

## SPRAWOZDANIE Z REALIZACJI PROJEKTU INNOWACJI PEDAGOGICZNEJ

### Informatyka dla twórców i konsumentów

1. Miejsce realizacji: **Zespół Szkół Integracyjnych nr 1 im. Roberta Oszka w Katowicach, Gimnazjum Integracyjne nr 36.**
2. Autor i realizator innowacji: **Krystian Płotkowski** – nauczyciel ZSI nr 1 w Katowicach.
3. Nazwa innowacji: „**Informatyka dla twórców i konsumentów**”
4. Rodzaj innowacji: **programowo – metodyczny i organizacyjny.**
5. Zakres i adresat: trzeci etap edukacyjny – gimnazjum; gimnazjaliści rozpoczynający naukę w gimnazjum w klasie Ic.
6. Czas realizacji:  
Data rozpoczęcia innowacji: **1 września 2016 r.**  
Data zakończenia innowacji: **30 czerwca 2019r.**

Czas trwania innowacji: **3 lata** (jedna godzina lekcyjna tygodniowo w klasie pierwszej, po dwie godziny dydaktyczne tygodniowo w klasie drugiej i trzeciej).

7. Program na którym oparta jest innowacja: „**Informatyka. Lekcje z komputerem**” Wandy Jochemczyk, Witolda Kranasa i Mirosława Wyczółkowskiego. WSiP.
8. Opis realizacji innowacji w roku szkolnym 2016/2017:

Realizacja innowacji rozpoczęła się w drugiej połowie października 2016 roku i objęła 19 uczniów klasy Ic gimnazjum w dwóch równoległych i niemal równolicznych grupach. W grupie pierwszej zrealizowano w całym roku szkolnym 2016/2017 29 lekcji, a w grupie drugiej 28 lekcji. Zgodnie z programem autorskim „Informatyka dla twórców i konsumentów” objęły one trzy działy: Lekcje z komputerem, Lekcje z edytorem tekstów i Lekcje programowania w Scratch.

Lekcje z komputerem objęły podstawy: jednostkę informacji i jej krotności, zapis binarny liczb, konwersje decymalnie – binarne, folderowe polecenia systemowe z elementami programowania systemowego, adresowanie IP i domenowe, budowa sieci komputerowych z klasami podsieci i ich maskami. Uczniowie programowali struktury folderowe korzystając z poleceń systemu DOS.

Lekcje z edytorem tekstów objęły: zasady poprawnej edycji tekstów z właściwą terminologią, charakterystykę czcionek i ich parametrów, edycję równań, wstawianie i edytowanie różnych obiektów w dokumencie, tworzenie algorytmów i ich graficznych ilustracji (sieci działań).

Lekcje ze Scratchem objęły omówienie zasad działania języka programowania, algorytmy sortowania danych różnymi technikami (bąbelkowe, przez porównanie, głupie itp.), pokaz możliwości multimedialnych języka Scratch i programowe rozwiązywanie różnych problemów: *Kalejdoskop*, *Kwadrat*, a po wprowadzeniu zmiennych i list – *Pole prostokąta*, *Objęto-*

*ści brył (kuli, walców i stożków), Generatory pseudolosowe, Tabliczka mnożenia, Kalkulator i Rysowanie w Scratchu.*

9. W pierwszym roku wdrażania pilotażu w zasadzie osiągnięto zamierzone cele ogólne i niektóre edukacyjne, czyli:

- a. Stymulowanie rozwoju intelektualnego ucznia.
- b. Przygotowanie ucznia do ról konsumenta i twórcy produktów informatycznych (aplikacji).
- c. Rozbudzenie twórczych aspiracji uczniów w różnych dziedzinach życia w oparciu o algorytmiczne i programowe narzędzia informatyki.
- d. Wdrażanie do pracy zespołowej przy odpowiednim poziomie komunikacji.
- e. Przekonanie uczniów do posługiwania się metodami algorytmicznymi i programowymi, warunkującymi sukces w ich przyszłości zawodowej i życiowej.
- f. Zrozumienie przez uczniów podstawowych zasad działania i budowy sprzętu informatycznego i oprogramowania.
- g. Opanowanie przez uczniów umiejętności właściwego dobierania narzędzi informatycznych do wykonywanych zadań.
- h. Praktyczne opanowanie przez uczniów gromadzenia informacji i efektywnego jej sortowania.

10. Wnioski i sugestie:

- a. W pierwszym roku wdrażania pilotażu dysponuję jedną lekcją tygodniowo. To zbyt mało, aby wykreować niezbędne nawyki programisty. Należy rozważyć możliwość koncentracji zajęć. W klasie drugiej i trzeciej będę dysponował dwiema lekcjami i wiąże z tym duże nadzieje.
- b. Uczniowie bardzo chętnie, sprawnie i szybko opanowują sztukę programowania, jeżeli dydaktykę realizuje się przez przykład lub przez bardzo skonkretyzowane i jasne, ale poważne zadania.
- c. Pod koniec roku szkolnego osiągnąłem fazę, w której przed uczniami stawiam problem (np. tabliczka mnożenia dla młodszego rodzeństwa z nagrodą dźwiękową) a oni sami tworzą program.
- d. W bardziej złożonych problemach ograniczam swoją rolę do podania zestawu procedur, które należy wykorzystać w programie.
- e. Ewaluacja na poziomie doboru środowiska programistycznego wypada bardzo dobrze – zarówno w środowisku tekstowym (DOS), jak i graficznym (Scratch) – uczniowie lubią rozkazywać maszynie w dowolny sposób, a jeżeli ona wykonuje ich polecenia to jest świetnie. Problemy pokonywane są błyskawicznie, ponieważ uczniowie na lekcji mogą

się ze sobą komunikować (oczywiście z wyjątkiem czasu sprawdzianów). Jest to dla mnie wymarzona okazja śledzenia sposobów myślenia poszczególnych uczniów i tworzenia okazji do pracy grupowej. Jeżeli chodzi o trwałość pozyskanej wiedzy, to trzeba poczekać; na razie jest w porządku – wprowadzone w listopadzie polecenia systemowe (akronimy kilkuliterowe typu DIR, CD, MD, RD, TREE) funkcjonują w środowisku uczniowskim i zostały zapamiętane trwale mimochodem.

## 11. Przykłady prac uczniowskich:

### Przykład zadania:

Stwórz program, w którym użytkownik będzie zgadywał losowo wybraną liczbę przez komputer. Będziesz potrzebował duszka, który powie użytkownikowi, że pomyślał sobie o pewnej liczbie od 1 do 100, a następnie poprosi go o odgadnięcie tej liczby. Duszek powinien mówić "Za dużo" i "Za mało" w momencie, gdy użytkownik będzie źle zgadywał. Gdy użytkownik odgadnie wylosowaną liczbę, duszek powinien zmienić swój kostium. Skorzystaj z bloczków widocznych na rysunku poniżej

Pol



### Przykładowy kalkulator:

# KALKULATOR

