**Przedmiotowy system oceniania**

**KLASA 8**

Podstawa programowa określa cele kształcenia, a także obowiązkowy zakres treści programowych i oczekiwanych umiejętności, które uczeń o przeciętnych uzdolnieniach powinien przyswoić na danym etapie kształcenia. Opisane w niej wymagania szczegółowe można przypisać do pięciu kategorii.

1. Analizowanie i rozwiązywanie problemów – problemy powinny być raczej proste i dotyczyć zagadnień, z którymi uczniowie spotykają się w szkole (np. na matematyce) lub
na co dzień; rozwiązania mogą przyjmować postać planu działania, algorytmu lub programu (nie należy wymagać od uczniów biegłości w programowaniu w jakimkolwiek języku).
2. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi – uczniowie powinni w trakcie lekcji bez większych problemów wykonywać konkretne zadania za pomocą dostępnego oprogramowania, w tym sprawnie korzystać z menu, pasków narzędzi i pomocy programów użytkowych i narzędziowych, oraz tworzyć dokumenty i przedstawiać efekty swojej pracy np. w postaci dokumentu tekstowego lub graficznego, arkusza, prezentacji, programu, baz danych czy wydruku.
3. Zarządzanie informacjami oraz dokumentami – uczniowie powinni umieć wyszukiwać informacje, porządkować je, analizować, przedstawiać w syntetycznej formie
i udostępniać, a także gromadzić i organizować pliki w sieci lokalnej lub w chmurze.
4. Przestrzeganie zasad bezpiecznej pracy z komputerem – uczniowie powinni przestrzegać regulaminu pracowni komputerowej oraz zasad korzystania z sieci lokalnej i rozległej,
a także rozumieć zagrożenia związane z szybkim rozwojem technologii informacyjnej.
5. Przestrzeganie prawa i zasad współżycia – uczniowie powinni przestrzegać praw autorskich dotyczących korzystania z oprogramowania i innych utworów, a podczas korzystania z sieci i pracy w chmurze stosować się do zasad netykiety.

Ocenianie uczniów na lekcjach informatyki powinno być zgodne z założeniami szkolnego systemu oceniania. Uczniom i rodzicom powinny być znane wymagania stawiane przez nauczycieli i sposoby oceniania. Niniejszy dokument zawiera najważniejsze informacje, które można zaprezentować na początku roku szkolnego. Ważne jest, aby standardowej ocenie towarzyszył opis osiągnięć ucznia – szczegółowe uwagi dotyczące sposobu rozumowania, podejścia do zagadnienia. Trzeba pamiętać, że treści programowe są różnorodne. Obejmują zarówno operowanie elementami algorytmiki, jak i posługiwanie się narzędziami informatycznymi, czyli technologią informacyjną. Umiejętności te należy oceniać
w sposób równorzędny, ponieważ zdarza się, że uczniowie, którzy świetnie radzą sobie z programami użytkowymi, mają duże trudności z rozwiązywaniem problemów w postaci algorytmicznej, i odwrotnie – uczniowie rozwiązujący trudne problemy algorytmiczne i potrafiący sprawnie programować słabo posługują się programami użytkowymi. Należy uświadamiać uczniom ich braki, ale wystawiając ocenę, przykładać większą wagę do mocnych stron.

Sprawdzając wiadomości i umiejętności uczniów, należy brać pod uwagę osiem form aktywności.

|  |  |  |
| --- | --- | --- |
| **Forma aktywności** | **Częstość formy aktywności** | **Uwagi** |
| zadania i ćwiczenia wykonywane podczas lekcji | na każdej lekcji | oceniać należy przede wszystkim zgodność efektu pracy ucznia nad zadaniami i ćwiczeniami z postawionym problemem (np. czy funkcja utworzona przez ucznia daje właściwy wynik), mniejsze znaczenie ma sposób rozwiązania |
| praca na lekcji | na każdej lekcji | oceniać należy sposób pracy, aktywność, przestrzeganie regulaminu pracowni |
| odpowiedzi ustne, udział w dyskusjach | czasami |  |
| sprawdziany | po każdym dziale | mogą mieć formę testu |
| prace domowe | czasami | jeśli praca domowa wymaga użycia komputera, należy przypomnieć uczniom, że w razie potrzeby mogą skorzystać z komputera np. w bibliotece lub w pracowni komputerowej – w trakcie zajęć dodatkowych |
| referaty, opracowania, projekty | czasami |  |
| przygotowanie do lekcji | w razie potrzeby | oceniać należy pomysły i materiały przygotowane do pracy na lekcji |
| udział w konkursach |  | nieobowiązkowa forma aktywności; przejście do kolejnych etapów powinno odpowiednio podwyższyć ocenę końcową |

**Opis wymagań ogólnych, które uczeń musi spełnić, aby uzyskać daną ocenę**

**Ocena celująca** **(6)** – uczeń wykonuje samodzielnie i bezbłędnie wszystkie zadania z lekcji oraz dostarczone przez nauczyciela trudniejsze zadania dodatkowe; jest aktywny i pracuje systematycznie; posiada wiadomości i umiejętności wykraczające poza te, które są wymienione w planie wynikowym; w konkursach informatycznych przechodzi poza etap szkolny; w razie potrzeby pomaga nauczycielowi (np. przygotowuje potrzebne na lekcję materiały pomocnicze, pomaga kolegom w pracy); pomaga nauczycielom innych przedmiotów
w wykorzystaniu komputera na ich lekcjach.

**Ocena bardzo dobra (5)** – uczeń wykonuje samodzielnie i bezbłędnie wszystkie zadania z lekcji; jest aktywny i pracuje systematycznie; posiada wiadomości i umiejętności wymienione w planie wynikowym; w razie potrzeby pomaga nauczycielowi (pomaga kolegom w pracy).

**Ocena dobra (4)** – uczeń wykonuje samodzielnie i niemal bezbłędnie łatwiejsze oraz niektóre trudniejsze zadania z lekcji; pracuje systematycznie i wykazuje postępy; posiada wiadomości i umiejętności wymienione w planie wynikowym.

**Ocena dostateczna (3)** – uczeń wykonuje łatwe zadania z lekcji, czasem z niewielką pomocą, przeważnie je kończy; stara się pracować systematycznie i wykazuje postępy; posiada większą część wiadomości i umiejętności wymienionych w planie wynikowym.

**Ocena dopuszczająca (2)** – uczeń czasami wykonuje łatwe zadania z lekcji, niektórych zadań nie kończy; posiada tylko część wiadomości i umiejętności wymienionych w planie wynikowym, jednak brak systematyczności nie przekreśla możliwości uzyskania przez niego podstawowej wiedzy informatycznej oraz odpowiednich umiejętności w toku dalszej nauki.

**Uwagi dodatkowe**

* Wymaga się od uczniów prowadzenia zeszytu . Konieczne jest systematyczne zapisywanie wykonanych w pracowni ćwiczeń w określonym miejscu, swoim folderze,
w sieci lokalnej lub w chmurze. Można też poprosić uczniów o przynoszenie na lekcje pamięci USB – w celu prowadzenia własnego archiwum plików.
* Warto zachęcać uczniów do samodzielnego oceniania swojej pracy – powinni umieć stwierdzić, czy ich rozwiązanie jest poprawne. W miarę możliwości należy uzasadniać oceny i dyskutować je z uczniami.
* Aby poprawić ocenę, uczeń powinien wykonać powtórnie najgorzej ocenione zadania (lub zadania podobnego typu) w trakcie prowadzonych w pracowni zajęć dodatkowych albo w domu, jeśli jest taka możliwość i można wierzyć, że dziecko będzie pracować samodzielnie.

**Katalog wymagań programowych na poszczególne oceny szkolne**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **1. Lekcje z HTML-em** |
| **1.1** | **Jak to zrobić w HTML-u i CSS?** | Programy do tworzenia stron WWW, wprowadzenie w historię języka znaczników hipertekstu (HTML) oraz kaskadowych arkuszy stylów (CSS), ogólna struktura dokumentu HTML, definiowanie stylów w dokumencie HTML (rodzaje arkuszy stylów, podstawowe znaczniki) | 2 | * z pomocą nauczyciela tworzy prosty dokument HTML.
 |
| 3 | * wprowadza w edytorze tekstu ustawienia dotyczące kodowania znaków;
* samodzielnie tworzy prosty dokument HTML.
 |
| 4 | * poprawnie stosuje elementy CSS.
 |
| 5 | * tworzy dokument HTML zgodnie z zaleceniami W3C;
* wyjaśnia specyfikę różnych rodzajów kaskadowych arkuszy stylów.
 |
| 6 | * przygotowuje prezentację wyjaśniającą rolę, jaką w historii języka HTML odegrali Tim Berners-Lee, Robert Cailliau, Håkon Wium Lie i Bert Bos, oraz cel powołania W3C.
 |
| **1.2** | **Strona w dobrym stylu** | Tworzenie dokumentu HTML z zastosowaniem CSS – definiowanie właściwości czcionki i akapitu, definiowanie jednostek miar i kolorów, osadzanie elementów graficznych, korzystanie ze znaków specjalnych | 2 | * stosuje style wpisane w celu sformatowania tekstu.
 |
| 3 | * stosuje znaki specjalne (zwłaszcza &nbsp;).
 |
| 4 | * stosuje różne jednostki miary;
* definiuje kolory różnych elementów dokumentu;
* osadza w dokumencie elementy graficzne.
 |
| 5 | * definiuje właściwości czcionek (krój czcionki, styl czcionki, wariant czcionki, wysokość czcionki, odstępy między literami, zmiana wielkości znaków);
* definiuje właściwości akapitu (odstępy między wyrazami, dekorowanie tekstu, wyrównanie tekstu w poziomie).
 |
| 6 | * wykorzystuje style wpisane, osadzone i zewnętrzne;
* stosuje wybór przez klasę.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **1.3** | **Strona interaktywna** | Wprowadzenie do dynamicznego HTML, tworzenie elementów interaktywnych za pomocą CSS i JavaScript, budowanie galerii z wykorzystaniem elementów interaktywnych | 2 | * wyjaśnia pojęcie „dynamiczny HTML”.
* z pomocą nauczyciela tworzy i umieszcza na stronach HTML elementy interaktywne w CSS z wykorzystaniem pseudoklasy :hover.
 |
| 3 | * samodzielnie tworzy i umieszcza na stronach HTML interaktywne elementy w CSSz wykorzystaniem pseudoklasy :hover.
 |
| 4 | * z pomocą nauczyciela tworzy i umieszcza na stronach HTML elementy interaktywne w JavaScript z wykorzystaniem zdarzeń onclick, onmouseover, onmouseout.
 |
| 5 | * samodzielnie tworzy i umieszcza na stronach HTML elementy interaktywne w JavaScript z wykorzystaniem zdarzeń onclick, onmouseover, onmouseout;
* samodzielnie tworzy interaktywną galerię fotografii.
 |
| 6 | * stosuje inne dynamiczne pseudoklasy CSS;
* tworząc elementy interaktywne, stosuje własne rozwiązania.
 |
| **1.4** | **Witryna WWW** | Rodzaje witryn WWW, porządkowanie kodu dokumentu HTML, tworzenie witryny przez połączenie poszczególnych dokumentów HTML systemem odnośników | 2 | * opisuje budowę adresu strony WWW;
* wyjaśnia znaczenie rozszerzenia domeny.
 |
| 3 | * wyjaśnia znaczenie nazwy index.htm;
* tworzy odnośniki tekstowe i graficzne do innych dokumentów.
 |
| 4 | * omawia strukturalną budowę dokumentu HTML;
* opisuje rolę znaczników: header, nav, article, section, aside, footer.
* z pomocą nauczyciela stosuje ww. znaczniki do tworzenia dokumentu HTML.
 |
| 5 | * samodzielnie stosuje ww. znaczniki do tworzenia poprawnej struktury dokumentu.
 |
| 6 | * tworząc witrynę WWW, pracuje samodzielnie i stosuje własne rozwiązania;
* kopiuje pliki składowe na serwer WWW i weryfikuje poprawność działania witryny.
 |
| **1.5** | **Prawo w internecie** | Prawo autorskie a ochrona wizerunku oraz twórczości (ochrona elementów serwisów i całych serwisów WWW, ochrona oprogramowania), wolne oprogramowanie, bezpieczeństwo w sieci | 2 | * wyjaśnia konieczność chronienia utworów (np. programów, zdjęć, stron WWW).
 |
| 3 | * wyjaśnia, na czym polega naruszenie praw autorskich i jak go uniknąć.
 |
| 4 | * wyjaśnia pojęcia „dozwolony użytek prywatny” i „ochrona wizerunku”.
 |
| 5 | * wyjaśnia, czym są wolne oprogramowanie i cztery rodzaje wolności.
 |
| 6 | * wyjaśnia praktyczne znaczenie najważniejsze punktów *Ustawy o prawie autorskim i prawach pokrewnych*.
 |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **2. Lekcje programowania** |
| **2.1** | **Pisz i powtarzaj** | Instalowanie programu Python, stosowanie polecenia print i pętli for | 2 | * pisze i uruchamia prosty program wypisywania tekstu na ekranie (polecenie print).
 |
| 3 | * stosuje pętlę for.
 |
| 4 | * wyjaśnia, jak działa funkcja range w zależności od liczby parametrów.
 |
| 5 | * rysuje szlaczki i figury, wykorzystując pętlę for, polecenie print.
 |
| 6 |  |
| **2.2** | **Programuj obliczenia** | Operacje matematyczne, typy zmiennych, definiowanie funkcji bez parametru | 2 | * opisuje i odpowiednio wykorzystuje operacje matematyczne.
 |
| 3 | * opisuje i odpowiednio wykorzystuje zmienne.
 |
| 4 | * definiuje proste funkcje bez parametru.
 |
| 5 | * rozwiązuje problemy z wykorzystaniem funkcji bez parametru.
 |
| 6 |  |
| **2.3** | **Sumuj liczby** | Operowanie zmiennymi, definiowanie funkcji z parametrem, stosowanie instrukcji warunkowej i podstawowych algorytmów na liczbach naturalnych | 2 | * zmienia wartość zmiennej.
 |
| 3 | * omawia działanie parametru w funkcji.
 |
| 4 | * definiuje funkcję z parametrem służącą do wyodrębnienia cyfr danej liczby czterocyfrowej i obliczenia ich sumy.
 |
| 5 | * definiuje funkcję z parametrem służącą do wyodrębnienia cyfr dowolnej liczby całkowitej i obliczenia ich sumy;
* opisuje działanie instrukcji warunkowej i wykorzystuje ją do zbadania podzielności liczb.
 |
| 6 | * samodzielnie rozwiązuje dodatkowe zadania programistyczne.
 |
| **2.4** | **Liczby nie tylko doskonałe** | Wykorzystywanie funkcji do obliczeń, moduł math | 2 | * rozumie problem znajdowania podzielników właściwych liczby.
 |
| 3 | * korzysta z modułu math.
 |
| 4 | * z pomocą nauczyciela definiuje funkcję obliczania sumy dzielników właściwych liczby podanej jako parametr.
 |
| 5 | * definiuje funkcję wypisywania liczb doskonałych;
* testuje działanie funkcji dla różnych parametrów.
 |
| 6 | * samodzielnie rozwiązuje dodatkowe zadania programistyczne.
 |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **2.5** | **Szukaj z Pythonem** | Wyszukiwanie elementu w zbiorze uporządkowanym i nieuporządkowanym, moduł random, stosowanie pętli while | 2 | * rozumie zasady gry *Zgadnij liczbę*;
* biorąc udział w grze, potrafi zastosować optymalną strategię.
 |
| 3 | * znajduje maksymalną liczbę kroków odgadywania danej liczby.
 |
| 4 | * losuje liczby całkowite z danego zakresu;
* wykorzystuje pętlę while do znajdowania sumy cyfr liczby.
 |
| 5 | * analizuje schemat blokowy algorytmu obliczania sumy cyfr dowolnej liczby;
* samodzielnie implementuje grę *Zgadnij liczbę* w Pythonie, korzystając ze wskazówek w podręczniku.
 |
| 6 | * samodzielnie rozwiązuje dodatkowe zadania programistyczne.
 |
| **2.6** | **Zrób porządek** | Porządkowanie elementów zbioru przez prosty wybór i zliczanie, wykorzystywanie list | 2 | * opisuje porządkowanie zbioru przez proste wybieranie i zliczanie.
 |
| 3 | * opisuje, czym jest lista, i potrafi z niej korzystać.
 |
| 4 | * korzysta z funkcji związanych z listami.
 |
| 5 | * definiuje funkcje zliczania.
 |
| 6 | * samodzielnie rozwiązuje dodatkowe zadania programistyczne.
 |
| **3. Lekcje z danymi** |
| **3.1** | **Jak to z Gaussem było**  | Sumowanie w arkuszu kalkulacyjnym, porządkowanie danych w tabelach, analizowanie danych zapisanych w arkuszu, obliczeń i prawidłowości | 2 | * korzysta z arkusza kalkulacyjnego w podstawowym zakresie.
 |
| 3 | * wykonuje w arkuszu proste obliczenia;
* wykorzystuje arkusz do szybkiego rozwiązywania zadań związanych z sumowaniem;
* wprowadza dane różnych typów;
* wprowadza i kopiuje proste formuły obliczeniowe;
* korzysta z funkcji Autosumowania.
 |
| 4 | * rozwiązuje w arkuszu proste zadania matematyczne.
 |
| 5 | * planuje wykonywanie obliczeń w arkuszu;
* analizuje dane zawarte w arkuszu w poszukiwaniu prawidłowości.
 |
| 6 | * samodzielnie formułuje wnioski.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **3.2** | **Liczby, potęgi, ciągi** | Wprowadzanie serii danych, formuł i funkcji do arkusza kalkulacyjnego, porównywanie ciągów liczbowych, włączanie ochrony arkusza | 2 | * rozumie, czym jest formuła i format liczbowy, i używa ich w zadaniu;
* drukuje tabele przygotowane w arkuszu.
 |
| 3 | * wprowadza do arkusza serie danych formuły i funkcje;
* odróżnia i wprowadza różne formaty liczbowe.
 |
| 4 | * planuje wykonywanie obliczeń w arkuszu;
* porównuje ciągi liczbowe i odnajduje występujące w nich prawidłowości.
 |
| 5 | * analizuje dane zawarte w arkuszu;
* tworzy prosty kalkulator matematyczny;
* uniemożliwia zmianę danych w arkuszu (włącza ochronę arkusza).
 |
| 6 | * samodzielnie formułuje wnioski.
 |
| **3.3** | **Z tabeli – wykres** | Rysowanie wykresów funkcji za pomocą kreatora wykresów arkusza kalkulacyjnego, wstawianie i formatowanie wykresu punktowego | 2 | * rozumie, czym jest wykres, i drukuje go wraz z tabelą danych.
 |
| 3 | * przygotowuje dane do wykonania wykresu funkcji liniowej.
 |
| 4 | * tworzy wykresy funkcji liniowych za pomocą kreatora wykresów.
 |
| 5 | * opisuje i formatuje elementy wykresu.
 |
| 6 | * samodzielnie formułuje wnioski.
 |
| **3.4** | **Przestawianie i przedstawianie danych** | Przeglądanie i sortowanie dużych zestawów danych w arkuszu kalkulacyjnym, tworzenie tabeli przestawnej, wykonywanie prostych obliczeń statystycznych i prezentowanie ich w arkuszu | 2 | * rozumie, czym jest funkcja, i z pomocą nauczyciela korzysta z kreatora funkcji.
 |
| 3 | * przegląda, sortuje i filtruje w arkuszu duże zestawy danych.
 |
| 4 | * samodzielnie korzysta z funkcji statystycznych LICZ.JEŻELI i CZĘSTOŚĆ.
 |
| 5 | * tworzy tabelę przestawną.
 |
| 6 | * samodzielnie formułuje wnioski.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **3.5** | **Dużo danych**  | Przeglądanie i analizowanie dużych zestawów danych w arkuszu kalkulacyjnym, zastosowanie wybranych funkcji statystycznych oraz linii trendu, przetwarzanie rozproszone i projekty realizowane w tym systemie | 2 | * korzysta z arkusza kalkulacyjnego w podstawowym zakresie.
 |
| 3 | * przegląda w arkuszu duże tabele i wyszukuje dane;
* korzysta z funkcji statystycznych ŚREDNIA, MIN, MAX i MEDIANA.
 |
| 4 | * omawia specyfikę przetwarzania rozproszonego i opisuje wybrane projekty.
 |
| 5 | * tworzy wykres zależności XY i wstawia linię trendu.
 |
| 6 | * samodzielnie formułuje wnioski.
 |
| **3.6** | **Moi znajomi**  | Wprowadzenie do pracy z kartotekową bazą danych – przygotowanie, filtrowanie, uzupełnianie, poprawianie i sortowanie danych, zastosowanie formularza do wpisywania danych | 2 | * wyjaśnia, czym jest kartotekowa baza danych.
 |
| 3 | * wpisuje dane do arkusza udostępnionego do edycji w chmurze.
 |
| 4 | * sortuje i filtruje dane;
* sprawnie wyszukuje dane o wybranych kryteriach.
 |
| 5 | * tworzy formularz w celu dopisywania lub poprawiania rekordów.
 |
| 6 | * rozbudowuje bazę danych;
* oblicza wystąpienia pewnych danych za pomocą wbudowanych funkcji.
 |
| **4. Lekcje z modelami** |
| **4.1** | **Kości zostały rzucone** | Wykorzystanie funkcji losowych w arkuszu kalkulacyjnym, przeprowadzanie symulacji procesu o losowym przebiegu | 2 | * wyjaśnia, czym jest doświadczenie losowe, i używa prostej funkcji losującej;
* drukuje wykresy obrazujące wyniki doświadczenia.
 |
| 3 | * korzysta z funkcji losowych w arkuszu;
* trafnie ocenia wynik prostego doświadczenia losowego.
 |
| 4 | * przeprowadza zadaną symulację prostego doświadczenia z użyciem funkcji losującej;
* wykonuje wykres wyników doświadczenia.
 |
| 5 | * samodzielnie planuje i przeprowadza symulację procesu o losowym przebiegu.
 |
| 6 | * samodzielnie planuje obliczenia i formułuje wnioski;
* proponuje doświadczenie losowe i zawczasu ocenia jego przebieg.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **4.2** | **Fraktale w Scratchu i w Pythonie** | Rysowanie drzew binarnych zwykłego i losowego w Scratchu i w Pythonie | 2 | * otwiera i analizuje projekt w Scratchu.
 |
| 3 | * opisuje algorytm tworzenia drzewa binarnego.
 |
| 4 | * z pomocą nauczyciela realizuje w Pythonie algorytm dla zwykłego drzewa binarnego.
 |
| 5 | * realizuje w Pythonie algorytm dla drzew binarnych zwykłego i losowego.
 |
| 6 | * tworzy własne wariacje programu, np. dodając parametry (dwa kąty odchylenia itp.).
 |
| **4.3** | **Fraktale w smartfonie** | Rysowanie płatka Kocha i trójkąta Sierpińskiego w środowisku App Lab | 2 | * otwiera i analizuje projekty w Scratchu.
 |
| 3 | * opisuje algorytmy tworzenia trójkąta Sierpińskiego i płatka Kocha.
 |
| 4 | * z pomocą nauczyciela realizuje przynajmniej jeden z algorytmów w środowisku App Lab.
 |
| 5 | * realizuje oba algorytmy w środowisku App Lab.
 |
| 6 | * realizuje własne pomysły rysunków fraktali w środowisku App Lab.
 |
| **4.4** | **Kolorowa płaszczyzna** | Programowanie gry w ciepło–zimno w Scratchu i w środowisku Processing JS Akademii Khana | 2 | * otwiera i analizuje projekt w Scratchu.
 |
| 3 | * opisuje algorytm rysowania.
 |
| 4 | * z pomocą nauczyciela realizuje algorytm w środowisku Processing JS Akademii Khana.
 |
| 5 | * korzysta z dokumentacji Processing JS i wprowadza własne zmiany.
 |
| 6 | * realizuje własne pomysły interaktywnej animacji.
 |
| **4.5** | ***Gra w życie*** | Symulacja procesu dla różnych ustawień początkowych | 2 | * uruchamia gotowe symulacje *Gry w życie* na wybranej stronie internetowej.
 |
| 3 | * opisuje zasady *Gry w życie*.
 |
| 4 | * eksperymentuje i obserwuje etapy życia na planecie.
 |
| 5 | * znajduje układy, w których populacja zachowuje się w określony sposób.
 |
| 6 | * realizuje własną symulację *Gry w życie* w wybranym języku programowania.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **4.6** | **Podróże z komputerem** | Korzystanie z map internetowych, transpozycja tabel w arkuszu kalkulacyjnym | 2 | * wskazuje serwisy i aplikacje zawierające mapy.
 |
| 3 | * w podstawowym zakresie korzysta z serwisów zawierających mapy.
 |
| 4 | * korzysta z serwisów zawierających mapy i przy ich pomocy planuje podróż;
* wyjaśnia, czym są GIS i GPS.
 |
| 5 | * wykonuje potrzebne obliczenia w arkuszu kalkulacyjnym i znajduje na mapie najbardziej centralnie położone miasto;
* wyjaśnia, czym jest transpozycja tabeli i jak ją można wykonać w arkuszu.
 |
| 6 | * samodzielnie planuje działania w arkuszu i formułuje wnioski;
* samodzielnie planuje podróż, porównuje i weryfikuje dane z różnych serwisów.
 |
| **5. Lekcje z mobilnym internetem** |
| **5.1** | **Mały robot – Android** | Omówienie narzędzi i aplikacji użytkowych wbudowanych w system Android oraz zewnętrznych, instalacja i obsługa Tiny Scanner – PDF Scanner App | 2 | * charakteryzuje podstawowe narzędzia systemu Android.
 |
| 3 | * szuka aplikacji w Sklepie Play;
* z pomocą nauczyciela instaluje aplikację zewnętrzną na urządzeniu mobilnym.
 |
| 4 | * instaluje aplikację na urządzeniu mobilnym z zachowaniem zasad bezpieczeństwa.
 |
| 5 | * biegle posługuje się samodzielnie zainstalowanym skanerem dokumentów.
 |
| 6 | * świadomie i celowo korzysta z wbudowanych i zewnętrznych aplikacji systemu Android.
 |
| **5.2** | **Ze smartfonem na piechotę** | Planowanie i dokumentowanie wycieczki z wykorzystaniem urządzenia mobilnego, publikowanie trasy wycieczki w internecie | 2 | * z pomocą nauczyciela instaluje aplikację Traseo.
 |
| 3 | * omawia podstawowe punkty regulaminu korzystania z usługi Traseo;
* z pomocą nauczyciela tworzy konto na portalu **www.traseo.pl**.
 |
| 4 | * samodzielnie tworzy konto na portalu **www.traseo.pl**;
* z pomocą nauczyciela rejestruje i publikuje przebytą trasę;
* podczas rejestracji trasy zaznacza ciekawe miejsca na mapie i dodaje zdjęcia.
 |
| 5 | * samodzielnie rejestruje i publikuje przebytą trasę.
 |
| 6 | * opisuje zarejestrowaną i opublikowaną trasę, stosując trafne i wyczerpujące komentarze.
 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr lekcji** | **Temat lekcji** | **Omawiane zagadnienia** | **Ocena** | **Zgodnie z wymaganiami programowymi uczeń:** |
| **5.3** | **Rozszerzona rzeczywistość – tuż obok** | Technologia rozszerzonej rzeczywistości i jej zastosowanie | 2 | * wyjaśnia, co oznaczają termin „rozszerzona rzeczywistość” oraz skrótowiec „AR”.
 |
| 3 | * korzysta z technologii AR;
* odróżni rozszerzoną rzeczywistość od rzeczywistości wirtualnej.
 |
| 4 | * podaje przykłady wykorzystania technologii AR.
 |
| 5 | * podaje przykłady sytuacji, w których zastosowanie technologii AR byłoby przydatne.
 |
| 6 | * wyszukuje aplikacje wykorzystujące technologię AR, instaluje je i omawia ich możliwości.
 |
| **5.4** | **Rozszerzona rzeczywistość – kosmos** | Wybrane aplikacje wykorzystujące technologię rozszerzonej rzeczywistości | 2 | * wyszukuje i opisuje omawiane na lekcji aplikacje.
 |
| 3 | * instaluje omawiane na lekcji aplikacje.
 |
| 4 | * wykorzystuje aplikacje, np. wykonując zdjęcia w aplikacji Spacecraft 3D.
 |
| 5 | * wyszukuje i obsługuje inne aplikacje wykorzystujące technologię AR.
 |
| 6 | * wyszukuje aplikacje wykorzystujące technologię AR, instaluje je i omawia ich możliwości.
 |
| **5.5** | **Ucz się w sieci – Akademia Khana** | Wykorzystanie portalu Akademii Khana do dokształcania się i rozwijania zainteresowań | 2 | * opisuje możliwości nauki informatyki w Akademii Khana;
* wyjaśnia pojęcie „MOOC”.
 |
| 3 | * znajduje serwisy oferujące MOOC;
* krótko charakteryzuje kursy informatyczne w Akademii Khana.
 |
| 4 | * znajduje kursy w serwisach oferujących MOOC;
* korzysta z kursów informatycznych w Akademii Khana.
 |
| 5 | * potrafi zanalizować przydatność kursów w serwisach oferujących MOOC.
 |
| 6 | * samodzielnie wykonuje ćwiczenia w ramach kursów informatycznych w Akademii Khana.
 |
| **5.6** | **Ucz się i rozwijaj zainteresowania w sieci** | Ciekawe serwisy wspomagające samodzielną naukę i rozwijanie zainteresowań – platforma Zooniverse.org, portale TED.com i Ed.TED.com | 2 | * w podstawowym zakresie korzysta ze wskazanych aplikacji i serwisów.
 |
| 3 | * w pełnym zakresie korzysta ze wskazanych aplikacji i serwisów.
 |
| 4 | * korzysta z samodzielnie znalezionych aplikacji i serwisów wspomagających naukę i rozwijających zainteresowania.
 |
| 5 | * buduje własną bazę wiedzy.
 |
| 6 | * prezentuje w klasie wyszukane aplikacje i serwisy wspomagające naukę i rozwijające zainteresowania i poddaje je krytycznej ocenie pod kątem użytkowości oraz przydatności.
 |