacyjnych: od szkół podstawowych po edukację na poziomie akademickim. W ramach tego przedsięwzięcia odbyły się również trzy cykle warsztatów metodyczno-dydaktycznych dla nauczycieli informatyki oraz obozów naukowych dla młodych instruktorów programowania, przygotowujące do samodzielnego prowadzenia zajęć pozalekcyjnych.

Jednym z podstawowych celów planowania strategii projektu było dążenie do upowszechnienia nauki programowania wśród uczniów wszystkich poziomów edukacji, również tych nieposiadających w tej dziedzinie doświadczenia i umiejętności. Egalitaryzm metody objawiał się przede wszystkim w dostosowaniu trudności rozwiązywanych zadań do posiadanej wiedzy i predyspozycji danej grupy. Elementem nowatorskim w projekcie było wykorzystanie potencjału uczniów szczególnie uzdolnionych w zakresie umiejętności programowania poprzez włączenie ich w proces kształcenia programistów o mniejszych kompetencjach. W ten sposób rozwijał się nie tylko uczeń korzystający z zajęć, ale także sam instruktor, który poprzez opracowanie zadań i przygotowanie się do zajęć porządkował zdobytą wcześniej wiedzę i przyswajał ją w większym stopniu.

2.1. Młody instruktor w roli nauczyciela

Zamierzonym działaniem realizatorów projektu było przeprowadzenie trzech edycji warsztatów metodyczno-dydaktycznych dla nauczycieli oraz obozów instruktorskich dla uczniów szczególnie uzdolnionych w zakresie nauk ścisłych oraz informatyki w jednym czasie. Włączenie młodych instruktorów w harmonogram zajęć przewidzianych dla nauczycieli sprawiło, że dydaktycy na co dzień pracujący według klasycznego modelu przekazywania wiedzy, z jednej strony mieli możliwość obserwowania talentów najlepszych młodych programistów, natomiast z drugiej zdystansowania się do podawczego sposobu przekazywania wiedzy, która do tej pory dominowała w ich pracy zawodowej. Zgodnie z ideą „szkoły odwróconej” [4] [*flipped learning*] rolę nauczycieli przejmowali uczestnicy obozów instruktorskich, których wiedza i umiejętności przekraczały niejednokrotnie kompetencje dydaktyków zgłaszających się na warsztaty.

Pierwszym wnioskiem, który nasuwał się w trakcie przeprowadzania łączonych zajęć instruktorów z nauczycielami, była znaczna różnica między nimi w poziomie posiadanej wiedzy. Programowanie na zaawansowanym poziomie, zawierające

w sobie nie tylko umiejętność tzw. „kodowania” (a więc zapisywania danego schematu w określonym języku programowania), ale wymagające przede wszystkim znajomości algorytmiki do rozwiązywania złożonych problemów, stanowiło narzędzie, którym znacznie sprawniej posługiwali się instruktorzy niż nauczyciele. Nie wdając się w dyskusję na temat poziomu szkoleń informatycznych dla nauczycieli można stwierdzić, że różnica w stopniu zaawansowania wiedzy i umiejętności młodzieży i dorosłych otworzyła przestrzeń dla podjęcia między nimi współpracy.

Co istotne, chęć wykorzystania potencjału intelektualnego młodych instruktorów wymagała od nauczycieli ograniczenia przywilejów związanych z posiadaniem autorytetu nadanego im przez odgórnie przyjętą hierarchizację systemu oświaty. W kontakcie z uczniem zdolnym, nauczyciel nie posiadał już monopolu na wiedzę i jedyne właściwe rozwiązanie, lecz sam przyjmował pozycję aktywnego i gorliwego ucznia. Chcąc zyskać przychylność młodego instruktora, nauczyciel dążył do wypracowania nowego rodzaju autorytetu, opartego na dialogicznej relacji z uczniem. Model nauczyciela-mentora został w ten sposób wyparty przez postawę nauczyciela-facilitatora, towarzyszącego uczniowi w procesie uczenia się. Tego typu autorytet bazuje na zaakcentowaniu podmiotowości ucznia, dąży do ciągłego rozwijania jego indywidualnych predyspozycji obecnych w każdej jednostce w formie załączkowej, co w filozofii określane było pojęciem *entelechia* [7].

Odwrócenie ról widoczne było najwyraźniej w procesie opracowywania i omawiania zadań algorytmicznych przez instruktorów dla nauczycieli. Jak pokazały badania ewaluacyjne przeprowadzone na zakończenie warsztatów, metoda odwróconej szkoły została pozytywnie odebrana przez uczestników szkolenia. 87% z nich wyraziło chęć wzięcia udziału w podobnych warsztatach, a 100% nauczycieli zostało zainspirowanych do podjęcia dalszych inicjatyw na rzecz rozwoju nauki programowania w swoim środowisku zawodowym¹.

Na potrzeby tego artykułu warto podkreślić, że praktyczną konsekwencją zorganizowania warsztatów metodyczno-dydaktycznych było przeszkolenie grupy nauczycieli i przygotowanie ich do roli prowadzących zajęcia z programowania, ale dostosowanie harmonogramu warsztatów nauczycielskich do programu wieloaspektowego kształcenia uczniów zdolnych stanowiło przede wszystkim drogę dla weryfikacji wypracowanych strategii w przekazywaniu umiejętności programowania.

Do najważniejszych wyzwań, wśród tych, jakie postawiono przed młodymi instruktorami, należało omawianie treści zadań przed publicznością, na którą składali się nauczyciele, opiekunowie oraz rówieśnicy. Oprócz konieczności przygotowania merytorycznego, instruktorzy uczyli się skutecznego i zrozumiałego prezentowania złożonych treści oraz redukcji tremy związanej z lękiem przed wystąpieniami

¹ Dane z raportu ewaluacyjnego projektu Ogólnopolska Akademia Programowania.

publicznymi. W tym celu uczestnicy obozów instruktorskich brali udział w warsztatach psychologicznych, podczas których zaznajamiali się z technikami autoprezentacji oraz redukcji napięć wywołanych stresem przed społeczną ekspozycją. Skutkiem powiązania posiadanej wiedzy z ładunkiem emocjonalnym, jaki niesło ze sobą wystąpienie przed nauczycielami, było uporządkowanie, przyswojenie oraz skuteczne zapamiętanie opracowanego materiału.

2.2. Młody instruktor a uczeń – nauka przez współpracę

Powyżej opisano, na czym polegała rola młodych instruktorów podczas warsztatów metodyczno-dydaktycznych dla nauczycieli. W rolę osoby kierującej procesem nauczania wcielał się młody instruktor również podczas prowadzenia zajęć w strefach programowania dla swoich młodszych kolegów, posiadających znacznie uboższe kompetencje z zakresu algorytmiki i programowania. Niejednokrotnie zdarzało się, że pomiędzy prowadzącym a uczestnikiem zajęć występowała nieznaczna różnica wieku. Za sprawą zmniejszenia dystansu dzielącego instruktora, który posiadał wiedzę i młodego adepta programowania, który dąży do jej pozyskania, ułatwiony był proces komunikacji. Podstawą sukcesu tego typu sposobu przekazywania wiedzy było z jednej strony solidne przygotowanie merytoryczne instruktorów, a z drugiej ich otwartość na dzielenie się uzyskaną wiedzą z innymi.

Oprócz wymiernych korzyści mierzonych przyrostem wiedzy i umiejętności beneficjentów projektu, prowadzenie zajęć przez instruktorów miało wartość w postaci kształtowania właściwych postaw społecznych oraz budowaniu środowiska koleżeńkiego. Dzięki zaangażowaniu młodego programisty w proces przyswajania wiedzy przez młodszych uczniów, zyskiwał on poczucie odpowiedzialności za innych, czując się zobligowanym do rzetelnego przekazania posiadanych umiejętności w zrozumiałym i przystępnym języku. W tym sensie możemy mówić o wychowawczej funkcji, jaka wiązała się z występowaniem ucznia zdolnego w roli instruktora; samodzielnie dochodził on do przekonania, że posiadanie wiedzy czy umiejętności wiąże się z obowiązkiem dzielenia się nią w atmosferze współpracy i życzliwości. Niełatwe zadanie przekazywania młodszemu zdobytej wiedzy wymagało od młodych instruktorów dostosowania języka i metod przekazu do poziomu zaawansowania i wieku odbiorców, poprzez co uwrażliwiała ich na potrzeby słabszego.

2.3. Proces dochodzenia do rozwiązania

Oprócz dostarczenia warunków do skutecznego przyswajania wiedzy w trakcie realizacji obozów instruktorskich oraz warsztatów, w Ogólnopolskiej Akademii Programowania wypracowano metodę pracy z uczniem, której nadrzędnym celem było ukierunkowanie jego wysiłku intelektualnego na samodzielne poszukiwanie rozwiązań stawianych mu wyzwań. Założono, że jednostką czasową procesu przyswajania wiedzy będzie jeden tydzień. Każdy dzień stanowił określony etap w procesie

poszerzania wiedzy na temat postawionego problemu. Rozszerzenie okresu dzielącego poznanie treści zadania od poznania jego prawidłowego rozwiązania, dawał przestrzeń do samodzielnych lub częściowo samodzielnych poszukiwań intelektualnych. Nawet jeśli uczeń tylko do pewnego momentu był w stanie aktywnie pracować nad danym zagadnieniem, po tygodniu prób i testowania własnych pomysłów, mógł „zamknąć” owy proces wiedząc jak należało rozwiązać zadanie i przejść do kolejnego według tego samego schematu.

W pierwszym dniu uczeń był zaznajamiany z treścią zadania za pośrednictwem platformy <http://programuj.edu.pl>.

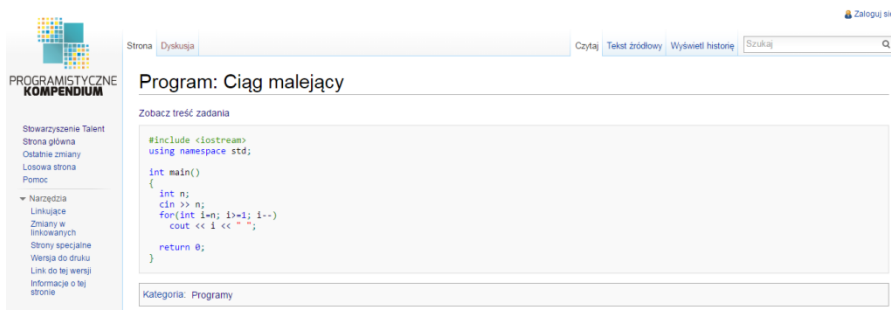
Rysunek 1. Treść przykładowego zadania zamieszczonego na platformie programuj.edu.pl

Na podstawie poznanej treści problemu, uczeń mógł rozpocząć dążenia do uzyskania prawidłowego rozwiązania przez samodzielne próby. Wspomniany etap realizowany był podczas zajęć i pod kontrolą nauczyciela prowadzącego.

Następnie uczeń, któremu nie udało się rozwiązać zadania otrzymywał dostęp do konta, przez które mógł rozwiązać zadanie samodzielnie po zakończeniu zajęć. Ewentualne pytania i wątpliwe kwestie mógł konsultować komunikując się z ekspertami za pośrednictwem poczty elektronicznej. Kolejnego dnia otrzymywał omówienie zadania, które skutecznie ułatwiało rozwiązanie problemu.

Rysunek 2. Omówienie przykładowego zadania na platformie programuj.edu.pl

Ostatnim etapem zgłębiania nowego problemu było opublikowanie wzorcowego rozwiązania.



Rysunek 3. Rozwiązanie wzorcowe na platformie programuj.edu.pl

2.4. Osiągnięcia uczestników Ogólnopolskiej Akademii Programowania

Wypracowane sposoby nauczania programowania wykorzystane podczas realizacji Ogólnopolskiej Akademii Programowania, przyniosły trwałe rezultaty w postaci znaczącego przyrostu wiedzy i umiejętności uczestników projektu. Znaczna część uczniów, którzy brali udział w przedsięwzięciu w roli młodych instruktorów lub uczestników, uzyskała wysokie wyniki w ogólnopolskich oraz międzynarodowych konkursach programistycznych.

Poniżej zamieszczono listę laureatów wybranych zawodów wraz z rolami, jakie pełnili w Ogólnopolskiej Akademii Programowania:

IX Junior Balcan Olympiad in Informatics, wrzesień 2015

<http://jboi2015.cs.org.mk/content/results>

- Juliusz Pham – złoty medalista – instruktor Ogólnopolskiej Akademii Programowania
- Kacper Walentynowicz – srebrny medalista – instruktor Ogólnopolskiej Akademii Programowania
- Marek Skiba – srebrny medalista – uczestnik Ogólnopolskiej Akademii Programowania

Romanian Master of Informatics, październik 2015

http://rmi.lbi.ro/rmi_2015/index.php?id=results_info

- Jakub Boguta – złoty medalista – instruktor Ogólnopolskiej Akademii Programowania
- Paweł Burzyński – srebrny medalista – instruktor Ogólnopolskiej Akademii Programowania
- Marek Żochowski – srebrny medalista – instruktor Ogólnopolskiej Akademii Programowania

XXIII Olimpiada Informatyczna, kwiecień 2016

http://www.oi.edu.pl/I/23oi_3etap_wyniki/

- Mateusz Radecki – laureat I miejsca – instruktor Ogólnopolskiej Akademii Programowania
- Paweł Burzyński – laureat I miejsca – instruktor Ogólnopolskiej Akademii Programowania
- Juliusz Pham – laureat I miejsca – instruktor Ogólnopolskiej Akademii Programowania
- Mariusz Trela – laureat I miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Maciej Sypetkowski – laureat I miejsca – instruktor Ogólnopolskiej Akademii Programowania
- Franciszek Budrowski – laureat II miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Piotr Kowalewski – laureat II miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Michał Górniak – laureat II miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Piotr Grabowski – laureat II miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Jakub Boguta – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Anadi Agrawal – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Piotr Borowski – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Tomasz Nowak – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Konrad Staniszewski – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Maciej Nadolski – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Jakub Bartmiński – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania
- Michał Niciejewski – laureat III miejsca – uczestnik Ogólnopolskiej Akademii Programowania

X Olimpiada Informatyczna Gimnazjalistów, maj 2016

- Mateusz Masłowski – złoty medalista – uczestnik Ogólnopolskiej Akademii Programowania

- Adrian Górecki – srebrny medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Mieszko Grodzicki – srebrny medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Justyna Jaworska – brązowa medalistka – uczestniczka Ogólnopolskiej Akademii Programowania
- Jakub Wornbard – brązowy medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Mateusz Smoliński – brązowy medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Marcin Mordecki – brązowy medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Antoni Strychalski – brązowy medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Marek Skiba – brązowy medalista – uczestnik Ogólnopolskiej Akademii Programowania

Baltic Olympiad in Informatics, maj 2016

<https://www.cs.helsinki.fi/group/boi2016/results.shtml>

- Mariusz Trela – złoty medalista – uczestnik Ogólnopolskiej Akademii Programowania
- Franciszek Budrowski – złoty medalista – instruktor Ogólnopolskiej Akademii Programowania
- Juliusz Pham – złoty medalista – instruktor Ogólnopolskiej Akademii Programowania
- Paweł Burzyński – srebrny medalista – instruktor Ogólnopolskiej Akademii Programowania

Podsumowanie

Prezentacja elementów metody dydaktycznej w nauce programowania, które realizatorzy Ogólnopolskiej Akademii Programowania wykorzystali w procesie kształcenia uczniów wszystkich poziomów edukacyjnych, miało na celu zwrócenie uwagi na konieczność uwzględnienia roli samego ucznia w procesie nauczania oraz włączenia aspektów jego rozwoju osobowościowego w tworzeniu strategii prowadzenia zajęć.

Poza wyraźnym przyrostem wiedzy i umiejętności uczniów w zakresie nabywania kompetencji programistycznych, za sprawą opisanych sposobów przyswajania wiedzy, uczniowie mają możliwość rozwijania kompetencji miękkich, takich jak umiejętność krytycznego i kreatywnego myślenia, zdolność selekcjonowania infor-

macji czy intelektualna ciekawość świata, które stanowią podstawę funkcjonowania w społeczeństwie informacyjnym XXI wieku².

Literatura

1. Goban-Klas T., Sienkiewicz P., *Spółeczeństwo informacyjne: Szanse, zagrożenia, wyzwania*, Wydawnictwo Postępu Telekomunikacji, Kraków 1999.
2. Grzegorzewska, Emocje w procesie uczenia się i nauczania, *Teraźniejszość-Człowiek-Edukacja*, 1(57)2012.
3. Harel D., *Rzecz o istocie informatyki. Algorytmika*, WN-T, Warszawa 2000.
4. Rostkowska M., Odwrócona lekcja, czyli praca z otwartymi zasobami edukacyjnymi, *MERITUM* 4/2012.
5. Sysło M.M. (red.), *Homo informaticus, czyli człowiek w z informatyzowanym świecie*, WWSI, Warszawa 2012.
6. Sysło M.M., *Myślenie komputacyjne. Nowe spojrzenie na kompetencje informatyczne*, (w:) M.M. Sysło, A.B. Kwiatkowska (red.). *Informatyka w edukacji. Informatyka dla wszystkich od najmłodszych lat*, Wydawnictwo UMK, Toruń 2014.
7. Żlobicki W., *Gestalt. Filozoficzne i psychologiczne inspiracje dla pedagogiki*, http://www.gestalt-dolnyslask.pl/teksty/W_Zlobicki.pdf ostatni dostęp 13.06.2016r.
8. <https://ec.europa.eu/digital-single-market/desi> ostatni dostęp 13.06.2016r.

² M.M. Sysło, *Komputer – obiekt i narzędzie edukacji. Poznawcze walory informatyki i technologii informacyjno-komunikacyjnej*, s. 217, w [5].