



KARTA KOŃCOWEJ OCENY OPISOWEJ KOMPETENCJI UCZNIA

SZKOLENIE: „ZAJĘCIA PROGRAMOWANIA W KLASACH I-III SZKOŁY PODSTAWOWEJ W ŚRODOWISKU SCRATCH”

w ramach projektu „Razem odkryjmy świat programowania – szkolenia dla nauczycieli i uczniów z podregionu suwalskiego” Nr umowy POPC.03.02.00-00-0145/17 współfinansowanego ze środków Europejskiego Funduszu Rozwoju Regionalnego, Działanie 3.2 „Innowacyjne rozwiązania na rzecz aktywizacji cyfrowej” w ramach Programu Operacyjnego Polska Cyfrowa na lata 2014-2020.

NAZWA SZKOŁY/PLACÓWKI OŚWIATOWEJ.....

IMIĘ I NAZWISKO NAUCZYCIELA.....

IMIĘ I NAZWISKO UCZNIA

Uwaga! Proszę wstawić znak „X” w rubryce odpowiedniego poziomu opanowania kompetencji przez ucznia.

Lp	KOMPETENCJE UCZNIA (wiadomości i umiejętności)	POZIOM WIEDZY LUB UMIEJĘTNOŚCI UCZNIA		
		niski	średni	wysoki
1.	Definiowanie problemu / sytuacji problemowej samodzielnie lub w grupie. Uczeń opanował: <ul style="list-style-type: none"> • oznaczanie i nazywanie położenia punktu według dwóch współrzędnych • kodowanie i rozkodowywanie obrazu wg. podanych współrzędnych • dodawanie oraz tworzenie nowego duszka • zasady prostych gier rozwijających logiczne myślenie typu: „Gra w statki”, „Kółko i krzyżyk” 			
2.	Analiza problemu/ sytuacji problemowej. Uczeń zna: <ul style="list-style-type: none"> • wybrane metody oraz zasady prowadzenia analizy problemu/sytuacji problemowej • kodowanie liter w systemie binarnym • odkodowywanie liter z systemu binarnego • zasady pracy na platformie GODZINA KODOWANIA • główne menu programu SCRATCH • narzędzia programu SCRATCH 			
3.	Szukanie różnych dróg rozwiązań/ problemu/ sytuacji problemowej. Uczeń potrafi: <ul style="list-style-type: none"> • poszukiwać różnych dróg rozwiązań problemu: podaje rozwiązanie jako schemat na papierze lub proponuje gotowy algorytm wg. własnego pomysłu • pisać proste polecenia i instrukcje w środowisku SCRATCH • rozrysować problem na jego mniejsze podproblemy, np. algorytm parzenia herbaty, mycie zębów • tworzyć tła sceny • korzystać z narzędzi edytora tła • implementować gry w środowisku Scratch 			
4.	Wybór najefektywniejszej (np. najszybszej, najkrótszej) drogi rozwiązania problemu/sytuacji problemowej. Uczeń potrafi:			



	<ul style="list-style-type: none"> znaleźć najefektywniejszą drogę rozwiązania zarówno w wybranym środowisku programowania, jak i poza nim (SCRATCH, gry edukacyjne, wizualne karty pracy, dyktanda graficzne) napisać skrypt krok po kroku napisać skrypt uproszczony z wykorzystaniem bloku „powtórz” stosować bloczek „zawsze” stosować bloczek „jeżeli” 			
5.	<p>Opracowanie algorytmu prowadzącego do rozwiązania problemu/ sytuacji problemowej. Uczeń rozumie i zna:</p> <ul style="list-style-type: none"> podstawowe pojęcia i ich znaczenie, typu <i>duszek, scena, skrypt, kostium, remiks, udostępnianie, algorytm, programowanie wizualne, blok funkcjonalny, serwis społecznościowy, lista kroków, pętla</i> - w sensie intuicyjnym (bez znajomości definicji) dostępne bloczki za pomocą, których można kodować różne sposoby uruchamiania skryptów 			
6.	<p>Sprawdzenie poprawności działania opracowanego algorytmu (czyli otrzymania zakładanego wyniku/ osiągnięcia zakładanego celu) poza środowiskiem wizualnego programowania lub innym środowiskiem programistycznym. Uczeń potrafi:</p> <ul style="list-style-type: none"> uruchomić algorytm modyfikować skrypt prostej gry dostrzec błędy w skrypcie poprawić błąd –dokonać debugowania 			
7.	<p>Tworzenie prostego programu będącego realizacją opracowanego algorytmu w środowisku wizualnego programowania lub innym środowisku programistycznym. Uczeń potrafi</p> <ul style="list-style-type: none"> pracować indywidualnie i w grupie narysować własną scenę stworzyć własnego duszka określić zdarzenia dotyczące zachowania duszka zmieniać kostiumy duszka napisać skrypt określać położenie duszka grać w gry edukacyjne poprzez pisanie skryptów: FLAPPY, MINECRAFT 			
8.	<p>Testowanie programu w środowisku wizualnego programowania lub innym środowisku programistycznym. Uczeń potrafi:</p> <ul style="list-style-type: none"> uruchomić program wprowadzać niezbędne parametry dostrzec błąd zaproponować rozwiązanie niwelujące błąd w skrypcie 			
9.	<p>Prezentacja rozwiązania problemu/ sytuacji problemowej. Uczeń potrafi:</p> <ul style="list-style-type: none"> omówić problem przedstawić zastosowane przez siebie rozwiązania ocenić stopień trudności dostrzec podobieństwa w innych rozwiązaniach dokonać samooceny wykonania zadania 			
PODSUMOWANIE – ŁĄCZNIE				

INTERPRETACJA WYNIKÓW:

- POZIOM NISKI** – uczeń w stopniu niezadowolającym opanował wiedzę i umiejętności przewidziane w programie „Zajęcia programowania w klasach I-III szkoły podstawowej w



środowisku SCRATCH” w zakresie programowania w wizualnym środowisku. W przyszłości należałoby ponownie przepracować z dzieckiem treści ujęte w ww. programie. W przypadku braku pozytywnych efektów wskazane jest przebadanie dziecka w poradni psychologiczno-pedagogicznej oraz stworzenie indywidualnego programu wspomagania rozwoju ucznia.

2. **POZIOM ŚREDNI** – uczeń w stopniu dobrym opanował wiedzę i umiejętności przewidziane w programie „Zajęcia programowania w klasach I-III szkoły podstawowej w środowisku SCRATCH” w zakresie programowania w wizualnym środowisku. Może mieć jeszcze problem z niektórymi zagadnieniami. Aby mogło w przyszłości sprostać wymaganiom, konieczne jest wprowadzenie dodatkowych działań w klasach programowo wyższych, mających na celu rozwijanie słabszych obszarów kompetencji. Zasadne byłoby zastosowanie indywidualnej dodatkowej pracy z uczniem nauczycieli lub rodziców.
3. **POZIOM WYSOKI** – uczeń w stopniu bardzo dobrym opanował wiedzę i umiejętności przewidziane w programie „Zajęcia programowania w klasach I-III szkoły podstawowej w środowisku SCRATCH” w zakresie programowania w wizualnym środowisku. Dzięki dalszej systematycznej pracy wychowawczej i edukacyjnej szkoły będzie w stanie pogłębiać swoje kompetencje w klasach programowo wyższych.

.....
(Miejscowość, dnia)

.....
(Czytelny podpis nauczyciela)

Dziękujemy za wypełnienie ankiety