

Przedmiotowe zestawienie oceniania

KLASA 8

- I. Postanowienia ogólne Przedmiotowe Zasady Oceniania zostały opracowane na podstawie: 1. Rozporządzenia Ministra Edukacji Narodowej w sprawie warunków i sposobu oceniania, klasyfikowania i promowania uczniów w szkołach oraz przeprowadzania sprawdzianów i egzaminów w szkołach publicznych; 2. Programu nauczania zajęć komputerowych, 3. Podstawy programowej kształcenia ogólnego z zajęć komputerowych w drugim etapie kształcenia; 4. Wewnątrzszkolnych Zasad Oceniania; 5. Statutu szkoły.
- II. Przedmiotem oceny są: wiedza i umiejętności oraz wykorzystywanie własnych możliwości; wiadomości i umiejętności ucznia wynikające z podstawy programowej nauczania informatyki oraz wymagań programu nauczania; wysiłek wkładany przez ucznia; aktywność i systematyczność.
- III. Ogólne cele kształcenia:
 1. Rozwijanie umiejętności posługiwania się sprzętem komputerowym.
 2. Przygotowanie do korzystania ze środków techniki informacyjnej.
 3. Kształtowanie umiejętności posługiwania się językiem komputerowym
 4. Rozwijanie zainteresowań techniką i wdrażanie do świadomego korzystania z niej.
 5. Wskazanie użyteczności komputerów w nauce, pracy i zabawie.
- IV. Ocenie podlegają:
 1. Praca na lekcji: ćwiczenia praktyczne; odpowiedzi ustne (znajomość danych zagadnień, posługiwanie się terminami i pojęciami informatycznymi); prezentowanie samodzielnie opracowanych zagadnień; aktywność, systematyczność oraz jakość pracy; współpraca w grupie; stosowanie zasad bezpieczeństwa i właściwej organizacji pracy oraz higieny na stanowisku komputerowym.
 2. Sprawdziany i testy wiadomości i umiejętności.
 3. Kartkówki.
 4. Prace domowe.
 5. Prace podejmowane z własnej inicjatywy na przykład: referaty, prezentacje, plansze poglądowe, instrukcje itp.
 6. Wykonane prace dodatkowe.
 7. Udział w konkursach, olimpiadach.

W przypadku nieobecności ucznia ma obowiązek zaliczenia sprawdzianu w terminie nieprzekraczającym dwa tygodnie od momentu przyjścia do szkoły. W przypadku stwierdzenia, że uczeń unika zajęć (wagaruje) nauczyciel może wstawić za brak zaliczenia danego działu programowego ocenę niedostateczną. Uczeń ma prawo poprawiać ocenę niedostateczną z pracy klasowej. Poprawkowy sprawdzian należy napisać w terminie jednego tygodnia od otrzymania sprawdzonej pracy. Nauczyciel może wyrazić zgodę na poprawienie oceny innej niż niedostateczna. Obszary aktywności ucznia będące przedmiotem oceny: posługiwanie się pojęciami, narzędziami oraz prawidłową terminologią informatyczną; stosowanie zasad bezpieczeństwa i właściwej organizacji pracy oraz higieny na stanowisku komputerowym; efektywna praca z poznanymi programami komputerowymi służąca osiągnięciu przewidzianych rezultatów; 3 umiejętność rozwiązywania problemów oraz dobór skutecznych metod; zastosowanie zdobytej wiedzy i umiejętności w sytuacjach praktycznych; aktywność i systematyczność.
- III. Kryteria i sposoby oceniania

Oceny ustala się w stopniach według tradycyjnej skali: Stopień celujący - 6 Stopień bardzo dobry - 5 Stopień dobry - 4 Stopień dostateczny - 3 Stopień dopuszczający - 2 Stopień niedostateczny - 1 Ustala się ogólne kryteria ocen z informatyki:

1. Ocenę celującą otrzymuje uczeń, który: Posiadał wiedzę i umiejętności znacznie wykraczające poza program nauczania;• Prowadzi samodzielną i twórczą działalność rozwijającą własne uzdolnienia;• Biegłe posługuje się zdobytymi wiadomościami w rozwiązywaniu problemów• teoretycznych lub praktycznych, proponuje rozwiązania nietypowe. Osiągnął sukcesy w konkursach i olimpiadach informatycznych•
2. Ocenę bardzo dobrą otrzymuje uczeń, który: Opanował pełen zakres wiedzy i umiejętności określonych programem nauczania;• Sprawnie komunikuje się z komputerem za pomocą systemu operacyjnego i w pełni• wykorzystuje jego możliwości; Swobodnie posługuje się oprogramowaniem użytkowym, umiejętnie dobiera je do• wykonywanych zadań; Dobrze zna pojęcia informatyczne, występujące w programie nauczania i swobodnie• je stosuje; Posiadaną wiedzę informatyczną stosuje w zadaniach praktycznych i teoretycznych;•
3. Ocenę dobrą otrzymuje uczeń, który: 4 Posiadał niepełny zakres wiedzy i umiejętności określonych programem nauczania w• danej klasie; Poprawnie stosuje nabyte wiadomości, rozwiązuje samodzielnie typowych zadania• teoretycznych i praktyczne; Poprawnie posługuje się oprogramowaniem użytkowym;• Umiejętnie korzysta z pomocy wszelakich środków masowego przekazu• Zakres jego wiadomości przekracza wymagania zawarte w podstawie programowej.• Sprawnie komunikuje się z systemem operacyjnym;•
4. Ocenę dostateczną otrzymuje uczeń, który: Opanował wiadomości i umiejętności określone programem nauczania na poziomie• nie przekraczającym wymagań zawartych w podstawach programowych; Rozwiązuje typowe zadania teoretyczne i praktyczne o średnim stopniu trudności i• przy pomocy nauczyciela; Stosuje zdobytą wiedzę do celów poznawczych i teoretycznych pod kierunkiem• nauczycieli; Umie komunikować się z komputerem za pomocą systemu operacyjnego;• Umie uruchomić omawiane oprogramowanie użytkowe;• Popelnia liczne błędy merytoryczne;
- 5. Ocenę dopuszczającą otrzymuje uczeń, który: Posiada braki w opanowaniu podstawy programowej, ale braki te nie przekreślają• możliwości uzyskania przez ucznia podstawowej wiedzy z informatyki; Rozumie pytania i polecenia;• Zna pojęcia informatyczne występujące w materiale nauczania;• Wie, czym zajmuje się informatyka i jakie programy użytkowe są omawiane;• Poprawnie uruchamia komputer i omawiane programy użytkowe;• Potrafi zastosować omawiane wiadomości do wykonania bardzo prostych czynności;• Popelnia liczne błędy merytoryczne;•
6. Ocenę niedostateczną otrzymuje uczeń, który: Nie opanował wiadomości i umiejętności określonych w podstawie programowej, a• braki te uniemożliwiają mu dalsze zdobywanie wiedzy w zakresie tego przedmiotu; Nie zna pojęć informatycznych występujących w programie nauczania;• Nie potrafi zastosować nabytych wiadomości do zadań praktycznych;• Nie rozumie poleceń i pytań;• Nie wie, czym zajmują się informatyka i nie wie, jakie są jej metody;• Nie potrafi uruchomić omawianego programu użytkowego;• Nie potrafi komunikować się z systemem operacyjnym;• W wypowiedziach popełnia liczne błędy merytoryczne;• VI. Ustalanie oceny semestralnej i końcowej a. Ocena końcowa (semestralna, roczna) nie jest średnią arytmetyczną ocen częściowych. b. Ocena semestralna lub końcowa może być podwyższona przez nauczyciela do oceny o jeden stopień wyższej w przypadku, gdy uczeń osiągał sukcesy w konkursach informatycznych. c. Uczniowie, którzy w semestrze mają ponad 50% godzin nieobecności, nie będą klasyfikowani.

VII. Nieprzygotowanie się ucznia do zajęć lekcyjnych.

Uczeń ma prawo raz w semestrze zgłosić nieprzygotowanie do lekcji, które♣ nauczyciel odnotuje w dzienniku i nie będzie ono brane pod uwagę przy klasyfikacji semestralnej czy rocznej. Zgłoszenie braku przygotowania musi się odbyć na początku lekcji, a nie w czasie jej♣ trwania. Jeżeli w danym dniu nauczyciel planuje kartkówkę, uczeń zgłaszający♣nieprzygotowanie również pisze, lecz jego praca w momencie niepowodzenia nie będzie oceniana. Prawo to nie dotyczy sprawdzianów i kartkówek zapowiadanych wcześniej.♣Osoby, które były nieobecne na lekcjach dłużej niż tydzień mają automatycznie♣ usprawiedliwione nieprzygotowanie.

VIII. Postanowienia końcowe

1. O przedmiotowych zasadach oceniania uczniowie informowani są na pierwszych zajęciach lekcyjnych.
 2. 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

- Jeśli przyjęte w szkole zasady na to pozwalają, nie trzeba wymagać od uczniów prowadzenia zeszytu (należy wówczas poprosić o gromadzenie wydruków oraz notatek wykonywanych podczas lekcji w teczce lub segregatorze). Konieczne jest natomiast systematyczne zapisywanie wykonanych w pracowni ćwiczeń w określonym miejscu 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.
- Uczeń powinien mieć możliwość zgłoszenia nieprzygotowania dwa razy w semestrze. Nieprzygotowanie powinno zostać zgłoszone przed rozpoczęciem lekcji (np. podczas sprawdzania obecności). Nie zwalnia ono ucznia z udziału w lekcji – jeśli to konieczne, uczniowi powinni podczas zajęć pomagać nauczyciel i koledzy.
- Uczeń, który był dłużej nieobecny, powinien w miarę możliwości nadrobić istotne ćwiczenia i zadania wykonane na opuszczonych lekcjach. Można określić, że jeśli np. liczba niewykonanych ćwiczeń przekroczy 20% wszystkich prac z danego działu, uczeń powinien to nadrobić.

Katalog wymagań programowych na poszczególne oceny szkolne

| Nr lekcji | Temat lekcji | Omawiane zagadnienia | Ocena | Zgodnie z wymaganiami programowymi uczeń: |
|-----------|-------------------------------|--|-------|---|
| 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; • wyjaśniaspecyfikę różnych rodzajów kaskadowych arkuszy stylów. |
| | | | 6 | • przygotowuje prezentację wyjaśniającą rolę, jaką w historii języka HTML odegrali Tim Berners-Lee, RobertCailliau,HåkonWiumLie 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 <code>&nbsp;</code> ;). |
| | | | 4 | • stosuje różne jednostki miary; • definiuje koloryróżnych 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 uczniów: |
|-----------|---------------------|--|-------|---|
| 1.3 | Strona interaktywna | Wprowadzenie do dynamicznego HTML, tworzenie elementów interaktywnych za pomocą CSS i JavaScript, budowanie galerii z wykorzystaniem elementów interaktywnych | 2 | <ul style="list-style-type: none"> • wyjaśnia pojęcie „dynamiczny HTML”. • z pomocą nauczyciela tworzy i umieszcza na stronach HTML elementy interaktywne w CSS z wykorzystaniem pseudoklasy: hover. |
| | | | 3 | <ul style="list-style-type: none"> • samodzielnie tworzy i umieszcza na stronach HTML interaktywne elementy w CSS z wykorzystaniem pseudoklasy: hover. |
| | | | 4 | <ul style="list-style-type: none"> • z pomocą nauczyciela tworzy i umieszcza na stronach HTML elementy interaktywne w JavaScript z wykorzystaniem zdarzeń onclick, onmouseover, onmouseout. |
| | | | 5 | <ul style="list-style-type: none"> • samodzielnie tworzy i umieszcza na stronach HTML elementy interaktywne w JavaScript z wykorzystaniem zdarzeń onclick, onmouseover, onmouseout; • samodzielnie tworzy interaktywną galerię fotografii. |
| | | | 6 | <ul style="list-style-type: none"> • 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 | <ul style="list-style-type: none"> • opisuje budowę adresu strony WWW; • wyjaśnia znaczenie rozszerzenia domeny. |
| | | | 3 | <ul style="list-style-type: none"> • wyjaśnia znaczenie nazwy index.htm; • tworzy odnośniki tekstowe i graficzne do innych dokumentów. |
| | | | 4 | <ul style="list-style-type: none"> • 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 | <ul style="list-style-type: none"> • samodzielnie stosuje ww. znaczniki do tworzenia poprawnej struktury dokumentu. |
| | | | 6 | <ul style="list-style-type: none"> • 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 | <ul style="list-style-type: none"> • wyjaśnia konieczność chronienia utworów (np. programów, zdjęć, stron WWW). |
| | | | 3 | <ul style="list-style-type: none"> • wyjaśnia, na czym polega naruszenie praw autorskich i jak go uniknąć. |
| | | | 4 | <ul style="list-style-type: none"> • wyjaśnia pojęcia „dozwolony użytek prywatny” i „ochrona wizerunku”. |
| | | | 5 | <ul style="list-style-type: none"> • wyjaśnia, czym są wolne oprogramowanie i cztery rodzaje wolności. |
| | | | 6 | <ul style="list-style-type: none"> • wyjaśnia praktyczne znaczenie najważniejszych punktów <i>Ustawy o prawie autorskim i prawach pokrewnych</i>. |

| Nr lekcji | Temat lekcji | Omawiane zagadnienia | Ocena | Zgodnie z wymaganiami programowymi uczniów: |
|-----------|----------------------------|---|-------|---|
| 2.1 | Pisz i powtarzaj | Instalowanie programu Python, stosowanie polecenia print i pętli for | 2 | • pisze i uruchamia prosty program wypisywania tekstu na ekranie (polecenieprint). |
| | | | 3 | • stosuje pętlę for. |
| | | | 4 | • wyjaśnia, jak działa funkcja rangew zależności od liczby parametrów. |
| | | | 5 | • rysuje szlaczki i figury, wykorzystując pętlę for,polecenieprint. |
| | | | 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 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 dzielników 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 | <ul style="list-style-type: none"> rozumie zasady gry <i>Zgadnij liczbę</i>; biorąc udział w grze, potrafi zastosować optymalną strategię. |
| | | | 3 | <ul style="list-style-type: none"> znajduje maksymalną liczbę kroków odgadywania danej liczby. |
| | | | 4 | <ul style="list-style-type: none"> losuje liczby całkowite z danego zakresu; wykorzystuje pętlę while do znajdowania sumy cyfr liczby. |
| | | | 5 | <ul style="list-style-type: none"> analizuje schemat blokowy algorytmu obliczania sumy cyfr dowolnej liczby; samodzielnie implementuje grę <i>Zgadnij liczbę</i> w Pythonie, korzystając ze wskazówek w podręczniku. |
| | | | 6 | <ul style="list-style-type: none"> samodzielnie rozwiązuje dodatkowe zadania programistyczne. |
| 2.6 | Zrób porządek | Porządkowanie elementów zbioru przez prosty wybór i zliczanie, wykorzystywanie list | 2 | <ul style="list-style-type: none"> opisuje porządkowanie zbioru przez proste wybieranie i zliczanie. |
| | | | 3 | <ul style="list-style-type: none"> opisuje, czym jest lista, i potrafi z niej korzystać. |
| | | | 4 | <ul style="list-style-type: none"> korzysta z funkcji związanych z listami. |
| | | | 5 | <ul style="list-style-type: none"> definiuje funkcje zliczania. |
| | | | 6 | <ul style="list-style-type: none"> samodzielnie rozwiązuje dodatkowe zadania programistyczne. |
| 3.1 | Jak to z Gaussem było | Sumowanie w arkuszu kalkulacyjnym, porządkowanie danych w tabelach, analizowanie danych zapisanych w arkuszu, obliczeń i prawdziwości | 2 | <ul style="list-style-type: none"> korzysta z arkusza kalkulacyjnego w podstawowym zakresie. |
| | | | 3 | <ul style="list-style-type: none"> 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 | <ul style="list-style-type: none"> rozwiązuje w arkuszu proste zadania matematyczne. |
| | | | 5 | <ul style="list-style-type: none"> planuje wykonywanie obliczeń w arkuszu; analizuje dane zawarte w arkuszu w poszukiwaniu prawdziwości. |
| | | | 6 | <ul style="list-style-type: none"> samodzielnie formułuje wnioski. |

| Nr lekcji | Temat lekcji | Omawiane zagadnienia | Ocena | Zgodnie z wymaganiami programowymi uczniów: |
|-----------|---------------------------------------|--|-------|---|
| 3.2 | Liczby, potęgi, ciągi | Wprowadzanie serii danych, formuł i funkcji do arkusza kalkulacyjnego, porównywanie ciągów liczbowych, włączanie ochrony arkusza | 2 | <ul style="list-style-type: none"> rozumie, czym jest formuła i format liczbowy, i używa ich w zadaniu; drukuje tabele przygotowane w arkuszu. |
| | | | 3 | <ul style="list-style-type: none"> wprowadza do arkusza serie danych formuły i funkcje; odróżnia i wprowadza różne formaty liczbowe. |
| | | | 4 | <ul style="list-style-type: none"> planuje wykonywanie obliczeń w arkuszu; porównuje ciągi liczbowe i odnajduje występujące w nich prawidłowości. |
| | | | 5 | <ul style="list-style-type: none"> analizuje dane zawarte w arkuszu; tworzy prosty kalkulator matematyczny; uniemożliwia zmianę danych w arkuszu (włącza ochronę arkusza). |
| | | | 6 | <ul style="list-style-type: none"> samodzielnie formułuje wnioski. |
| 3.3 | Z tabeli – wykres | Rysowanie wykresów funkcji za pomocą kreatora wykresów arkusza kalkulacyjnego, wstawianie i formatowanie wykresu punktowego | 2 | <ul style="list-style-type: none"> rozumie, czym jest wykres, i drukuje go wraz z tabelą danych. |
| | | | 3 | <ul style="list-style-type: none"> przygotowuje dane do wykonania wykresu funkcji liniowej. |
| | | | 4 | <ul style="list-style-type: none"> tworzy wykresy funkcji liniowych za pomocą kreatora wykresów. |
| | | | 5 | <ul style="list-style-type: none"> opisuje i formatuje elementy wykresu. |
| 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 | <ul style="list-style-type: none"> rozumie, czym jest funkcja, i z pomocą nauczyciela korzysta z kreatora funkcji. |
| | | | 3 | <ul style="list-style-type: none"> przegląda, sortuje i filtruje w arkuszu duże zestawy danych. |
| | | | 4 | <ul style="list-style-type: none"> samodzielnie korzysta z funkcji statystycznych LICZ, JEŻELI i CZĘSTOŚĆ. |
| | | | 5 | <ul style="list-style-type: none"> tworzy tabelę przestawną. |
| | | | 6 | <ul style="list-style-type: none"> samodzielnie formułuje wnioski. |

| Nr lekcji | Temat lekcji | Omawiane zagadnienia | Ocena | Zgodnie z wymaganiami programowymi uczniów: |
|-----------|-----------------------|--|-------|---|
| 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; |
| | | | 4 | • korzysta z funkcji statystycznych ŚREDNIA, MIN, MAX i MEDIANA. |
| | | | 5 | • omawia specyfikę przetwarzania rozproszonego i opisuje wybrane projekty. |
| | | | 6 | • tworzy wykres zależności XY i wstawia linię trendu. |
| 3.6 | Moi znajomi | Wprowadzenie do pracy z kartotekową bazą danych – przygotowanie, filtrowanie, uzupełnianie, poprawianie i sortowanie danych, zastosowanie formularza do wpisywania danych | 2 | • samodzielnie formułuje wnioski. |
| | | | 3 | • wyjaśnia, czym jest kartotekowa baza danych. |
| | | | 4 | • wpisuje dane do arkusza udostępnionego do edycji w chmurze. |
| | | | 5 | • sortuje i filtruje dane; |
| | | | 6 | • sprawnie wyszukuje dane o wybranych kryteriach. |
| | | | | • tworzy formularz w celu dopisywania lub poprawiania rekordów. |
| | | | | • tworzy formularz w celu dopisywania lub poprawiania rekordów. |
| | | | | • rozbudowuje bazę danych; |
| | | | | • oblicza wystąpienia pewnych danych za pomocą wbudowanych funkcji. |
| 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; |
| | | | 3 | • drukuje wykresy obrazujące wyniki doświadczenia. |
| | | | 4 | • korzysta z funkcji losowych w arkuszu; |
| | | | 5 | • trafnie ocenia wynik prostego doświadczenia losowego. |
| | | | 6 | • przeprowadza zadaną symulację prostego doświadczenia z użyciem funkcji losującej; |
| | | | | • wykonuje wykres wyników doświadczenia. |
| | | | | • samodzielnie planuje i przeprowadza symulację procesu o losowym przebiegu. |
| | | | | • samodzielnie planuje obliczenia i formułuje wnioski; |
| | | | | • proponuje doświadczenie losowe i z góry 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 płatką Kocha i trójkąta Sierpińskiego w środowisku App Lab | 2 | • otwiera i analizuje projekty w Scratchu. |
| | | | 3 | • opisuje algorytmy tworzenia trójkąta Sierpińskiego i płatką 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 <i>Gry w życie</i> na wybranej stronie internetowej. |
| | | | 3 | • opisuje zasady <i>Gry w życie</i> . |
| | | | 4 | • eksperymentuje i obserwuje etapy życia na planecie. |
| | | | 5 | • znajduje układy, w których populacja zachowuje się w określony sposób. |
| | | | 6 | • realizuje własną symulację <i>Gry w życie</i> w wybranym języku programowania. |

| Nr lekcji | Temat lekcji | Omawiane zagadnienia | Ocena | Zgodnie z wymaganiami programowymi uczniów: |
|-----------|---------------------------|---|-------|---|
| 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. |
| 5.1 | Mały robot – Android | Omówienie narzędzi i aplikacji użytkowych wbudowanych w system Android oraz zewnętrznych, instalacja i obsługa TinyScanner – PDF ScannerApp | 2 | • charakteryzuje podstawowe narzędzia systemu Android. |
| 5.2 | Ze smartfonem na piechotę | Planowanie i dokumentowanie wycieczki z wykorzystaniem urządzenia mobilnego, publikowanie trasy wycieczki w internecie | 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. |
| | | | 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 uczniów: |
|-----------|--|--|-------|---|
| 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; |
| | | | 4 | • odróżni rozszerzoną rzeczywistość od rzeczywistości wirtualnej. |
| | | | 5 | • podaje przykłady wykorzystania technologii AR. |
| | | | 6 | • podaje przykłady sytuacji, w których zastosowanie technologii AR byłoby przydatne. |
| 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; |
| | | | 3 | • wyjaśnia pojęcie „MOOC”. |
| | | | 4 | • znajduje serwisy oferujące MOOC; |
| | | | 5 | • krótko charakteryzuje kursy informatyczne w Akademii Khana. |
| | | | 6 | • znajduje kursy w serwisach oferujących MOOC; |
| 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 | • korzysta z kursów informatycznych w Akademii Khana. |
| | | | 3 | • potrafi zanalizować przydatność kursów w serwisach oferujących MOOC. |
| | | | 4 | • samodzielnie wykonuje ćwiczenia w ramach kursów informatycznych w Akademii Khana. |
| | | | 5 | • w podstawowym zakresie korzysta ze wskazanych aplikacji i serwisów. |
| | | | 6 | • w pełnym zakresie korzysta ze wskazanych aplikacji i serwisów. |
| | | | 2 | • korzysta z samodzielnie znalezionych aplikacji i serwisów wspomagających naukę i rozwijających zainteresowania. |
| | | | 3 | • buduje własną bazę wiedzy. |
| | | | 4 | • prezentuje w klasie wyszukane aplikacje i serwisy wspomagające naukę i rozwijające zainteresowania i poddaje je krytycznej ocenie pod kątem użyteczności oraz przydatności. |
| | | | 5 | |
| | | | 6 | |

