



MARIUSZ NYCZ¹, SARA NIENAJADŁO², MIROSLAW HAJDER³

Koncepcja niskokosztowej platformy edukacyjnej

The Concept of Low-cost Educational Platform

¹ Doktor inżynier, Politechnika Rzeszowska im. Ignacego Łukasiewicza, Katedra Energoelektroniki, Elektroenergetyki i Systemów Złożonych, Polska

² Inżynier, Politechnika Rzeszowska im. Ignacego Łukasiewicza, Polska

³ Doktor inżynier, Wyższa Szkoła Informatyki i Zarządzania w Rzeszowie, Polska

Streszczenie

W pracy przedstawiono koncepcje niskokosztowej platformy edukacyjnej. Autorzy prezentują rozwiązania sprzętowe oraz programowe, których głównym celem jest podniesienie jakości edukacji informatycznej w szkołach podstawowych i gimnazjalnych.

Słowa kluczowe: Raspberry Pi, Scratch, nauka programowania

Abstract

This article presents concept of low-cost educational platform. The authors introduce software and hardware solutions which can help increase quality of IT education in primary and middle schools.

Keywords: Raspberry Pi, Scratch, programming learning

Wstęp

Rozwój technologii informatycznych w minionych dziesięcioleciach odgrywa kluczowe znaczenie nie tylko w dziedzinie gospodarki, ale praktycznie we wszystkich sferach życia codziennego. Według danych GUS dla 2016 r. w Polsce 80,1% gospodarstw domowych było wyposażonych w komputer, przy czym 80,4% posiadało dostęp do internetu. W odniesieniu do przedsiębiorstw statystycznie 94,7% wykorzystywało komputery w swojej działalności oraz 93,7% posiadało dostęp do internetu (GUS, 2016). W najbliższych latach należy się spodziewać dalszego wzrostu znaczenia ICT, których zastosowanie nie ogranicza się tylko do branży informatycznej, ale jest obecne praktycznie we wszystkich gałęziach przemysłu. Implikuje to zapewnienie wykwalifikowanej kadry, której wiedza informatyczna nie będzie się ograniczać do szablonowej obsługi

systemów informatycznych ale posiadzie umiejętności rozwiązywania problemów z różnych dziedzin z wykorzystaniem narzędzi wywodzących się z informatyki (Rada ds. Informatyzacji Edukacji, 2015; Hajder, Nycz, Kolbusz, 2014).

Wyzwania i problemy edukacji ICT

Bazując na prognozach, zapowiada się, że w najbliższych latach gospodarka będzie narażona na niedobór kadry z odpowiednim przygotowaniem informatycznym. Dlatego konieczna staje się zmiana procesu edukacji, który obecnie jest ukierunkowany na nauczanie obsługi komputera z perspektywy administracyjnej, w kierunku rozwiązywania problemów analitycznych. Wymusza to nowe podejście do programu nauczania, w którym duży nacisk kładziony będzie na analizy i rozwiązywanie problemów poprzez zastosowanie odpowiednich technik obliczeniowych z wykorzystaniem samodzielnie opracowanych programów.

Dostosowanie oferty edukacyjnej do aktualnego zapotrzebowania w sferze edukacji informatycznej powoduje wygosparowanie przez samorządy dodatkowych nakładów potrzebnych na pokrycie ewentualnych zakupów sprzętu komputerowego, oprogramowania czy szkoleń nauczycieli. Dodatkowe koszty będą bardzo częstą argumentacją uniemożliwiającą płynne wdrożenie procesu modernizacji zajęć informatycznych. Dlatego niezwykle ważne jest opracowanie koncepcji platformy edukacyjnej, która nie będzie powodować dodatkowego obciążenia dla samorządów. Zaproponowana koncepcja platformy realizowana byłaby przy użyciu modułowego minikomputera Raspberry Pi oraz oprogramowania dydaktycznego na licencji *open source* czy *free software*, które również z powodzeniem można zaimplantować na platformie sprzętowej będącej w posiadaniu szkół.

Sprzętowa platforma edukacyjna oparta na Raspberry PI

Platforma Raspberry Pi jest minikomputerem wielkości karty kredytowej, który został zaprojektowany na Uniwersytecie Cambriach jako narzędzie ukierunkowane do projektowania niskokosztowego laboratorium komputerowego przeznaczonego do prowadzenia zajęć informatycznych. Raspberry Pi jest również uniwersalną platformą o szerokim przeznaczeniu, której zastosowanie nie ogranicza się tylko do nauki programowania, ale może być z powodzeniem wykorzystana np. jako domowe centrum rozrywki, konsola do gier, komputer pod kontrolą systemu Linuksa oraz Windows 10, sterownik do systemów w inteligentnym domu.

Architektura minikomputera na bazie platformy Raspberry PI

Raspberry Pi jest minikomputerem dedykowanym do wspierania nauki informatyki, który opracowany został przez Raspberry Pi Foundation. Urządzenie miało premierę w 2012 r. i jest oparte na procesorze Broadcom BCM2835, który zbudowany jest na architekturze ARM. Do tej pory stworzono 6 modeli Rasp-

berry Pi, z których najnowszym jest Raspberry Pi 3. Ten model minikomputera został wyposażony w moduł Bluetooth 4.1 oraz moduł Wi-Fi wspierający 802.11n, dzięki czemu możliwości wykorzystania urządzenia znacząco wzrosły. Ponadto wersja 3 jest dużo bardziej wydajna w stosunku do swoich poprzedników, posiada 4-rdzeniowy procesor 1.2GHz 64-bit quad-core ARMv8 oraz 1GB pamięci RAM. Mikrokomputer może być zasilany przez złącze micro USB bądź z wykorzystaniem ładowarki telefonu komórkowego. Raspberry PI wyposażony jest w kompozytowe wyjście video RCA oraz HDMI, które oprócz transmisji wideo ma możliwość przesyłania dźwięku. Złącze Raspberry Pi HD video camera zapewnia możliwość obsługi dedykowanej kamerki. Rozbudowę platformy o dodatkową funkcjonalność umożliwi umieszczony na płycie port GPIO (Świostek, Spicer, 2013; Bates, 2015).

W tabeli 1 przedstawiono charakterystykę podstawowych parametrów dla 3 wersji.

Tabela 1. Charakterystyka parametrów dostępnych modeli Raspberry Pi

Model	Procesor	Pamięć RAM	LAN	USB	Cena
Raspberry Pi	700 MHz ARMv6 Single Core	512 MB SDRAM	100 Mbit	4	30\$
Raspberry Pi 2	900 MHz ARMv7 Quad Core	1 GB DDR2	100 Mbit	4	35\$
Raspberry Pi 3	1 GHz ARM Cortex- A53 64 Bit Quad Core	1GB DDR2	100 Mbit	4	45\$

Źródło: opracowanie własne.


Platforma do nauki programowania dla dzieci oparta na interpretowanym wizualnym języku programowania – Scratch

W podejściu konstruktywistycznego modelu nauczania, w którym od ucznia wymaga się logicznego i strategicznego myślenia w procesie rozwiązywania problemów w dużym stopniu zależy ono od wybranych strategii dydaktycznych. Jedną z takich strategii jest nauka programowania, która wymaga większego zaangażowania uczniów podczas rozwiązywania problemów o charakterze twórczym niż w przypadku lekcji polegających na edukacji administracyjnej obsługi komputera. Konieczny w takiej sytuacji staje się odpowiedni dobór narzędzi edukacyjnych, które w sposób łatwy i przejrzysty umożliwią realizowanie procesu dydaktycznego na bazie nauki programowania. Wybrany język programowania powinien być oparty na programowaniu wizualnym i nieskomplikowanej strukturze. Powyższe cechy spełnia język Scratch, który został opracowany przez Massachusetts Institute of Technology (MIT) przy wsparciu finansowym National Science Foundation (NSF). Scratch jest skierowany przede wszystkim dla dzieci w wieku 6–15 lat i jego konstrukcja umożliwia naukę programowania przez łączenie odpowiednich bloków funkcjonalnych podobnie jak budowa konstrukcji z klocków Lego. Uczniowie, tworząc programy, uczą się ważnych pojęć

matematycznych i komputerowych, które wymuszają na nich twórcze myślenie, logiczne rozumowanie, rozwiązywanie problemów i umiejętność współpracy.

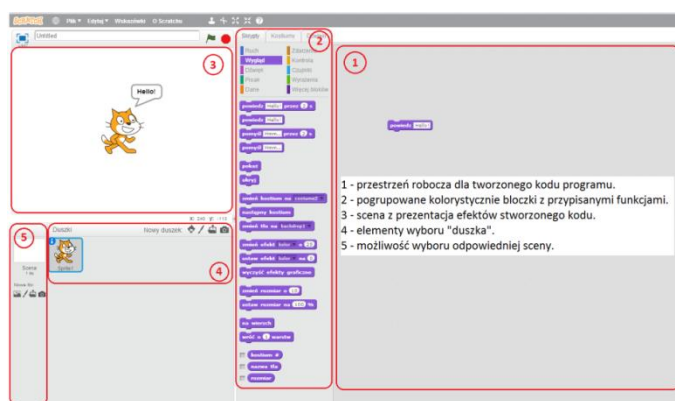
Scratch dostarcza przyjazny interfejs, w którym uczniowie zamiast pisać kod, mogą manipulować „blokami” i z ich pomocą tworzyć: gry, animowane filmiki, historie, symulacje, animowane pocztówki. Na przykład aby wyświetlić słowo *Hello!* na ekranie, w językach tekstowych należy napisać odpowiednią składnię polecenia, co w przypadku uczniów zaczynających przygodę z nauką programowania może być nie tylko trudne, ale i zniechęcające do dalszej nauki. W języku Scratch wystarczy umieścić odpowiedni blok, co pokazano w tabeli 2. Dodatkowym atutem języka wizualnego jest możliwość udostępniania on-line gotowych projektów na platformie MIT Scratch (Coravu, Marian, Ganea, 2015).

Tabela 2. Porównanie składni polecenia dla różnych języków programowania

Składania polecenia	Język programowania
<code>print('Hello!')</code>	Python
<code>std::cout << "Hello!" << std::endl;</code>	C++
<code>System.out.print("Hello!");</code>	Java
	Scratch

Źródło: opracowanie własne.

Programując w środowiska Scratch, uczeń ma możliwość natychmiastowej informacji zwrotnej z przygotowywanych programów, co pozwala szybko i łatwo sprawdzić wynik działania programu. Struktura wizualna sprawia, że z łatwością można śledzić przepływ programów i je udoskonalać. W konsekwencji programowanie jest przejrzyste oraz motywujące i dodatkowo zachęca do samodzielnej nauki. Jedynym ograniczeniem jest kreatywność i wyobraźnia (Marji, 2014). Na rysunku 1 przedstawiono interfejs programu.



Rysunek 1. Interfejs on-line aplikacji Scratch

Źródło: opracowanie własne.

Podsumowanie

Postępująca cyfryzacja spowodowała nieskrępowany dostęp do urządzeń i treści multimedialnych. W szczególności dotyczy to dzieci w wieku szkolnym. Dorastając w otoczeniu różnych rodzajów technologii, dzieci już w wieku przedszkolnym opanowują podstawową obsługę komputera, tabletu. Dlatego tak istotne jest, aby lekcje informatyki były ukierunkowane na rozwijanie własnej kreatywności, a także realizowanie aktywnego modelu uczenia się. Wymaga to nowoczesnego podejścia do procesu nauczania zawierającego następujące cechy:

- Zajęcia prowadzone są w sposób interaktywny, zapewniający integrację i dający możliwość prowadzenia dyskusji.
- Proces nauczania w głównej mierze powinien być oparty na pracy w grupie.
- Zajęcia ukierunkowane na realizację projektów.
- Obecność nauczyciela nie ogranicza się tylko to wizyty w klasie, ale uczniowie mają możliwość kontaktu z wykorzystaniem narzędzi ICT w sytuacji wystąpienia problemów.
- Model nauczania i uczenia się powinien być realizowany w sposób dynamiczny, gdzie jest zapewniona odpowiednia równowaga pracy indywidualnej i grupowej, a także istnieje możliwość dostosowania postępu nauki w miarę przyswajanych treści.

Literatura

- Bates, D. (2015). *Raspberry Pi Projects for Kids*. Birmingham: Packt Publishing Ltd.
- Coravu, L., Marian, M., Ganea, E. (2015), *RoEduNet International Conference – Networking in Education and Research*. Craiova: Scratch and recreational coding for kids.
- GUS (2016). *Spółeczeństwo informacyjne w Polsce w 2016 r.* Szczecin.
- Hajder, M., Nycz, M., Kolbusz, J. (2014). *Innowacyjna Gmina. Informatyka w jednostkach samorządu terytorialnego*. Rzeszów: Wyd. WSliZ.
- Marji, M. (2014). *Learn to Program with Scratch*. San Francisco: No Starch Press.
- Rada ds. Informatyzacji Edukacji (2015). *„Powszechne kształcenie informatyczne w polskim systemie edukacji*. Warszawa.
- Świostek, A., Spicer, M. (2013). Pobrane z: <http://www.komputerswiat.pl/jak-to-dziala/2013/08/wszystko-o-raspberry-pi.aspx> (20.05.2017).