

Program nauczania

INFORMATYKA **dla szkół ponadgimnazjalnych.** **Zakres rozszerzony**

Grażyna Koba

Spis treści

1. Założenia programu.....	3
2. Podstawa programowa – informatyka – zakres rozszerzony.....	4
3. Szczegółowe cele kształcenia i wychowania oraz treści nauczania w korelacji z treściami podstawy programowej.....	9
3.1. Podstawy algorytmiki i programowania.....	9
3.2. Komputer i sieci komputerowe.....	14
3.3. Opracowywanie informacji za pomocą komputera.....	19
3.4. Bazy danych.....	22
3.5. Algorytmika i programowanie – problemy zaawansowane.....	26
3.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci.....	31
4. Sposoby osiągania celów kształcenia i wychowania.....	32
4.1. Podstawy algorytmiki i programowania.....	32
4.2. Komputer i sieci komputerowe.....	36
4.3. Opracowywanie informacji za pomocą komputera.....	40
4.4. Bazy danych.....	43
4.5. Algorytmika i programowanie – problemy zaawansowane.....	45
4.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci.....	48
5. Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne.....	49
5.1. Podstawy algorytmiki i programowania.....	49
5.2. Komputer i sieci komputerowe.....	53
5.3. Opracowywanie informacji za pomocą komputera.....	56
5.4. Bazy danych.....	60
5.5. Algorytmika i programowanie – problemy zaawansowane.....	63
5.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci.....	66
6. Propozycje metod sprawdzania osiągnięć ucznia.....	67

1. Założenia programu

*Wiedzieć, że wiemy, co wiemy,
i że nie wiemy tego, czego nie wiemy
– oto prawdziwa wiedza.
Konfucjusz*

W programie ujęte są w pełni cele i treści nauczania zawarte w podstawie programowej przedmiotu informatyka, realizowanego w zakresie rozszerzonym. Realizacja programu pozwala na prawidłowe przekazanie i skuteczne utrwalenie wiedzy i umiejętności w całym cyklu nauczania.

Program został przygotowany tak, aby jak najlepiej wspierać nauczycieli w nauczaniu informatyki w zakresie rozszerzonym. Informatyka jest dziedziną rozwijającą się tak szybko, że nie jest możliwe przekazanie uczniom wszystkich informacji z wszystkich obszarów jej zastosowania. Obszary te są wskazane w programie, ale pełną wiedzę na ich temat uczniowie zdobędą na studiach wyższych.

Jednym z celów podstawy programowej jest „przygotowanie do świadomego wyboru kierunku i zakresu dalszego kształcenia informatycznego”. Na przykład uczeń poznaje algorytmikę i programowanie, aby był przygotowany do dalszego kształcenia kierunkowego, np. informatycznego.

Zagadnienia programowe na tym etapie powinny inspirować uczniów i zachęcić do samodzielnego zgłębiania interesującej ich wiedzy i rozwijania umiejętności, które będą wykorzystywać na studiach wyższych lub w pracy zawodowej.

Podchodząc pragmatycznie do nauczania, zakładam realną realizację prezentowanego programu w każdych warunkach szkolnych. Dążąc do wszechstronnego rozwoju ucznia, należy pamiętać, że kształcenie i wychowanie powinny być ze sobą zharmonizowane. Powinny być one również konsekwentną kontynuacją założeń edukacyjnych realizowanych w poprzednich etapach edukacyjnych. Każdy nauczyciel powinien zapoznać się z nimi, zanim rozpocznie planowanie zajęć. W przypadku informatyki realizowanej w zakresie rozszerzonym należy również poznać szczegółowe założenia programowe informatyki realizowanej w zakresie podstawowym.

Wyjątkowo istotnym założeniem edukacyjnym jest odpowiednie przygotowanie uczniów do podjęcia studiów i pracy zawodowej. Realizując cele kształcenia i wychowania, wskazujemy, jak w praktyce wykorzystać wiedzę ogólną i nabyte umiejętności oraz jak w poszukiwaniu nowej wiedzy pogodzić zasady etyczne z otworzeniem się na świat.

Uczniowie poznają informatykę jako dziedzinę wiedzy (jest to szczególnie ważne dla tych, którzy zamierzają kontynuować naukę na studiach informatycznych). Uczą się algorytmiki i programowania. Szczególną uwagę zwracamy na podstawowe zasady programowania,

posługując się wybranymi językami programowania (w przypadku podstawowych zagadnień nauczyciel może wybrać język Pascal lub C++). Takie podejście jest uzasadnione metodycznie. Jeśli uczeń zrozumie metody programowania na przykładzie jednego języka, to bez większych trudności będzie je potrafił zastosować także w innych językach.

Uczniowie powinni sprawnie korzystać z komputera i sieci komputerowych. Dlatego w programie zakładamy poznawanie schematu logicznego działania komputera i działania sieci.

Pogłębione w stosunku do informatyki podstawowej zostały zagadnienia dotyczące grafiki komputerowej rastrowej i wektorowej, przygotowywania prezentacji multimedialnych, edycji tekstu oraz wykorzystania arkusza kalkulacyjnego do obrazowania zależności funkcyjnych i do zapisywania algorytmów.

Poszerzone zostały wiadomości dotyczące przetwarzania danych w bazach danych, w tym sieciowych, w zakresie samodzielnego przygotowania relacyjnej bazy danych i wyszukiwania informacji z użyciem języka zapytań. Na przykładzie projektowania systemów informatycznych uczniowie mogą dowiedzieć się, na czym polegają: podejmowanie indywidualnych i grupowych decyzji, skuteczne działanie w grupie, umiejętne planowanie pracy oraz nadzór nad wykonywanym projektem.

Cele edukacyjne i treści nauczania (punkt 3.) oraz procedury osiągania celów edukacyjnych (punkt 4.) przygotowane zostały w przejrzystej, tabelarycznej formie. Szczegółowe treści nauczania skorelowano z odpowiednimi punktami podstawy programowej. Zakładane osiągnięcia ucznia są przedstawione w postaci szczegółowych wymagań na poszczególne oceny szkolne – od 2 do 6 (punkt 5.).

Uwaga:

Podstawową wiedzę i umiejętności uczeń zdobywa na informatyce realizowanej w zakresie podstawowym. Program informatyki w zakresie rozszerzonym nie ujmuje więc zagadnień z poszczególnych działów tematycznych wynikających z podstawy programowej informatyki realizowanej w zakresie podstawowym, a tylko je rozszerza i uzupełnia.

2. Podstawa programowa – informatyka – zakres rozszerzony

W rozporządzeniu Ministra Edukacji Narodowej z dnia 27 sierpnia 2012 r. w sprawie podstawy programowej wychowania przedszkolnego oraz kształcenia ogólnego w poszczególnych typach szkół określono m.in. podstawę programową do informatyki w zakresie rozszerzonym dla IV etapu edukacyjnego.

Cele kształcenia – wymagania ogólne

- I. Bezpieczne posługiwanie się komputerem i jego oprogramowaniem, wykorzystanie sieci komputerowej; komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych.

- II. Wyszukiwanie, gromadzenie i przetwarzanie informacji z różnych źródeł; opracowywanie za pomocą komputera: rysunków, tekstów, danych liczbowych, motywów, animacji, prezentacji multimedialnych.
- III. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.
- IV. Wykorzystanie komputera oraz programów i gier edukacyjnych do poszerzania wiedzy i umiejętności z różnych dziedzin oraz do rozwijania zainteresowań.
- V. Ocena zagrożeń i ograniczeń, docenianie społecznych aspektów rozwoju i zastosowań informatyki.

Treści nauczania - wymagania szczegółowe

1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. Uczeń:
 - 1) przedstawia sposoby reprezentowania różnych form informacji w komputerze: liczb, znaków, obrazów, animacji, dźwięków;
 - 2) wyjaśnia funkcje systemu operacyjnego i korzysta z nich; opisuje różne systemy operacyjne;
 - 3) przedstawia warstwowy model sieci komputerowych, określa ustawienia sieciowe danego komputera i jego lokalizacji w sieci, opisuje zasady administrowania siecią komputerową w architekturze klient-serwer, prawidłowo posługuje się terminologią sieciową, korzysta z usług w sieci komputerowej, lokalnej i globalnej, związanych z dostępem do informacji, wymianą informacji i komunikacją;
 - 4) zapoznaje się z możliwościami nowych urządzeń związanych z technologiami informacyjno-komunikacyjnymi, poznaje nowe programy i systemy oprogramowania.
2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:
 - 1) projektuje relacyjną bazę danych z zapewnieniem integralności danych;
 - 2) stosuje metody wyszukiwania i przetwarzania informacji w relacyjnej bazie danych (język SQL);
 - 3) tworzy aplikację bazodanową, w tym sieciową, wykorzystującą język zapytań, kwerendy, raporty; zapewnia integralność danych na poziomie pól, tabel, relacji;
 - 4) znajduje odpowiednie informacje niezbędne do realizacji projektów z różnych dziedzin;
 - 5) opisuje mechanizmy związane z bezpieczeństwem danych: szyfrowanie, klucz, certyfikat, zaporę ogniową.
3. Komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych. Uczeń:
 - 1) wykorzystuje zasoby i usługi sieci komputerowych w komunikacji z innymi użytkownikami, w tym do przesyłania i udostępniania danych;
 - 2) bierze udział w dyskusjach w sieci (forum internetowe, czat).

4. Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów. Uczeń:
- 1) opisuje podstawowe modele barw i ich zastosowanie;
 - 2) określa własności grafiki rastrowej i wektorowej oraz charakteryzuje podstawowe formaty plików graficznych, tworzy i edytuje obrazy rastrowe i wektorowe z uwzględnieniem warstw i przekształceń;
 - 3) przetwarza obrazy i filmy, np.: zmienia rozdzielczość, rozmiar, model barw, stosuje filtry;
 - 4) wykorzystuje arkusz kalkulacyjny do obrazowania zależności funkcyjnych i do zapisywania algorytmów.
5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:
- 1) analizuje, modeluje i rozwiązuje sytuacje problemowe z różnych dziedzin;
 - 2) stosuje podejście algorytmiczne do rozwiązywania problemu;
 - 3) formułuje przykłady sytuacji problemowych, których rozwiązanie wymaga podejścia algorytmicznego i użycia komputera;
 - 4) dobiera efektywny algorytm do rozwiązania sytuacji problemowej i zapisuje go w wybranej notacji;
 - 5) posługuje się podstawowymi technikami algorytmicznymi;
 - 6) ocenia własności rozwiązania algorytmicznego (komputerowego), np. zgodność ze specyfikacją, efektywność działania;
 - 7) opracowuje i przeprowadza wszystkie etapy prowadzące do otrzymania poprawnego rozwiązania problemu: od sformułowania specyfikacji problemu po testowanie rozwiązania;
 - 8) posługuje się metodą „dziel i zwyciężaj” w rozwiązywaniu problemów;
 - 9) stosuje rekurencję w prostych sytuacjach problemowych;
 - 10) stosuje podejście zachłanne w rozwiązywaniu problemów;
 - 11) opisuje podstawowe algorytmy i stosuje:
 - a) algorytmy na liczbach całkowitych, np.:
 - reprezentacja liczb w dowolnym systemie pozycyjnym, w tym w dwójkowym i szesnastkowym,
 - sprawdzanie, czy liczba jest liczbą pierwszą, doskonałą,
 - rozkładanie liczby na czynniki pierwsze,
 - iteracyjna i rekurencyjna realizacja algorytmu Euklidesa,
 - iteracyjne i rekurencyjne obliczanie wartości liczb Fibonacciego,
 - wydawanie reszty metodą zachłanną,
 - b) algorytmy wyszukiwania i porządkowania (sortowania), np.:
 - jednoczesne znajdowanie największego i najmniejszego elementu w zbiorze: algorytm naiwny i optymalny,
 - algorytmy sortowania ciągu liczb: bąbelkowy, przez wybór, przez wstawianie liniowe lub binarne, przez scalanie, szybki, kubelkowy,

- c) algorytmy numeryczne, np.:
 - obliczanie wartości pierwiastka kwadratowego,
 - obliczanie wartości wielomianu za pomocą schematu Hornera,
 - zastosowania schematu Hornera: reprezentacja liczb w różnych systemach liczbowych, szybkie podnoszenie do potęgi,
 - wyznaczanie miejsc zerowych funkcji metodą połowienia,
 - obliczanie pola obszarów zamkniętych,
 - d) algorytmy na tekstach, np.:
 - sprawdzanie, czy dany ciąg znaków tworzy palindrom, anagram,
 - porządkowanie alfabetyczne,
 - wyszukiwanie wzorca w tekście,
 - obliczanie wartości wyrażenia podanego w postaci odwrotnej notacji polskiej,
 - e) algorytmy kompresji i szyfrowania, np.:
 - kody znaków o zmiennej długości, np. alfabet Morse'a, kod Huffmana,
 - szyfr Cezara,
 - szyfr przestawieniowy,
 - szyfr z kluczem jawnym (RSA),
 - wykorzystanie algorytmów szyfrowania, np. w podpisie elektronicznym,
 - f) algorytmy badające własności geometryczne, np.:
 - sprawdzanie warunku trójkąta,
 - badanie położenia punktów względem prostej,
 - badanie przynależności punktu do odcinka,
 - przecinanie się odcinków,
 - przynależność punktu do obszaru,
 - konstrukcje rekurencyjne: drzewo binarne, dywan Sierpińskiego, płatek Kocha;
- 12) projektuje rozwiązanie problemu (realizację algorytmu) i dobiera odpowiednią strukturę danych;
- 13) stosuje metodę zstępującą i wstępującą przy rozwiązywaniu problemu;
- 14) dobiera odpowiednie struktury danych do realizacji algorytmu, w tym struktury dynamiczne;
- 15) stosuje zasady programowania strukturalnego i modularnego do rozwiązywania problemu;
- 16) opisuje własności algorytmów na podstawie ich analizy;
- 17) ocenia zgodność algorytmu ze specyfikacją problemu;
- 18) oblicza liczbę operacji wykonywanych przez algorytm;

- 19) szacuje wielkość pamięci potrzebnej do komputerowej realizacji algorytmu;
 - 20) bada efektywność komputerowych rozwiązań problemów;
 - 21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;
 - 22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;
 - 23) stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu;
 - 24) dobiera najlepszy algorytm, odpowiednie struktury danych i oprogramowanie do rozwiązania postawionego problemu;
 - 25) dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;
 - 26) ocenia poprawność komputerowego rozwiązania problemu na podstawie jego testowania;
 - 27) wyjaśnia źródło błędów w obliczeniach komputerowych (błąd względny, błąd bezwzględny);
 - 28) realizuje indywidualnie lub zespołowo projekt programistyczny z wydzieleniem jego modułów, w ramach pracy zespołowej dokumentuje pracę zespołu.
6. Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:
 - 1) opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;
 - 2) korzysta z zasobów edukacyjnych udostępnianych na portalach przeznaczonych do kształcenia na odległość.
 7. Uczeń wykorzystuje komputer i technologie informacyjno-komunikacyjne do rozwijania swoich zainteresowań, opisuje zastosowania informatyki, ocenia zagrożenia i ograniczenia, docenia aspekty społeczne rozwoju i zastosowań informatyki:
 - 1) opisuje najważniejsze elementy procesu rozwoju informatyki i technologii informacyjno-komunikacyjnych;
 - 2) wyjaśnia szanse i zagrożenia dla rozwoju społecznego i gospodarczego oraz dla obywateli, związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych;
 - 3) stosuje normy etyczne i prawne związane z rozpowszechnianiem programów komputerowych, bezpieczeństwem i ochroną danych oraz informacji w komputerze i w sieciach komputerowych;
 - 4) omawia zagadnienia przestępczości komputerowej, w tym piractwo komputerowe, nielegalne transakcje w sieci;
 - 5) przygotowuje się do świadomego wyboru kierunku i zakresu dalszego kształcenia informatycznego.

3. Szczegółowe cele kształcenia i wychowania oraz treści nauczania w korelacji z treściami podstawy programowej

3.1. Podstawy algorytmiki i programowania

Szczegółowe cele wychowawcze:

- Przestrzeganie zasad bezpiecznej pracy przy komputerze.
- Odpowiedzialność za ogólny porządek na stanowisku komputerowym.
- Szanowanie pracy innych. Przestrzeganie zasad ochrony zasobów komputera.
- Rozwijanie dociekliwości poznawczej ukierunkowanej na rzetelne zdobywanie wiedzy.
- Uczenie się precyzyjnego wyrażania myśli.
- Dostrzeganie mechanizmów logicznych obecnych w otaczającym świecie.
- Odpowiedzialność za podejmowane działania.
- Wykorzystywanie zasad projektowania algorytmów i programowania w życiu codziennym.

Prezentacja algorytmu liniowego w wybranej notacji		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Posługiwanie się algorytmami w rozwiązywaniu zadań szkolnych i problemów życia codziennego.</p> <p>Poznanie różnych sposobów prezentacji algorytmów.</p> <p>Przestrzeganie zasad zapisu algorytmów w wybranej postaci (notacji).</p> <p>Stosowanie poznanych metod prezentacji algorytmów do opisywania zadań z innych przedmiotów szkolnych oraz sytuacji z różnych dziedzin życia.</p>	<p>Rozwiązywanie problemów algorytmicznych.</p> <p>Formułowanie sytuacji problemowej.</p> <p>Określenie specyfikacji problemu (zadania).</p> <p>Usystematyzowanie pojęć: <i>algorytm, algorytm liniowy, specyfikacja zadania, lista kroków, schemat blokowy, program komputerowy, język programowania, program źródłowy i wynikowy, zmienna.</i></p> <p>Przedstawianie algorytmu liniowego w postaci listy kroków i schematu blokowego.</p> <p>Zapisywanie algorytmów liniowych w języku programowania.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>1) analizuje, modeluje i rozwiązuje sytuacje problemowe z różnych dziedzin;</i></p> <p><i>2) stosuje podejście algorytmiczne do rozwiązywania problemu;</i></p> <p><i>3) formułuje przykłady sytuacji problemowych, których rozwiązanie wymaga podejścia algorytmicznego i użycia komputera;</i></p> <p><i>17) ocenia zgodność algorytmu ze specyfikacją problemu;</i></p> <p><i>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</i></p>

Podstawowe zasady programowania		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Rozumienie, na czym polega programowanie.</p> <p>Poznanie podstawowych zasad tworzenia programu komputerowego oraz stosowanie podstawowych instrukcji języka programowania.</p> <p>Rozumienie, czym jest zmienna w programie komputerowym (m.in. co oznacza przypisanie zmiennej konkretnej wartości).</p> <p>Poznanie zastosowań wybranych typów danych i właściwego doboru struktur danych do algorytmu.</p> <p>Rozumienie konsekwencji deklaracji zmiennych określonego typu (zajętość pamięci komputera, minimalne i maksymalne wartości zmiennej danego typu).</p> <p>Podstawowe metodologie programowania: programowanie zstępujące (ang. <i>top-down</i>) i wstępujące (ang. <i>bottom-up</i>).</p>	<p>Zasady programowania. Struktura programu w wybranym języku programowania.</p> <p>Pojęcie zmiennej. Nazwa zmiennej. Zmienne lokalne i globalne. Deklaracja zmiennych. Typy zmiennych. Klasyfikacja typów danych. Przykłady typów prostych i złożonych.</p> <p>Wprowadzanie danych i wyprowadzanie wyników. Podstawowe polecenia: instrukcja przypisania, instrukcja warunkowa, instrukcja wyboru, instrukcje pętli. Operatory arytmetyczne i logiczne.</p> <p>Modele programowania (programowanie liniowe, strukturalne, modułowe, obiektowe, zdarzeniowe).</p> <p>Metoda zstępująca i wstępująca.</p> <p>Deklaracja procedur i funkcji. Sposoby przekazywania parametrów – przez wartość, przez zmienną (przez referencję). Zasięg zmiennej. Stosowanie procedur i funkcji w zadaniach.</p> <p>Realizacja algorytmów iteracyjnych i rekurencyjnych w postaci programu.</p> <p>Metody reprezentowania danych złożonych – tablice. Deklaracja tablicy. Wczytywanie i wyprowadzanie elementów tablicy.</p> <p>Metody reprezentowania ciągu znaków – typ łańcuchowy. Deklarowanie łańcuchów. Wprowadzanie danych tekstowych.</p> <p>Metody doboru struktur danych do algorytmu. Szacowanie wielkości pamięci zajmowanej przez dane określonego typu.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>12) projektuje rozwiązanie problemu (realizację algorytmu) i dobiera odpowiednią strukturę danych;</i></p> <p><i>13) stosuje metodę zstępującą i wstępującą przy rozwiązywaniu problemu;</i></p> <p><i>14) dobiera odpowiednie struktury danych do realizacji algorytmu [...];</i></p> <p><i>15) stosuje zasady programowania strukturalnego [...] do rozwiązywania problemu;</i></p> <p><i>17) ocenia zgodność algorytmu ze specyfikacją problemu;</i></p> <p><i>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</i></p> <p><i>22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;</i></p> <p><i>23) stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu;</i></p> <p><i>24) dobiera najlepszy algorytm, odpowiednie struktury danych i oprogramowanie do rozwiązania postawionego problemu;</i></p> <p><i>26) ocenia poprawność komputerowego rozwiązania problemu na podstawie jego testowania;</i></p>

Uwaga: Podstawowe zasady programowania proponuję omawiać na przykładzie języków: Pascal lub C++.

Techniki algorytmiczne i wybrane algorytmy		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Rozumienie, czym charakteryzuje się algorytm z warunkami i czym różni się od algorytmu liniowego.</p> <p>Rozumienie, na czym polega iteracja i w jaki sposób określa się warunek zakończenia iteracji.</p> <p>Poznanie, czym jest rekurencja oraz czym różni się od iteracji.</p> <p>Rozumienie, na czym polega sortowanie elementów. Stosowanie techniki iteracji w algorytmach sortowania.</p> <p>Stosowanie metody zachłannej w algorytmie wydawania reszty.</p> <p>Stosowanie technik algorytmicznych (sortowania, iteracji i rekurencji) do rozwiązywania zadań szkolnych i problemów życia codziennego.</p>	<p>Algorytmy z warunkami: prosta sytuacja warunkowa, warunki zagnieżdżone.</p> <p>Iteracja. Zmienna sterująca pętli. Określanie kroku iteracji. Pętla prosta i zagnieżdżona. Sposoby zakończenia iteracji: liczba powtórzeń z góry określona lub zależna od warunku.</p> <p>Rekurencja. Pojęcie stosu. Zapis algorytmu rekurencyjnego.</p> <p>Zasady zapisu algorytmów z warunkami, iteracyjnych i rekurencyjnych w postaci listy kroków, schematu blokowego, w języku programowania.</p> <p>Ocenianie własności rozwiązania: zgodność ze specyfikacją, testowanie rozwiązania dla wybranych danych.</p>	<p>5. <i>Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p>2) <i>stosuje podejście algorytmiczne do rozwiązywania problemu;</i></p> <p>4) <i>dobiera efektywny algorytm do rozwiązania sytuacji problemowej i zapisuje go w wybranej notacji;</i></p> <p>5) <i>posługuje się podstawowymi technikami algorytmicznymi;</i></p> <p>6) <i>ocenia własności rozwiązania algorytmicznego (komputerowego), np. zgodność ze specyfikacją, efektywność działania;</i></p> <p>7) <i>opracowuje i przeprowadza wszystkie etapy prowadzące do otrzymania poprawnego rozwiązania problemu: od sformułowania specyfikacji problemu po</i></p>

<p>Poznanie wybranych algorytmów i stosowanie ich w typowych sytuacjach problemowych.</p> <p>Stosowanie technik iteracji i rekurencji w wybranych algorytmach.</p> <p>Wykorzystanie algorytmów szukania minimum i maksimum w sortowaniu.</p>	<p>Przykłady algorytmów z rozgałęzieniami: sprawdzanie, czy liczba jest parzysta, sprawdzanie warunku istnienia trójkąta.</p> <p>Przykłady algorytmów: dodawanie n liczb, mnożenie n liczb, silnia, algorytm Euklidesa (NWD) – wersja z odejmowaniem i dzieleniem, NWW (najmniejsza wspólna wielokrotność), szukanie minimum lub maksimum wśród danych elementów; jednocześnie znajdowanie największego i najmniejszego elementu – algorytm naiwny i optymalny (metoda „dziel i zwyciężaj”), generowanie liczb Fibonacciego, schemat Hornera.</p> <p>Iteracyjna i rekurencyjna realizacja algorytmów: obliczania silni, Euklidesa, obliczania wartości liczb Fibonacciego i wartości wielomianu za pomocą schematu Hornera.</p> <p>Algorytm wydawania reszty metodą zachłanną.</p> <p>Przykłady algorytmów sortowania: przez wybór, bąbelkowe, pozycyjne.</p> <p>Algorytmy na tekstach: tworzenie anagramów, wyszukiwanie i zliczanie znaków w tekście, sprawdzanie, czy dany ciąg znaków tworzy palindrom.</p>	<p><i>testowanie rozwiązania;</i></p> <p><i>8) posługuje się metodą "dziel i zwyciężaj" w rozwiązywaniu problemów;</i></p> <p><i>9) stosuje rekurencję w prostych sytuacjach problemowych;</i></p> <p><i>10) stosuje podejście zachłanne w rozwiązywaniu problemów;</i></p> <p><i>11) opisuje podstawowe algorytmy i stosuje:</i></p> <p><i>a) algorytmy na liczbach całkowitych – iteracyjna i rekurencyjna realizacja algorytmu Euklidesa, iteracyjne i rekurencyjne obliczanie wartości liczb Fibonacciego, wydawanie reszty metodą zachłanną;</i></p> <p><i>b) algorytmy wyszukiwania i porządkowania (sortowania) – jednocześnie znajdowanie największego i najmniejszego elementu w zbiorze: algorytm naiwny i optymalny, algorytmy sortowania ciągu liczb (bąbelkowy, przez wybór);</i></p> <p><i>c) algorytmy numeryczne – obliczanie wartości wielomianu za pomocą schematu Hornera;</i></p> <p><i>g) algorytmy na tekstach – sprawdzanie, czy dany ciąg znaków tworzy palindrom, anagram;</i></p> <p><i>f) algorytmy badające własności geometryczne – sprawdzanie warunku trójkąta;</i></p> <p><i>17) ocenia zgodność algorytmu ze specyfikacją problemu;</i></p> <p><i>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</i></p>
--	---	--

Elementy analizy algorytmów		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Poznanie najważniejszych własności algorytmów.</p> <p>Ocenianie poprawności działania i skończoności algorytmów.</p> <p>Ocenianie złożoności obliczeniowej algorytmów – czasowej i pamięciowej.</p> <p>Rozumienie, na czym polega efektywność algorytmów.</p>	<p>Analiza działania algorytmu (zapisanego w wybranej postaci) dla przykładowych danych.</p> <p>Poprawność algorytmu – określanie, czy algorytm działa zgodnie ze specyfikacją zadania.</p> <p>Określenie algorytmu uniwersalnego.</p> <p>Przykład określania skończoności algorytmu.</p> <p>Złożoność obliczeniowa algorytmu – czasowa (określanie liczby operacji wykonywanych na danych, np. określanie liczby porównań elementów w algorytmie sortowania) i pamięciowa (wielkość pamięci niezbędnej do wykonania algorytmu).</p> <p>Porównywanie złożoności różnych algorytmów rozwiązywania tego samego zadania dla tych samych danych, np. porównanie dwóch algorytmów sortowania.</p> <p>Określanie efektywności algorytmów.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>16) opisuje własności algorytmów na podstawie ich analizy;</i></p> <p><i>17) ocenia zgodność algorytmu ze specyfikacją problemu;</i></p> <p><i>18) oblicza liczbę operacji wykonywanych przez algorytm;</i></p> <p><i>19) szacuje wielkość pamięci potrzebnej do komputerowej realizacji algorytmu;</i></p> <p><i>20) bada efektywność komputerowych rozwiązań problemów;</i></p>

3.2. Komputer i sieci komputerowe

Szczegółowe cele wychowawcze:

- Szanowanie pracy innych. Przestrzeganie zasad ochrony zasobów komputera.
- Rozwijanie dociekliwości poznawczej ukierunkowanej na rzetelną informację.
- Świadome korzystanie z sieci komputerowej i Internetu.
- Rozumienie zagrożeń wynikających z niewłaściwego wyboru źródła informacji i samej informacji.
- Rozumienie znaczenia dostępności do Internetu dla własnego rozwoju w różnych dziedzinach życia.
- Dokonywanie świadomego wyboru przeglądanych stron internetowych.
- Stosowanie zasad dobrego i taktownego zachowania w Internecie.
- Rozumienie zagrożeń wynikających z rozwoju technologii informacyjno-komunikacyjnej.
- Rozumienie znaczenia przestrzegania norm prawnych w zakresie korzystania z technologii informacyjno-komunikacyjnej.
- Przestrzeganie zasad współpracy w grupie.

Reprezentacja danych w komputerze		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Poznanie sposobów prezentacji liczb i znaków w komputerze.</p> <p>Rozumienie, jaką rolę odgrywają systemy pozycyjne: dwójkowy i szesnastkowy.</p>	<p>Reprezentacja danych (liczb i znaków) w komputerze.</p> <p>System dwójkowy. Obliczanie wartości dziesiętnej liczby zapisanej w systemie dwójkowym. Wyznaczanie rozwinięcia dwójkowego liczby dziesiętnej.</p> <p>System szesnastkowy. Obliczanie wartości liczby zapisanej w systemie szesnastkowym. Wyznaczanie rozwinięcia szesnastkowego liczby dziesiętnej.</p> <p>Zależność między systemem dwójkowym a szesnastkowym. Zamienianie liczb z jednego systemu pozycyjnego na inny.</p>	<p>1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. <i>Uczeń:</i></p> <p>1) przedstawia sposoby reprezentowania różnych form informacji w komputerze: liczb, znaków [...];</p> <p>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. <i>Uczeń:</i></p> <p>11) opisuje podstawowe algorytmy i stosuje:</p> <p>a) algorytmy na liczbach całkowitych, np. reprezentacja liczb w dowolnym systemie pozycyjnym, w tym w dwójkowym i szesnastkowym;</p>

Kompresja i szyfrowanie danych		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Rozumienie, na czym polega kompresja danych i w jakim celu się ją stosuje.</p> <p>Poznanie przykładowych algorytmów kompresji danych.</p> <p>Rozumienie, na czym polega szyfrowanie danych i w jakim celu się je wykonuje.</p> <p>Poznanie przykładowych algorytmów szyfrowania danych.</p>	<p>Kompresja danych. Współczynnik kompresji. Rodzaje kompresji – bezstratna i stratna.</p> <p>Algorytmy kompresji stratnej. Algorytmy kompresji bezstratnej (statyczne, słownikowe).</p> <p>Szyfrowanie. Szyfry podstawieniowe, przestawieniowe, szyfry z kluczem, szyfry z kluczem jawnym, jednokierunkowa funkcja klucza.</p> <p>Wykorzystanie algorytmów szyfrowania w podpisie elektronicznym.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>11) opisuje podstawowe algorytmy i stosuje:</i></p> <p><i>e) algorytmy kompresji i szyfrowania, np.: kody znaków o zmiennej długości, np. alfabet Morse'a, kod Huffmana, szyfr Cezara, szyfr przestawieniowy, szyfr z kluczem jawnym (RSA), wykorzystanie algorytmów szyfrowania, np. w podpisie elektronicznym;</i></p> <p><i>2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:</i></p> <p><i>5) opisuje mechanizmy związane z bezpieczeństwem danych: szyfrowanie, klucz, certyfikat, zapora ogniowa.</i></p>

Komputer i system operacyjny		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Analizowanie modelu logicznego komputera.</p> <p>Poznanie, na czym polega działanie procesora.</p> <p>Określanie funkcji systemu operacyjnego.</p> <p>Porównywanie różnych systemów operacyjnych.</p> <p>Poznawanie nowoczesnych rozwiązań związanych z technologiami informacyjno-komunikacyjnymi.</p>	<p>Logiczny model komputera.</p> <p>Działanie procesora. Dodawanie liczb przez procesor.</p> <p>Funkcje pamięci komputera (operacyjnej i masowej).</p> <p>System operacyjny. Funkcje systemu operacyjnego.</p> <p>Przykłady systemów operacyjnych, m.in. Windows, Mac OS, Unix, Linux.</p> <p>Przykłady systemów operacyjnych dla urządzeń mobilnych.</p>	<p><i>1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej.</i> <i>Uczeń:</i></p> <p><i>2) wyjaśnia funkcje systemu operacyjnego i korzysta z nich; opisuje różne systemy operacyjne;</i></p> <p><i>4) zapoznaje się z możliwościami nowych urządzeń związanych z technologiami informacyjno-komunikacyjnymi, poznaje nowe programy i systemy oprogramowania.</i></p>

Sieci komputerowe		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Posługiwanie się terminologią sieciową.</p> <p>Rozumienie działania komputerów w sieci. Korzystanie z sieci komputerowej.</p> <p>Poznanie modelu warstwowego sieci.</p> <p>Poznanie zasad administrowania siecią komputerową w architekturze klient-serwer.</p> <p>Korzystanie z podstawowych usług sieci komputerowej.</p>	<p>Terminologia sieciowa (<i>sieć komputerowa, zasoby sieciowe, klient, serwer, switch, router, brama, konfiguracja sieciowa, protokoły sieciowe</i>). Podział sieci ze względu na wielkość i na topologię. Podstawy pracy w sieci: logowanie, udostępnianie i mapowanie zasobów.</p> <p>Warstwowy model sieci (warstwa fizyczna, łącza danych, sieciowa, aplikacji, transportu, sesji, prezentacji). Funkcje i usługi poszczególnych warstw.</p> <p>Określanie ustawień sieciowych danego komputera i jego lokalizacji w sieci (adres sieciowy, podsieci IPv4, protokoły DHCP, DNS i TCP, określanie ustawień sieciowych komputera).</p> <p>Zasady administrowania siecią komputerową w architekturze „klient-serwer”.</p>	<p><i>1. Posługiwanie się komputerem i jego oprogramowaniem, korzystanie z sieci komputerowej. Uczeń:</i></p> <p><i>3) przedstawia warstwowy model sieci komputerowych, określa ustawienia sieciowe danego komputera i jego lokalizacji w sieci, opisuje zasady administrowania siecią komputerową w architekturze klient-serwer, prawidłowo posługuje się terminologią sieciową korzysta z usług w sieci komputerowej, lokalnej i globalnej, związanych z dostępem do informacji, wymianą informacji i komunikacją;</i></p> <p><i>4) zapoznaje się z możliwościami nowych urządzeń związanych z technologiami informacyjno-komunikacyjnymi, poznaje nowe programy i systemy oprogramowania.</i></p> <p><i>3. Komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych. Uczeń:</i></p> <p><i>1) wykorzystuje zasoby i usługi sieci komputerowych w komunikacji z innymi użytkownikami, w tym do przesyłania i udostępniania danych;</i></p> <p><i>2) bierze udział w dyskusjach w sieci (forum internetowe, czat).</i></p>

Zadania projektowe		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Opracowywanie indywidualnych i zespołowych projektów z wykorzystaniem metod i narzędzi informatyki.</p> <p>Rozwijanie własnych zainteresowań z wykorzystaniem technologii informacyjno-komunikacyjnych (TIK).</p> <p>Stosowanie norm etycznych i prawnych związanych z TIK.</p> <p>Dostrzeganie szans dla rozwoju społeczeństwa wynikających z rozwoju TIK.</p> <p>Świadomość zagrożeń, z jakimi możemy się spotkać, korzystając z technologii informacyjno-komunikacyjnych.</p>	<p>Przygotowanie projektu. Etapy pracy nad projektem.</p> <p>Wybrane normy etyczne i prawne, dotyczące m.in. korzystania z cudzych materiałów, zasad korzystania z programów komputerowych, rodzajów licencji, rozpowszechniania programów komputerowych.</p> <p>Bezpieczeństwo i ochrona danych oraz informacji w komputerze i w sieciach komputerowych (rodzaje zagrożeń, sposoby ochrony).</p> <p>Rozwój informatyki i technologii informacyjno-komunikacyjnych, m.in. najważniejsze elementy procesu rozwoju informatyki i TIK.</p> <p>Szanse i zagrożenia związane z rozwojem informatyki i TIK, m.in.: uzależnienie od komputera i Internetu, zanik związków międzyludzkich „twarzą w twarz”, anonimowość kontaktów, zagrożenia prywatności, cyberprzemoc, niewiarygodność informacji.</p> <p>Komunikowanie się z wykorzystaniem nowoczesnych technologii, m.in. za pomocą urządzeń mobilnych.</p> <p>Wykorzystywanie zasobów edukacyjnych zamieszczonych w Internecie, wymiana informacji z innymi użytkownikami.</p>	<p>6. <i>Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:</i></p> <p>1) <i>opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;</i></p> <p>2) <i>korzysta z zasobów edukacyjnych udostępnianych na portalach przeznaczonych do kształcenia na odległość.</i></p> <p>7. <i>Uczeń wykorzystuje komputer i technologie informacyjno-komunikacyjne do rozwijania swoich zainteresowań, opisuje zastosowania informatyki, ocenia zagrożenia i ograniczenia, docenia aspekty społeczne rozwoju i zastosowań informatyki:</i></p> <p>1) <i>opisuje najważniejsze elementy procesu rozwoju informatyki i technologii informacyjno-komunikacyjnych;</i></p> <p>2) <i>wyjaśnia szanse i zagrożenia dla rozwoju społecznego i gospodarczego oraz dla obywateli, związane z rozwojem informatyki i technologii informacyjno-komunikacyjnych;</i></p> <p>3) <i>stosuje normy etyczne i prawne związane z rozpowszechnianiem programów komputerowych, bezpieczeństwem i ochroną danych oraz informacji w komputerze i w sieciach komputerowych;</i></p> <p>4) <i>omawia zagadnienia przestępczości komputerowej, w tym piractwo komputerowe, nielegalne transakcje w sieci;</i></p> <p>5) <i>przygotowuje się do świadomego wyboru kierunku i zakresu dalszego kształcenia informatycznego.</i></p> <p>3. <i>Komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych. Uczeń:</i></p> <p>1) <i>wykorzystuje zasoby i usługi sieci komputerowych w komunikacji z innymi użytkownikami, w tym do przesyłania i udostępniania danych;</i></p> <p>2) <i>bierze udział w dyskusjach w sieci (forum internetowe, czat).</i></p>

3.3.Opracowywanie informacji za pomocą komputera

Szczegółowe cele wychowawcze:

- Odkrywanie nowych obszarów zastosowań programów komputerowych oraz wykorzystywanie ich do rozwiązywania problemów z różnych dziedzin
- Rozwijanie chęci samodzielnego poznawania nowych możliwości programów użytkowych.
- Stosowanie przepisów prawa w zakresie korzystania z cudzych materiałów pochodzących z różnych źródeł.
- Dbanie o poprawność językową przy tworzeniu dokumentów komputerowych, zwłaszcza publikowanych w Internecie.

Algorytmy i zależności funkcyjne w arkuszu kalkulacyjnym		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Poznanie wybranych możliwości arkusza kalkulacyjnego i stosowanie ich do rozwiązywania zadań z różnych dziedzin.</p> <p>Poznanie sposobu prezentacji danych w postaci wykresu. Dobieranie typu wykresu odpowiedniego do danych.</p>	<p>Algorytm liniowy w arkuszu kalkulacyjnym.</p> <p>Algorytm z warunkami w arkuszu kalkulacyjnym.</p> <p>Realizowanie iteracji w arkuszu kalkulacyjnym.</p> <p>Zastosowanie algorytmów iteracyjnych w rysowaniu fraktali, m.in.: śnieżynki Kocha, dywanu i trójkąta Sierpińskiego, paprotki Barnsleya.</p> <p>Prezentacja zależności funkcyjnych w arkuszu kalkulacyjnym. Rysowanie wykresu funkcji liniowej, wielomianu, wybranej funkcji trygonometrycznej i funkcji logarytmicznej.</p> <p>Dobór typu wykresu odpowiedniego do prezentowanych danych.</p>	<p>4. <i>Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów. Uczeń:</i></p> <p>4) <i>wykorzystuje arkusz kalkulacyjny do obrazowania zależności funkcyjnych i do zapisywania algorytmów.</i></p> <p>5. <i>Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p>25) <i>dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</i></p>

Multimedia i grafika komputerowa		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Uświadomienie twórczego charakteru tworzenia grafiki komputerowej.</p> <p>Rozróżnianie grafiki wektorowej i rastrowej.</p> <p>Poznanie przykładowych możliwości programów przeznaczonych do edycji obrazu, tworzenia animacji i obróbki filmu.</p> <p>Rozumienie zasad łączenia poszczególnych elementów multimedialnych, np. obrazu z dźwiękiem.</p> <p>Stosowanie właściwych formatów do zapisu plików graficznych.</p>	<p>Klasyfikacja urządzeń multimedialnych. Reprezentacja obrazu w komputerze.</p> <p>Modele barw. Sposoby zapisu barw.</p> <p>Własności grafiki bitmapowej, wektorowej oraz trójwymiarowej.</p> <p>Reprezentacja dźwięku w komputerze. Formaty: „Wave”, MP3, MIDI.</p> <p>Możliwości komputera w zakresie edycji obrazu, dźwięku, animacji i wideo.</p> <p>Reprezentacja obrazów ruchomych. Zasady działania animacji. Przykład tworzenia animacji.</p> <p>Opracowywanie filmu w programach komputerowych.</p> <p>Tworzenie i edytowanie obrazów rastrowych i wektorowych.</p> <p>Podstawy pracy z obrazem z wykorzystaniem przykładowych edytorów grafiki rastrowej i wektorowej.</p> <p>Tryby pracy narzędzi malarskich.</p> <p>Wybór fragmentów obrazu oraz praca z maskami.</p> <p>Operacje na barwach.</p> <p>Przekształcenia geometryczne i filtry.</p> <p>Narzędzie klonowania.</p> <p>Praca z warstwami obrazu.</p> <p>Fotomontaże.</p>	<p><i>4. Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów. Uczeń:</i></p> <p><i>1) opisuje podstawowe modele barw i ich zastosowanie;</i></p> <p><i>2) określa własności grafiki rastrowej i wektorowej oraz charakteryzuje podstawowe formaty plików graficznych, tworzy i edytuje obrazy rastrowe i wektorowe z uwzględnieniem warstw i przekształceń;</i></p> <p><i>3) przetwarza obrazy i filmy, np.: zmienia rozdzielczość, rozmiar, model barw, stosuje filtry;</i></p> <p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>25) dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</i></p> <p><i>6. Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:</i></p> <p><i>1) opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;</i></p>

Opracowywanie tekstu i prezentacji multimedialnej		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Rozumienie ogólnych metod opracowywania dokumentu tekstowego (w tym wielostronicowego) w edytorze tekstu, niezależnych od programu i jego wersji.</p> <p>Poznanie zasad śledzenia zmian w dokumencie.</p>	<p>Zasady opracowywania dokumentów tekstowych w edytorze tekstu.</p> <p>Recenzja dokumentu tekstowego. Śledzenie zmian. Wstawianie komentarzy. Porównywanie dokumentów.</p>	<p>4. <i>Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów.</i></p> <p>6. <i>Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:</i></p>
<p>Dostosowywanie treści, formy i rodzaju prezentacji multimedialnej do celu prezentacji.</p> <p>Wykorzystanie możliwości tworzenia prezentacji komputerowej do przygotowywania prac domowych z różnych przedmiotów.</p>	<p>Zasady przygotowywania prezentacji multimedialnej.</p> <p>Typy prezentacji i ich zastosowanie: prezentacja wspomagająca wystąpienie prelegenta, prezentacja typu kiosk (samouruchamiająca się), prezentacja typu kiosk (do samodzielnego przeglądania przez odbiorcę).</p> <p>Zapisywanie dokumentu tekstowego i prezentacji multimedialnej w formacie PDF.</p>	<p>1) <i>opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;</i></p> <p>2) <i>korzysta z zasobów edukacyjnych udostępnianych na portalach przeznaczonych do kształcenia na odległość.</i></p> <p>5. <i>Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p>25) <i>dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</i></p>

3.4. Bazy danych

Szczegółowe cele wychowawcze:

- Przestrzeganie zasad bezpiecznej pracy przy komputerze.
- Odpowiedzialność za ogólny porządek na stanowisku komputerowym.
- Stosowanie zasad zdrowej pracy przy komputerze, w tym planowanie przerw w pracy i rekreacji na świeżym powietrzu.
- Szanowanie pracy innych. Przestrzeganie zasad ochrony zasobów komputera.
- Rozwijanie dociekliwości poznawczej ukierunkowanej na rzetelne zdobywanie wiedzy.
- Przestrzeganie zasad współpracy w grupie.

Projektowanie relacyjnej bazy danych		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Stosowanie techniki komputerowej do gromadzenia i przetwarzania informacji.</p> <p>Poznanie zasad działania i obszarów zastosowań programów baz danych oraz korzyści z pozyskiwania i wymiany informacji.</p> <p>Poznanie sposobu projektowanie relacyjnej bazy danych.</p> <p>Rozumienie, na czym polega relacja w bazie danych.</p> <p>Stosowanie kwerend do wyszukiwania informacji.</p> <p>Rozumienie, na czym polega przetwarzanie danych.</p> <p>Poznanie zasad tworzenia korespondencji seryjnej.</p>	<p>Pojęcia związane z bazami danych: baza danych, przetwarzanie danych, tabela, rekord, pole, typ pola, relacja, formularz, kwerenda, raport, klucz podstawowy, indeks.</p> <p>Cechy relacyjnej bazy danych. Typy relacji: „jeden do wielu”, „jeden do jednego”, „wiele do wielu”.</p> <p>Podstawowe operacje wykonywane na danych (wprowadzanie, modyfikowanie, sortowanie, indeksowanie, wyszukiwanie, prezentacja).</p> <p>Projektowanie relacyjnej bazy danych, składającej się z trzech tabel połączonych relacją.</p> <p>Zasady przygotowania tabel. Określanie typu danych. Definiowanie kluczy podstawowych, w tym kluczy złożonych. Określanie relacji. Integralność danych.</p> <p>Projektowanie formularzy, w tym formularzy z podformularzem. Kontrolowanie wprowadzania danych. Ograniczenia dla danych i wartości domyślne. Umieszczanie pól kombi, przycisków poleceń i nawigacyjnych na formularzu.</p> <p>Tworzenie kwerendy wybierającej. Stosowanie wybranych funkcji standardowych w kwerendach. Korzystanie z parametrów w kwerendzie.</p> <p>Tworzenie raportu na podstawie kwerendy.</p> <p>Łączenie informacji z bazy danych z dokumentami tekstowymi – zasady tworzenia korespondencji seryjnej.</p> <p>Importowanie danych z arkusza kalkulacyjnego i z dokumentu tekstowego do tabeli bazy danych.</p> <p>Eksportowanie danych do dokumentu tekstowego i do arkusza kalkulacyjnego.</p> <p>Wykorzystywanie danych z bazy do korespondencji seryjnej: listy seryjne i etykiety adresowe.</p>	<p>2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. <i>Uczeń:</i></p> <p>1) projektuje relacyjną bazę danych z zapewnieniem integralności danych;</p> <p>2) stosuje metody wyszukiwania i przetwarzania informacji w relacyjnej bazie danych (język SQL);</p> <p>3) tworzy aplikację bazodanową, [...] wykorzystującą język zapytań, kwerendy, raporty; zapewnia integralność danych na poziomie pól, tabel, relacji;</p> <p>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. <i>Uczeń:</i></p> <p>25) dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</p> <p>28) realizuje indywidualnie lub zespołowo projekt programistyczny z wydzieleniem jego modułów, w ramach pracy zespołowej, dokumentuje pracę zespołu.</p>

Uwaga: Poszczególne etapy przygotowania bazy danych proponuję omówić na przykładzie tworzenia prostej bazy w programie Microsoft Access.

Tworzenie kwerend z wykorzystaniem języka SQL		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Dowiedzenie się, czym jest język SQL.</p> <p>Poznanie podstawowych instrukcji języka SQL.</p> <p>Stosowanie języka SQL do wyszukiwania i przetwarzania informacji w relacyjnej bazie danych.</p>	<p>Język SQL jako strukturalny język zapytań.</p> <p>Zastosowanie instrukcji SELECT.</p> <p>Wybrane klauzule instrukcji SELECT:</p> <ul style="list-style-type: none"> • klauzula ORDER BY, • klauzula INNER JOIN, • klauzula LEFT JOIN, • klauzula GROUP BY. <p>Dopisywanie rekordów.</p> <p>Aktualizacja danych.</p> <p>Usuwanie rekordów.</p>	<p><i>2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:</i></p> <p><i>2) stosuje metody wyszukiwania i przetwarzania informacji w relacyjnej bazie danych (język SQL);</i></p> <p><i>3) tworzy aplikację bazodanową, [...] wykorzystującą język zapytań, kwerendy, raporty; zapewnia integralność danych na poziomie pól, tabel, relacji;</i></p> <p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>25) dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</i></p>

Realizacja projektu programistycznego		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Poznanie pracy informatyków i organizacji pracy zespołowej.</p> <p>Współdziałanie w zespole, wypowiedzanie własnego zdania, ale również rozumienie i uwzględnianie poglądów innych.</p> <p>Stosowanie zasady dokładnego wykonywania swojej części pracy zgodnie z wcześniejszymi ustaleniami.</p>	<p>Pojęcia związane z projektowaniem systemów informatycznych: analiza systemu, założenia, system informacyjny i informatyczny, projekt techniczny, szablon.</p> <p>Zasady projektowania systemów informatycznych.</p> <p>Etapy projektowania: analiza systemu informacyjnego, opracowanie założeń, wykonanie projektu technicznego, wykonanie projektu informatycznego, testowanie systemu, wdrożenie.</p> <p>Realizacja projektu na zadany (lub samodzielnie wybrany) temat zgodnie z etapami projektowania.</p> <p>Podział zadania na zadania szczegółowe.</p> <p>Zastosowanie szablonów (w dokumentach tekstowych) do przygotowywania założeń projektu programistycznego.</p>	<p>2. <i>Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:</i></p> <p>3) <i>tworzy aplikację bazodanową, [...] wykorzystującą język zapytań, kwerendy, raporty; zapewnia integralność danych na poziomie pól, tabel, relacji;</i></p> <p>4) <i>znajduje odpowiednie informacje niezbędne do realizacji projektów z różnych dziedzin;</i></p> <p>5. <i>Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p>25) <i>dobiera właściwy program użytkowy lub samodzielnie napisany program do rozwiązywanego zadania;</i></p> <p>6. <i>Uczeń wykorzystuje komputer oraz programy i gry edukacyjne do poszerzania wiedzy i umiejętności z różnych dziedzin:</i></p> <p>1) <i>opracowuje indywidualne i zespołowe projekty przedmiotowe i międzyprzedmiotowe z wykorzystaniem metod i narzędzi informatyki;</i></p> <p>2) <i>korzysta z zasobów edukacyjnych udostępnianych na portalach przeznaczonych do kształcenia na odległość.</i></p>

3.5. Algorytmika i programowanie – problemy zaawansowane

Szczegółowe cele wychowawcze:

- Przestrzeganie zasad bezpiecznej pracy przy komputerze.
- Szanowanie pracy innych. Przestrzeganie zasad ochrony zasobów komputera.
- Rozwijanie dociekliwości poznawczej ukierunkowanej na rzetelne zdobywanie wiedzy.
- Ćwiczenie precyzyjnego wyrażania myśli.
- Dostrzeganie mechanizmów logicznych obecnych w otaczającym świecie.
- Odpowiedzialność za podejmowane działania.
- Wykorzystywanie zasad projektowania algorytmów i programowania w życiu codziennym.

Wprowadzanie danych i wyprowadzanie wyników		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Poznanie różnych sposobów wprowadzania danych i wyprowadzania wyników.</p> <p>Stosowanie różnych typów danych.</p> <p>Dobieranie odpowiedniego typu danych do realizacji danego algorytmu.</p> <p>Rozwijanie umiejętności sprawnego programowania.</p>	<p>Sposoby wprowadzania danych i wyprowadzania wyników.</p> <p>Formatowanie wyprowadzanych wyników.</p> <p>Własne typy danych (typ okrojony, typ wyczerpujący) i stałe.</p> <p>Stos jako przykład struktury danych. Operacje wykonywane na stosie. Zastosowanie stosu – odwrotna notacja polska (ONP).</p> <p>Losowanie danych.</p> <p>Sprawdzanie poprawności danych.</p>	<p>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</p> <p>6) ocenia własności rozwiązania algorytmicznego (komputerowego), np. zgodność ze specyfikacją, efektywność działania;</p> <p>7) opracowuje i przeprowadza wszystkie etapy prowadzące do otrzymania poprawnego rozwiązania problemu: od sformułowania specyfikacji problemu po testowanie rozwiązania;</p> <p>12) projektuje rozwiązanie problemu (realizację algorytmu) i dobiera odpowiednią strukturę danych;</p> <p>14) dobiera odpowiednie struktury danych do realizacji algorytmu [...];</p> <p>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</p> <p>22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;</p> <p>23) stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu;</p>

Wybrane typy i struktury danych, w tym dynamiczne		
Cele kształcenia	Treści nauczania	Podstawa programowa
Stosowanie odpowiednich struktur danych, w tym struktur dynamicznych, do realizacji algorytmu.	<p>Przetwarzanie danych typu rekordowego (strukturalnego). Deklaracja typu rekordowego (strukturalnego). Rekord (struktura) jako elementy rekordu (struktury).</p> <p>Przetwarzanie plików. Deklarowanie zmiennych typu plikowego. Korzystanie z plików. Odczyt danych z pliku. Zapis danych do pliku. Obsługa błędów. Przetwarzanie plików tekstowych.</p> <p>Typ plikowy a baza danych.</p> <p>Dynamiczne struktury danych. Wskaźniki. Deklarowanie zmiennych typu wskaźnikowego. Stosowanie zmiennej wskaźnikowej. Tworzenie zmiennych dynamicznych.</p> <p>Wybrane struktury dynamiczne: tablica dynamiczna, lista jednokierunkowa, drzewo binarne.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>12) projektuje rozwiązanie problemu (realizację algorytmu) i dobiera odpowiednią strukturę danych;</i></p> <p><i>14) dobiera odpowiednie struktury danych do realizacji algorytmu, w tym struktury dynamiczne;</i></p> <p><i>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</i></p> <p><i>22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;</i></p> <p><i>23) stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu;</i></p> <p><i>24) dobiera najlepszy algorytm, odpowiednie struktury danych i oprogramowanie do rozwiązania postawionego problemu;</i></p>

Programowanie modułowe i obiektowe		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Stosowanie zasad programowania modułowego.</p> <p>Stosowanie zasad programowania obiektowego.</p> <p>Rozumienie cech programowania obiektowego.</p>	<p>Programowanie modułowe. Określenie modułu.</p> <p>Deklarowanie korzystania ze standardowych modułów.</p> <p>Zasady tworzenia własnych modułów.</p> <p>Podstawowe pojęcia programowania obiektowego: <i>klasa, obiekt, pola, metody</i>.</p> <p>Definiowanie klas. Deklarowanie pól prywatnych.</p> <p>Cechy programowania obiektowego: dziedziczenie i polimorfizm. Metody wirtualne.</p>	<p><i>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</i></p> <p><i>15) stosuje zasady programowania strukturalnego i modułowego do rozwiązywania problemu;</i></p> <p><i>21) przeprowadza komputerową realizację algorytmu i rozwiązania problemu;</i></p> <p><i>22) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu i uruchamianiu programów;</i></p>

Wybrane algorytmy		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Stosowanie podejścia algorytmicznego w rozwiązywaniu zadań.</p> <p>Poznanie i stosowanie wybranych algorytmów na liczbach naturalnych, algorytmów sortowania, algorytmów numerycznych i badających własności geometryczne.</p> <p>Stosowanie wybranych struktur danych i ich właściwe dobieranie do rozwiązywanego problemu.</p>	<p>Wieże Hanoi.</p> <p>Przykłady algorytmów sortowania: sortowanie przez wstawianie, przez scalanie, metodą szybką.</p> <p>Przeszukiwanie binarne.</p> <p>Zamiana liczb między dowolnymi systemami pozycyjnymi: obliczanie wartości dziesiętnej liczby, zapisywanie liczby dziesiętnej w systemie liczbowym o określonej podstawie.</p> <p>Przykłady algorytmów na liczbach naturalnych: generowanie liczb pierwszych (podejście naiwne, sito Eratostenesa), rozkładanie liczb na czynniki pierwsze, liczby doskonałe.</p> <p>Szybkie podnoszenie do potęgi – algorytm binarny „od lewej do prawej”.</p> <p>Algorytmy badające własności geometryczne: przynależność punktu do odcinka, badanie położenia punktu względem prostej.</p> <p>Reprezentacja danych numerycznych w komputerze: reprezentacja binarna liczb ujemnych, reprezentacja stałopozycyjna liczb, reprezentacja zmiennopozycyjna liczb.</p> <p>Właściwości arytmetyki komputerowej. Błędy obliczeń i stabilność algorytmów.</p> <p>Przykłady algorytmów numerycznych: wyznaczanie miejsca zerowego funkcji, obliczanie wartości pierwiastka kwadratowego, obliczanie pola obszaru ograniczonego.</p>	<p>5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego. Uczeń:</p> <p>11) opisuje podstawowe algorytmy i stosuje:</p> <p>a) algorytmy na liczbach całkowitych, np.:</p> <ul style="list-style-type: none"> – reprezentacja liczb w dowolnym systemie pozycyjnym, w tym w dwójkowym i szesnastkowym, – sprawdzanie, czy liczba jest liczbą pierwszą, doskonałą, – rozkładanie liczby na czynniki pierwsze, <p>b) algorytmy wyszukiwania i porządkowania (sortowania), np.:</p> <ul style="list-style-type: none"> – algorytmy sortowania ciągu liczb: [...] przez wstawianie liniowe lub binarne, przez scalanie, szybki, [...], <p>c) algorytmy numeryczne, np.:</p> <ul style="list-style-type: none"> – obliczanie wartości pierwiastka kwadratowego, – zastosowania schematu Hornera: reprezentacja liczb w różnych systemach liczbowych, szybkie podnoszenie do potęgi, – wyznaczanie miejsc zerowych funkcji metodą połowienia, – obliczanie pola obszarów zamkniętych, <p>d) algorytmy na tekstach, np.:</p> <ul style="list-style-type: none"> – obliczanie wartości wyrażenia podanego w postaci odwrotnej notacji polskiej, <p>f) algorytmy badające własności geometryczne, np.:</p> <ul style="list-style-type: none"> – badanie położenia punktów względem prostej, – badanie przynależności punktu do odcinka] – konstrukcje rekurencyjne, np. drzewo binarne, <p>27) wyjaśnia źródło błędów w obliczeniach komputerowych (błąd względny, błąd bezwzględny);</p>

3.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci

Aplikacje bazodanowe dostępne za pośrednictwem		
Cele kształcenia	Treści nauczania	Podstawa programowa
<p>Rozumienie, czym różni się strona internetowa statyczna od dynamicznej.</p> <p>Poznanie technologii dynamicznego generowania stron internetowych.</p> <p>Poznanie zasad tworzenia sieciowej aplikacji bazodanowej.</p> <p>Dbłość o redakcyjną i merytoryczną poprawność oraz zgodność z prawem umieszczanych w Internecie tekstów i materiałów.</p>	<p>Strona internetowa jako wynik dynamicznego przetwarzania.</p> <p>Podstawy pisania skryptów w języku PHP. Wyświetlanie danych instrukcją <code>echo</code>. Kodowanie UTF-8. Stosowanie zmiennych i operatorów.</p> <p>Przesyłanie danych za pomocą formularzy HTML. Tworzenie kodu programu wyświetlającego prosty formularz i odbierającego dane z formularza.</p> <p>Witryny internetowe oparte na bazach danych. Korzystanie z baz danych z poziomu PHP. Tworzenie konta użytkownika i bazy danych na serwerze MySQL. Łączenie się z bazą MySQL z poziomu PHP. Wykonywanie zapytań do bazy danych z poziomu PHP.</p> <p>Tworzenie prostej księgi gości. Tworzenie tabeli na wpisy z księgi gości. Dodawanie wpisów do księgi gości za pomocą instrukcji INSERT. Tworzenie formularza dodającego wpisy do bazy danych. Odczytywanie danych z bazy za pomocą instrukcji SELECT – wyświetlanie listy wpisów z księgi gości.</p>	<p>2. Wyszukiwanie, gromadzenie, selekcjonowanie, przetwarzanie i wykorzystywanie informacji, współtworzenie zasobów w sieci, korzystanie z różnych źródeł i sposobów zdobywania informacji. Uczeń:</p> <p>3) tworzy aplikację bazodanową, w tym sieciową, wykorzystującą język zapytań, [...];</p> <p>3. Komunikowanie się za pomocą komputera i technologii informacyjno-komunikacyjnych. Uczeń:</p> <p>1) wykorzystuje zasoby i usługi sieci komputerowych w komunikacji z innymi użytkownikami, w tym do przesyłania i udostępniania danych;</p>

Uwaga: Samodzielna instalacja serwera Apache, interpretera PHP i serwera MySQL w systemie Windows jest czasochłonna i wymaga ręcznej edycji plików konfiguracyjnych, aby komponenty te poprawnie ze sobą współpracowały. Proponuję zastosowanie pakietu XAMPP Portable Lite 1.8.1 (w wersji dla systemu Windows).

4. Sposoby osiągania celów kształcenia i wychowania

4.1. Podstawy algorytmiki i programowania

Prezentacja algorytmów liniowych w wybranej notacji	
Czynności nauczyciela	Czynności uczniów
Przypomina regulamin pracowni komputerowej oraz zasady bezpieczeństwa i higieny pracy przy komputerze. Przypomina zasady organizacji pracy w szkolnej sieci komputerowej.	Zapoznają się z regulaminem, zasadami pracy i bezpieczeństwa w pracowni komputerowej, a także stosują się do nich w czasie zajęć.
Porządkuje wiedzę dotyczącą algorytmiki, nabytą przez uczniów w gimnazjum i na lekcjach informatyki realizowanej w zakresie podstawowym. Przypomina zależności między problemem, algorytmem a programem komputerowym (przedstawia program jako realizację algorytmu). Przypomina sposoby przedstawiania algorytmów. Zwraca uwagę na szczegółowe zasady zapisu algorytmów w różnych notacjach. Proponuje odpowiednie narzędzie TI umożliwiające prezentację graficzną algorytmów, np. oprogramowanie edukacyjne do konstrukcji schematów blokowych, edytor tekstu. Dyskutuje z uczniami o poprawności działania algorytmów wykonanych w postaci schematów.	Przypominają wiadomości z gimnazjum i lekcji informatyki realizowanej w zakresie podstawowym dotyczące algorytmiki. Podają przykłady poznanych algorytmów, w tym z innych przedmiotów szkolnych (matematyki, fizyki) oraz znane im sposoby ich przedstawiania. Określają specyfikację zadań. Zapisują algorytmy: <ul style="list-style-type: none"> • w postaci listy kroków, • w postaci schematu blokowego, • w języku programowania. Analizują poprawność budowy schematu blokowego.
Na przykładzie wybranego języka wysokiego poziomu wyjaśnia, na czym polega programowanie. Przedstawia etapy programowania – implementację, kompilację, uruchomienie, testowanie. Omawia, na czym polega zapis algorytmu liniowego w postaci programu.	Wyjaśniają pojęcia: <i>język programowania, program źródłowy, program wynikowy, kompilacja, translacja, interpretacja</i> . Zapoznają się z kompilatorem danego języka. Piszą prosty program, kompilują go i uruchamiają. Uczą się poprawiać błędy w programie. Wykonują program i testują go, podstawiając różne dane. Realizują przykładowe algorytmy liniowe w wybranym języku programowania.

Podstawowe zasady programowania	
Czynności nauczyciela	Czynności uczniów
<p>Omawia ogólną strukturę programu w wybranym języku programowania, zasady składni, podział typów danych, deklarowanie zmiennych, wprowadzanie danych, wyprowadzanie wyników, instrukcję przypisania.</p> <p>Omawia zasady zapisu algorytmów z warunkami i algorytmów iteracyjnych w postaci programów, w tym warunki i pętle zagnieżdżone. Zwraca uwagę na poprawne określanie warunków zakończenia iteracji w programie, aby uniknąć jego „zapętlenia”.</p> <p>Omawia zasady dobrego stylu programowania.</p>	<p>Pisząc programy w wybranym języku programowania, zapoznają się ze strukturą programu, zasadami składni i stosowaniem różnych typów danych.</p> <p>Na konkretnych przykładach poznają działanie instrukcji warunkowych i iteracyjnych w wybranym języku programowania wysokiego poziomu.</p> <p>Stosują, zależnie od zadania, odpowiednią instrukcję iteracyjną.</p>
<p>Omawia modele programowania, wskazując podobieństwa i różnice.</p> <p>Wspólnie z uczniami formułuje zasady zapisu strukturalnego (z wykorzystaniem procedur i funkcji) oraz omawia możliwości jego zastosowania. Określa metodę programowania zstępującego.</p> <p>Wyjaśnia na konkretnym przykładzie różne sposoby przekazywania parametrów.</p> <p>Podkreśla możliwość wielokrotnego wykorzystania tej samej procedury lub funkcji w różnych segmentach (częściach) programu oraz w innych programach.</p>	<p>Projektują rozwiązania zadań, stosując technikę programowania strukturalnego.</p> <p>Definiują procedury i funkcje w wybranym języku programowania – z parametrami i bez parametrów. Wykonując ćwiczenia, stosują różne sposoby przekazywania parametrów.</p> <p>Rozważają podział algorytmu na niezależne fragmenty, realizujące określone operacje. Analizują możliwość wielokrotnego wykorzystania zadeklarowanych procedur i funkcji w tym samym programie.</p>
<p>Wyjaśnia, na czym polega dobór struktur danych do algorytmu. Pokazuje na przykładach stosowanie tablic i łańcuchów.</p>	<p>Poznają przykładowe typy danych złożonych – tablice i łańcuchy. Deklarują dane typu tablicowego i łańcuchowego, dobierając typ danych do algorytmu. Wczytują i wyprowadzają elementy tablicy. Wprowadzają dane tekstowe.</p>
<p>Wyjaśnia na konkretnym algorytmie (np. obliczania silni), w jaki sposób zapisuje się algorytm iteracyjny i rekurencyjny w postaci programu. Wskazuje różnice między zastosowaniem instrukcji iteracyjnych a zapisem procedur i funkcji rekurencyjnych.</p> <p>Posługując się przykładami, wyjaśnia zasady zapisywania algorytmu sortowania (np. sortowanie bąbelkowe). Poleca zdefiniowanie potrzebnych procedur lub funkcji.</p>	<p>Projektują i realizują w języku programowania wysokiego poziomu wybrane algorytmy iteracyjne i rekurencyjne.</p> <p>Zapisują wybrane algorytmy, które zrealizowali iteracyjnie, w sposób rekurencyjny, np. algorytm generowania liczb Fibonacciego.</p> <p>Zapisują wybrany algorytm sortowania (np. algorytm sortowania bąbelkowego) w postaci programu.</p> <p>Oceniają poprawność komputerowego rozwiązania, testując programy na różnych danych.</p>

Techniki algorytmiczne i wybrane algorytmy	
Czynności nauczyciela	Czynności uczniów
<p>Określa różnicę między algorytmem liniowym a algorytmem z warunkami. Omawia na przykładach cechy tych algorytmów.</p> <p>Opisuje i demonstrowuje wykorzystanie techniki iteracji. Definiuje pojęcia: <i>iteracja</i>, <i>pętla</i>. Poleca zbudowanie schematu blokowego algorytmu realizującego iterację, np. obliczania iloczynu n liczb. Zwraca uwagę na stosowanie poprawnych sposobów zakończenia iteracji. Podaje je. Wyjaśnia zastosowanie tzw. licznika wykonywanych kroków iteracji. Definiuje pojęcie <i>zmiennej sterującej</i>.</p> <p>Zapoznaje uczniów z pojęciem rekurencji. Podaje przykłady definicji rekurencyjnych. Wyjaśnia różnicę między iteracją i rekurencją.</p> <p>Wyjaśnia, w jaki sposób w poszczególnych notacjach prezentować sytuacje warunkowe, w tym warunki zagnieżdżone, iterację, w tym pętle zagnieżdżone, rekurencję.</p>	<p>Podają przykłady algorytmów liniowych i z warunkami. Porównują cechy tych algorytmów. Określają sytuacje warunkowe w znanych im algorytmach.</p> <p>Podają przykłady wykorzystania technik iteracyjnych w algorytmach. Budują schematy blokowe realizujące działania w pętli; stosują również pętle zagnieżdżone. Na podanych przykładach ćwiczą różne sposoby zakończenia iteracji. Określają liczbę kroków iteracji.</p> <p>Zapoznają się z pojęciem rekurencji i odnajdują przykłady „zjawisk rekurencyjnych” w życiu codziennym oraz zadaniach szkolnych. Realizują wybrany algorytm rekurencyjnie. Porównują jego realizację iteracyjną z realizacją rekurencyjną.</p> <p>Poznają i wykorzystują w różnym zastosowaniu wybrane techniki porządkowania (sortowania) elementów.</p>
<p>Omawia przykładowe algorytmy z warunkami, np. warunek istnienia trójkąta. Wyjaśnia algorytm znajdowania elementu najmniejszego (lub największego) w zbiorze oraz jednoczesnego znajdowania minimum i maksimum, w tym metodę „dziel i zwyciężaj”.</p> <p>Podaje przykłady tego samego algorytmu zrealizowanego iteracyjnie i rekurencyjnie.</p> <p>Wspólnie z uczniami analizuje podejście zachłanne na przykładzie algorytmu wydawania reszty.</p> <p>Omawia wybrane algorytmy sortowania. Wskazuje ich zastosowanie w porządkowaniu różnych zbiorów danych.</p> <p>Podaje i omawia wybrane algorytmy operujące na tekstach.</p>	<p>Zapoznają się z wybranymi algorytmami. Zapisują je w wybranej notacji.</p> <p>Rozwiązują zadania, w których konieczne jest wykorzystanie poznanych technik algorytmicznych i algorytmów, np. techniki iteracji do wyboru elementu minimalnego w ciągu n liczb i algorytmu wyboru minimum do porządkowania (sortowania) elementów.</p> <p>Wykonując konkretne zadania, realizują iteracyjnie i rekurencyjnie ten sam algorytm, m.in.: algorytm obliczania silni, algorytm Euklidesa, generowanie liczb Fibonacciego i wartości wielomianu według schematu Hornera.</p> <p>Realizują algorytm wydawania reszty metodą zachłanną. Piszą listę kroków i rysują schemat blokowy tego algorytmu.</p> <p>Stosują w praktyce algorytmy sortowania (przez wybór, bąbelkowy, pozycyjny) i algorytmy na tekstach (np. tworzenie anagramów, sprawdzanie, czy dany ciąg znaków jest palindromem).</p>

Elementy analizy algorytmów	
Czynności nauczyciela	Czynności uczniów
<p>Omawia najważniejsze własności algorytmów.</p> <p>Wyjaśnia, kiedy algorytm jest poprawny i skończony</p> <p>Definiuje pojęcie złożoności obliczeniowej algorytmu – czasowej i pamięciowej.</p> <p>Wspólnie z uczniami ocenia złożoność obliczeniową wybranych algorytmów.</p> <p>Poleca uczniom, by porównali złożoność różnych rodzajów algorytmów, które realizują to samo zadanie (dla tych samych danych).</p> <p>Wyjaśnia, na czym polega efektywność algorytmów.</p> <p>Zainteresowanym uczniom proponuje dodatkową literaturę z dziedziny algorytmiki, m.in. na temat analizy algorytmów.</p>	<p>Sprawdzają, czy dany algorytm działa zgodnie ze specyfikacją zadania – oceniają poprawność działania algorytmu.</p> <p>Analizują (śledzą) działanie algorytmu zapisanego w wybranej notacji dla przykładowych danych, np. „krok po kroku” śledzą iterację i wypisują kolejne wartości zmiennych po zrealizowaniu każdego kroku iteracji.</p> <p>Określają poprawność wybranego algorytmu, np. algorytmu obliczania silni.</p> <p>Na konkretnym przykładzie sprawdzają, kiedy algorytm jest skończony.</p> <p>Określają liczbę operacji wykonywanych na danych, np. liczbę wykonanych porównań elementów w algorytmie sortowania bąbelkowego i przez wybór (złożoność czasową algorytmu). Porównują złożoność czasową różnych algorytmów tego samego zadania (dla tych samych danych), np. dwóch algorytmów sortowania.</p> <p>Szacują złożoność pamięciową wybranego algorytmu, np. rekurencyjnego algorytmu obliczania silni.</p> <p>Sprawdzają na konkretnym przykładzie efektywność algorytmów.</p>

4.2. Komputer i sieci komputerowe

Reprezentacja danych w komputerze	
Czynności nauczyciela	Czynności uczniów
<p>Przedstawia sposoby reprezentacji liczb w komputerze.</p> <p>Odwołując się do systemu dziesiętnego, omawia istotne dla informatyki systemy pozycyjne (system dwójkowy i szesnastkowy).</p> <p>Przedstawia algorytm konwersji liczb z systemu binarnego na dziesiętny i odwrotnie oraz z systemu szesnastkowego na dziesiętny i odwrotnie.</p>	<p>Obliczają wartość dziesiętną liczby zapisanej w systemie dwójkowym. Wyznaczają rozwinięcie dwójkowe liczby dziesiętnej.</p> <p>Uczniowie zainteresowani wykonują działania na liczbach binarnych: dodawanie, odejmowanie, operacje logiczne i przesunięcia bitowe.</p> <p>Poznają system szesnastkowy. Obliczają wartość liczby zapisanej w systemie szesnastkowym. Wyznaczają rozwinięcie szesnastkowe liczb dziesiętnych.</p> <p>Ustalają zależność między systemem dwójkowym a szesnastkowym. Zamieniają liczby z jednego systemu pozycyjnego na inny.</p> <p>Zapisują algorytm zamiany liczb z systemu dwójkowego na dziesiętny i odwrotnie w postaci programu komputerowego.</p> <p>Zainteresowani zapisują algorytm konwersji liczb z dowolnego systemu pozycyjnego na inny.</p>
Kompresja i szyfrowanie danych	
Czynności nauczyciela	Czynności uczniów
<p>Dyskutuje z uczniami na temat kompresji i szyfrowania danych.</p> <p>Omawia rodzaje kompresji i ich zastosowanie. Definiuje pojęcie <i>współczynnik kompresji</i>. Podaje przykłady algorytmów kompresji stratnej i bezstratnej.</p> <p>Prezentuje wybrane algorytmy szyfrowania.</p> <p>Omawia praktyczne zastosowanie szyfrowania.</p> <p>Zleca zapoznanie się z wykorzystaniem algorytmów szyfrowania w podpisie elektronicznym.</p>	<p>Dzieli się własną wiedzą na temat kompresji i szyfrowania danych, sięgając do historii kryptografii.</p> <p>Poznają przykładowe algorytmy kompresji i szyfrowania, m.in.: szyfry podstawieniowe, przestawieniowe, szyfry z kluczem, szyfry z kluczem jawnym, algorytmy kompresji stratnej i bezstratnej (statyczne, słownikowe).</p> <p>Wykonują ćwiczenia, w których szyfrują i rozszyfrowują podane słowa, jak też całe teksty.</p> <p>Zapoznają się z wykorzystaniem algorytmów szyfrowania w podpisie elektronicznym.</p> <p>Zainteresowani uczniowie zapisują algorytm szyfrowania w postaci programu.</p>

Komputer i system operacyjny	
Czynności nauczyciela	Czynności uczniów
<p>Poleca przypomnienie (z lekcji informatyki w gimnazjum i lekcji informatyki realizowanej w zakresie podstawowym) informacji o budowie i działaniu komputera.</p> <p>Prezentuje na schemacie logiczny model komputera.</p> <p>Wyjaśnia działanie procesora.</p> <p>Zwraca uwagę na charakterystyczne cechy i funkcje różnych systemów operacyjnych. Wskazuje na ich różne zastosowania.</p> <p>Poleca powtórzenie wiadomości o zarządzaniu folderami i plikami oraz o zasadach ich ochrony. Prezentuje podobieństwa i różnice w tym zakresie między różnymi systemami operacyjnymi.</p> <p>Zleca wyszukanie informacji na temat nowoczesnych rozwiązań związanych z technologiami informacyjno-komunikacyjnymi.</p>	<p>Prezentują ogólną klasyfikację środków (urządzeń) i narzędzi (oprogramowania) TI. Omawiają części składowe komputera – ich przeznaczenie i parametry – oraz organizację pamięci komputera.</p> <p>Omawiają ogólny model komputera według idei von Neumanna.</p> <p>Poznają działanie procesora, m.in. w jaki sposób procesor dodaje liczby.</p> <p>Omawiają urządzenia pamięci masowej, podając ich przykładowe parametry i zwracając uwagę na ich rozwój.</p> <p>W Internecie i innych źródłach wyszukują informacje na temat różnych typów aktualnie używanych komputerów, w tym wiadomości o nowoczesnych urządzeniach związanych z TIK, np. o urządzeniach mobilnych.</p> <p>Przypominają z lekcji informatyki w gimnazjum i lekcji informatyki realizowanej w zakresie podstawowym poznane funkcje systemu operacyjnego.</p> <p>Omawiają ogólną strukturę systemu operacyjnego.</p> <p>Charakteryzują różne systemy operacyjne, wskazując kierunek rozwoju ich funkcji.</p> <p>Podają przykłady systemów operacyjnych.</p> <p>Wykonują ćwiczenia, które porządkują ich wiedzę i umiejętności z zakresu operacji wykonywanych na plikach, m.in. nadawanie plikom atrybutów, wyszukiwanie plików, odzyskiwanie utraconych plików. Dyskutują nad potrzebą ochrony plików.</p>

Sieci komputerowe	
Czynności nauczyciela	Czynności uczniów
<p>Poleca przypomnienie zagadnień dotyczących sieci komputerowych, w tym terminologii sieciowej, z lekcji informatyki realizowanej w zakresie podstawowym.</p> <p>Przygotowuje (jeśli są takie możliwości) niezbędne składniki, aby można było w warunkach pracowni szkolnej zbudować prostą sieć, składającą się np. z dwóch-trzech komputerów.</p> <p>Omawia warstwowy model sieci.</p> <p>Przedstawia na przykładach zasady administrowania siecią komputerową w architekturze „klient-serwer”.</p> <p>Zwraca uwagę na zagrożenia płynące z sieci, omawia przykładowe oprogramowanie do ochrony zasobów własnych komputera, m.in. zapory sieciowe.</p>	<p>Dyskutują nad korzyściami płynącymi z łączenia komputerów w sieć. Klasyfikują sieć ze względu na wielkość, omawiają rodzaj i topologię sieci działającej w szkolnej pracowni komputerowej.</p> <p>Omawiają urządzenia i elementy sieciowe oraz zasady pracy w sieci. Zwracają uwagę na te elementy i ich parametry, które dotyczą pracowni. Omawiają zasady konfiguracji prostych sieci komputerowych.</p> <p>Wykonują ćwiczenia związane z udostępnianiem zasobów itp., przypominając w ten sposób zasady wymiany informacji w sieci komputerowej.</p> <p>Dzielą się swoimi doświadczeniami w zakresie budowania małych sieci domowych. Realizują w praktyce małą sieć komputerową – konfigurują składniki sieci, udostępniają pliki, dysk, drukarki, dodają użytkowników.</p> <p>Zapoznają się z warstwowym modelem sieci, z funkcjami i usługami poszczególnych warstw (warstwą aplikacji, prezentacji, sesji, transportu, sieciową, łącz danych, fizyczną).</p> <p>Wykonując ćwiczenia, określają ustawienia sieciowe danego komputera i jego lokalizacji w sieci (adres sieciowy, podsieci IPv4, protokoły DHCP, DNS i TCP, określanie ustawień sieciowych komputera).</p> <p>Poznają zasady administrowania siecią komputerową w architekturze „klient-serwer”.</p> <p>Zapoznają się z zagrożeniami płynącymi z sieci i sposobami konfigurowania przykładowego oprogramowania do ochrony zasobów.</p>

Projekty grupowe	
Czynności nauczyciela	Czynności uczniów
<p>Poleca opracowywanie indywidualnych i zespołowych projektów.</p> <p>Przedstawia zasady opracowania projektu. Wyznacza koordynatorów grup projektowych.</p> <p>Proponuje tematykę projektów, m.in.:</p> <ul style="list-style-type: none"> • normy etyczne i prawne, ze szczególnym uwzględnieniem rozpowszechniania programów komputerowych oraz bezpieczeństwa i ochrony danych w komputerze; • bezpieczeństwo i ochrona danych oraz informacji w komputerze i w sieciach komputerowych (rodzaje zagrożeń, sposoby ochrony), • historia najnowszych osiągnięć w dziedzinie informatyki, ze szczególnym uwzględnieniem wpływu rozwoju informatyki na zachowania społeczne i gospodarkę kraju, • zalety i zagrożenia, wynikające z wykorzystywania nowoczesnych technologii, szanse dla rozwoju społeczeństwa wynikające z rozwoju TIK. <p>Podsumowuje prace nad projektami, oceniając wykonane projekty.</p>	<p>Zapoznają się z etapami przygotowania projektu.</p> <p>Przygotowują projekty (np. w formie prezentacji multimedialnej) na tematy wybrane wśród tematów podanych przez nauczyciela.</p> <p>W danej grupie wyznaczają koordynatora. Dzielą zadanie projektowe na zadania szczegółowe, które są przydzielane pomiędzy poszczególnych członków grupy.</p> <p>Wykonują zadania szczegółowe. Wyszukują i gromadzą materiały niezbędne do przygotowania projektu.</p> <p>Komunikują się z innymi członkami grupy, wykorzystując nowoczesne technologie, m.in. za pomocą urządzeń mobilnych. Wykorzystują zasoby edukacyjne zamieszczone w Internecie.</p> <p>Dyskutują na temat prezentowanych zagadnień.</p>

4.3. Opracowywanie informacji za pomocą komputera

Algorytmy i zależności funkcyjne w arkuszu kalkulacyjnym	
Czynności nauczyciela	Czynności uczniów
<p>Poleca powtórzenie zasad pracy w arkuszu kalkulacyjnym (tworzenie, kopiowanie formuł, stosowanie funkcji).</p> <p>Zleca samodzielne zapoznanie się z realizacją algorytmów liniowych i iteracyjnych w arkuszu kalkulacyjnym.</p> <p>Omawia zastosowanie algorytmów iteracyjnych w rysowaniu fraktali. Inicjuje dyskusję na temat występowania tworów przypominających fraktale w naturze.</p> <p>Poleca powtórzenie zasad tworzenia wykresów w arkuszu kalkulacyjnym, zwracając uwagę na dobór typu wykresu do rodzaju prezentowanych danych.</p> <p>Podaje praktyczne wskazówki dotyczące obrazowania zależności funkcyjnych w arkuszu kalkulacyjnym.</p>	<p>Przypominają zasady tworzenia formuł i stosowania funkcji arkusza kalkulacyjnego.</p> <p>Wykonują ćwiczenia, w których zapisują w arkuszu kalkulacyjnym algorytm liniowy i z warunkami.</p> <p>Analizują przykładowe algorytmy iteracyjne w arkuszu kalkulacyjnym. Wykonując wskazane przez nauczyciela ćwiczenia, realizują iterację w arkuszu kalkulacyjnym.</p> <p>Przypominają sposób prezentacji danych z arkusza kalkulacyjnego w postaci wykresu, zwłaszcza dobieranie odpowiedniego typu wykresu do danych.</p> <p>Zapoznają się z możliwościami zastosowania algorytmów iteracyjnych w rysowaniu fraktali, m.in.: śnieżynki Kocha, dywanu i trójkąta Sierpińskiego, paprotki Barnsleya.</p> <p>Prezentują zależności funkcyjne w arkuszu kalkulacyjnym. Rysują wykres funkcji liniowej, wielomianu, wybranej funkcji trygonometrycznej i funkcji logarytmicznej. Dobierają odpowiedni typ wykresu do prezentowanych danych.</p> <p>Stosują wybrane możliwości arkusza kalkulacyjnego do rozwiązywania zadań z różnych dziedzin.</p>

Multimedia i grafika komputerowa	
Czynności nauczyciela	Czynności uczniów
<p>Porządkuje wiedzę uczniów na temat multimediów, nabytą w gimnazjum i na lekcjach informatyki realizowanej w zakresie podstawowym.</p> <p>Poleca, by uczniowie przypomnieli sobie sposób prezentacji danych liczbowych w komputerze.</p> <p>Omawia sposób prezentowania obrazu i dźwięku w komputerze.</p> <p>Wyjaśnia zasady działania animacji. Poleca utworzenie w dostępnej aplikacji prostej animacji. Wyjaśnia zasady łączenia poszczególnych elementów multimedialnych, np. obrazu z dźwiękiem.</p> <p>Przedstawia ogólne możliwości wybranych programów przeznaczonych do edycji obrazu i dźwięku oraz do tworzenia animacji i obróbki filmów. Zleca samodzielne zapoznanie się z przykładowym programem i opracowanie krótkiego filmu.</p>	<p>Podają przykłady urządzeń multimedialnych. Klasyfikują je pod względem ich przeznaczenia. Wyjaśniają pojęcie <i>komputer multimedialny</i>.</p> <p>Dyskutują nad możliwościami komputera w zakresie edycji obrazu, dźwięku, animacji i wideo. Przypominają odpowiednie treści nauczania z lekcji informatyki realizowanej w zakresie podstawowym.</p> <p>Zapoznają się z modelami barw, głębokością barw oraz formatami plików graficznych i dźwiękowych.</p> <p>Łączą wideo, dźwięk, animację i obraz statyczny, np. w programie do animacji lub programie do obróbki wideo. Montują krótki film.</p>
<p>Omawia podstawy pracy z użyciem wybranego programu graficznego do obróbki grafiki rastrowej.</p> <p>Wyjaśnia, na czym polega selekcja fragmentów obrazu i praca z maskami.</p> <p>Pokazuje na przykładzie, w jaki sposób definiować i przekształcać barwy oraz jak wykonywać przekształcenia geometryczne i stosować filtry.</p> <p>Omawia tworzenie fotomontaży i sposoby wykonywania retuszu.</p> <p>Omawia zasady przygotowania grafiki w celu wykorzystania jej w projektowaniu stron internetowych lub prezentacji multimedialnych.</p> <p>Zwraca uwagę na stosowanie właściwych formatów do zapisu plików graficznych, ze szczególnym uwzględnieniem Internetu.</p>	<p>Przypominają z lekcji informatyki realizowanej w zakresie podstawowym formaty plików graficznych, ich zastosowania, zalety i wady. Wyjaśniają różnice między grafiką rastrową a wektorową.</p> <p>Przypominają możliwości poznanych wcześniej programów graficznych w zakresie obróbki grafiki, zwłaszcza wykonywania przekształceń.</p> <p>Wykonując konkretne ćwiczenia, poznają obróbkę grafiki rastrowej. Stosują różne narzędzia malarskie i korekcyjne oraz wybierają tryb pracy narzędzi.</p> <p>Ćwiczą pracę z maskami (wybierają fragmenty obrazu).</p> <p>Definiują barwy i wykonują różne operacje na barwach.</p> <p>Ćwiczą przekształcenia obrazu (geometryczne i filtry).</p> <p>Tworzą przykładowe fotomontaże. Pracują z warstwami obrazu.</p>
<p>Omawia podstawy pracy z użyciem wybranego programu graficznego do obróbki grafiki wektorowej.</p> <p>Zwraca uwagę na możliwość tworzenia krzywych Béziera.</p>	<p>Zapoznają się z programem do obróbki grafiki wektorowej. Tworzą obrazy, rysują figury, ścieżki. Poznają, czym są krzywe Béziera.</p> <p>Stosują przekształcenia obrazu.</p> <p>Pracują z warstwami obrazu.</p> <p>Przygotowują grafikę do własnej strony internetowej lub prezentacji multimedialnej.</p>

Opracowywanie tekstu i prezentacji multimedialnej	
Czynności nauczyciela	Czynności uczniów
<p>Poleca przypomnienie zasad opracowywania dokumentu tekstowego (w tym wielostronicowego) w edytorach tekstu.</p> <p>Omawia zasady śledzenia zmian w dokumencie.</p> <p>Poleca wykonanie odpowiednich ćwiczeń.</p>	<p>Przypominają zasady opracowywania dokumentów tekstowych w edytorze tekstu.</p> <p>Zapoznają się z możliwościami wykonywania recenzji dokumentu tekstowego.</p> <p>Wykonując ćwiczenia, uczą się, w jaki sposób śledzić zmiany, wstawiać komentarze, porównywać dokumenty.</p>
<p>Poleca przypomnienie wiadomości o zasadach tworzenia prezentacji, poznanych na lekcjach informatyki realizowanej w zakresie podstawowym.</p> <p>Zwraca uwagę na dostosowanie treści i formy do rodzaju prezentacji. Wyjaśnia, czym różni się przygotowanie prezentacji wspomagającej wystąpienie prelegenta od prezentacji do samodzielnego przeglądania przez odbiorcę w dogodnym dla niego czasie lub samouruchamiającej się bez ingerencji odbiorcy.</p> <p>Poleca przygotować tematy do wykonania prezentacji wspomagającej wystąpienie prelegenta.</p> <p>Zaleca przygotowanie scenariusza prezentacji.</p> <p>Omawia zasady prezentowania pokazu slajdów.</p> <p>Proponuje przygotowanie w formie projektu grupowego prezentacji do samodzielnego przeglądania przez odbiorcę.</p> <p>Proponuje przygotowanie w formie projektu grupowego prezentacji samouruchamiającej się.</p> <p>Omawia sposoby zapisywania dokumentu tekstowego i prezentacji multimedialnej w formacie PDF.</p>	<p>Przypominają zasady tworzenia prezentacji multimedialnych. Opcjonalnie wyświetlają własne prezentacje, tworzone podczas zajęć informatyki realizowanej w zakresie podstawowym.</p> <p>Projektują własne prezentacje, które mają wspomagać ich wystąpienia. Dobierają temat. Przygotowują scenariusz, wyszukują i gromadzą materiały (teksty, obrazy, dźwięki i animacje).</p> <p>Planują układ slajdów i ich animację.</p> <p>Przygotowują pokaz slajdów. Odtwarzają prezentację.</p> <p>Poznają metody przygotowania prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</p> <p>Wybierają koordynatora grupy, który przydzieli zadania szczegółowe poszczególnym uczniom.</p> <p>Nagrywają narrację, testują chronometrą i przygotowują hiperłącza.</p> <p>Zapisują dokument tekstowy i prezentację multimedialną w formacie PDF.</p>

4.4. Bazy danych

Projektowanie relacyjnej bazy danych	
Czynności nauczyciela	Czynności uczniów
<p>Poleca uczniom, by przypomnieli sobie z lekcji informatyki realizowanej w zakresie podstawowym pojęcia związane z bazami danych.</p> <p>Na przykładzie gotowej bazy danych wyjaśnia dokładnie, na czym polega przetwarzanie danych i omawia cechy relacyjnej bazy danych.</p> <p>Poleca wykonanie podstawowych operacji na danych (wprowadzanie, modyfikowanie, sortowanie, indeksowanie, wyszukiwanie, prezentacja).</p> <p>Tłumaczy pojęcie relacji w bazach danych. Omawia typy relacji.</p> <p>Wyjaśnia rolę klucza podstawowego.</p> <p>Tłumaczy sposób przygotowania kwerend, zwracając uwagę na wybór odpowiedniego źródła danych dla kwerendy i właściwe zapisywanie kryteriów dla pól.</p> <p>Omawia główne rodzaje formularzy i raportów. Zwraca uwagę na profesjonalne przygotowywanie formularzy.</p> <p>Wyjaśnia sposób importowania i eksportowania danych. Poleca przypomnienie zasad tworzenia korespondencji seryjnej.</p>	<p>Wymieniają podstawowe obszary zastosowań baz danych na przykładach z najbliższego otoczenia: szkoły, zakładów pracy, instytucji naukowych i gospodarczych. Podają przykłady programów do tworzenia baz danych.</p> <p>Wykonując ćwiczenia na gotowych tabelach, przypominają zasady organizacji informacji w bazach danych oraz podstawowe pojęcia: <i>tabela, rekord, pole, typ pola, relacja, klucz podstawowy, formularz, kwerenda, raport</i>.</p> <p>Projektują relacyjną bazę danych, składającą się z trzech tabel połączonych relacją.</p> <p>Definiują klucz podstawowy. Określają relacje. Przygotowują odpowiednie kwerendy wybierające. Projektują formularze. Tworzą formularz z podformularzem.</p> <p>Importują dane z tabel arkusza kalkulacyjnego i dokumentu tekstowego do tabel bazy danych. Eksportują dane z tabel bazy danych do tabel arkusza kalkulacyjnego i do dokumentu tekstowego.</p> <p>Na formularzach umieszczają pola kombi, ograniczają wartości, wstawiają (gdy jest taka potrzeba) bieżącą datę, umieszczają przyciski poleceń i przyciski nawigacyjne.</p> <p>Tworzą kwerendy wybierające. Poznają zastosowanie funkcji standardowych w kwerendach. Stosują standardowe operatory w kryteriach wyszukiwania. Korzystają z parametrów w kwerendzie.</p> <p>Projektują raporty.</p> <p>Wykorzystują dane z bazy do tworzenia korespondencji seryjnej.</p>

Tworzenie kwerend z wykorzystaniem języka SQL	
Czynności nauczyciela	Czynności uczniów
<p>Systematyzuje (wspólnie z uczniami) poznane zagadnienia dotyczące wyszukiwania informacji w różnych bazach danych.</p> <p>Omawia podstawy języka SQL – zasady składni, podstawowe instrukcje.</p> <p>Poleca przejrzanie kodu SQL w kwerendach tworzonych wcześniej, np. za pomocą programu Microsoft Access.</p> <p>Przygotowuje ćwiczenia polegające na wyszukiwaniu różnych informacji z bazy.</p> <p>Poleca dodatkową literaturę oraz nadzoruje pracę uczniów uzdolnionych i zainteresowanych zaawansowanym przetwarzaniem danych za pomocą innych narzędzi.</p>	<p>Wykonują ćwiczenia powtórzeniowe, wyszukując informacje w bazie danych.</p> <p>Tworzą kwerendy z wykorzystaniem języka SQL.</p> <p>Zapoznają się ze składnią i działaniem podstawowych instrukcji.</p> <p>Stosują instrukcję <code>SELECT</code> i jej główne klauzule.</p> <p>Wykorzystują klauzulę <code>JOIN</code> do łączenia informacji z wielu tabel i kwerend oraz przedstawiania wyników jako jednego logicznego połączenia rekordów.</p> <p>Stosują instrukcje <code>INSERT</code> do dopisywania rekordów i <code>UPDATE</code> do modyfikowania rekordów w bazie. Usuwają rekordy, korzystając instrukcji <code>DELETE</code>.</p> <p>Zainteresowani uczniowie projektują bazy zawierające więcej niż trzy tabele. Zapoznają się samodzielnie z zaawansowanym tworzeniem kwerend z wykorzystaniem języka SQL.</p>

Realizacja projektu programistycznego	
Czynności nauczyciela	Czynności uczniów
<p>Omawia, na czym polega praca informatyków (projektantów systemów informatycznych).</p> <p>Wyjaśnia zasady projektowania systemów informatycznych.</p> <p>Szczegółowo prezentuje etapy projektowania.</p> <p>Pilnuje, by każdy z uczniów dokładnie wykonywał swoją część pracy.</p> <p>Nadzoruje przebieg realizacji projektu.</p> <p>Podsumowuje (wspólnie z uczniami) wykonanie projektu.</p> <p>Zwraca uwagę na takie umiejętności uczniów, jak: współdziałanie w zespole, wypowiedzenie własnego zdania, wysłuchiwanie i uwzględnianie poglądów innych, dochodzenie do kompromisu.</p>	<p>Przygotowują się do realizacji projektu grupowego.</p> <p>Planują temat projektu. Realizują go zgodnie z zamierzoną organizacją pracy zespołowej i wytyczonymi wcześniej etapami projektowania.</p> <p>Wybierają koordynatora grupy, który wspólnie z nauczycielem wyznacza zespoły do realizacji poszczególnych zadań.</p> <p>Przeprowadzają analizę systemu informacyjnego, opracowują założenia, wykonują projekt techniczny oraz informatyczny, testują i wdrażają system.</p> <p>Przygotowując założenia w postaci dokumentów edytora tekstu, korzystają z szablonów.</p>

4.5. Algorytmika i programowanie – problemy zaawansowane

Wprowadzanie danych i wyprowadzanie wyników	
Czynności nauczyciela	Czynności uczniów
<p>Systematyzuje wspólnie z uczniami różne sposoby wprowadzania danych i wyprowadzania wyników w wybranym języku programowania (w językach programowania). Poleca przypomnienie klasyfikacji typów danych.</p> <p>Zwraca uwagę na dobieranie typu danych odpowiedniego do realizacji danego algorytmu.</p> <p>Poleca przypomnienie pojęcia <i>stosu</i>. Omawia operacje wykonywane na stosie i zastosowania stosu.</p> <p>Omawia funkcje i procedury umożliwiające generowanie losowych danych.</p> <p>Wyjaśnia, w jaki sposób sprawdzić poprawność danych wprowadzanych przez użytkownika.</p>	<p>Przypominają poznane sposoby wprowadzania danych i wyprowadzania wyników.</p> <p>Wykonując ćwiczenia, uczą się formatowania wyprowadzanych wyników.</p> <p>Tworzą własne typy danych (typ okrojony, typ wycieniowy) i deklarują stałe.</p> <p>Poznają stos jako przykład struktury danych. Wykonują operacje na stosie. Implementują stos z wykorzystaniem tablicy. Poznają zastosowanie stosu (odwrotną notację polską – ONP).</p> <p>Poznają procedury i funkcje umożliwiające generowanie danych losowych. Stosują w programach generowanie losowych danych.</p> <p>Sprawdzają poprawność danych.</p>

Wybrane typy i struktury danych, w tym dynamiczne	
Czynności nauczyciela	Czynności uczniów
<p>Objaśnia, na czym polega przetwarzanie danych typu rekordowego w wybranym języku programowania (w językach programowania). Odnosi się do pojęcia rekordu z baz danych. Omawia na przykładach deklarowanie danych typu rekordowego oraz wczytywanie danych do rekordu.</p> <p>Wyjaśnia sposób przetwarzania plików w wybranym języku programowania (w językach programowania). Poleca zapoznanie się z zasadami korzystania z plików.</p> <p>Wspólnie z uczniami analizuje przykładowe programy umożliwiające wyświetlanie na ekranie zawartości plików i zapisywanie do pliku znaków wprowadzanych z klawiatury.</p> <p>Omawia typy wskaźnikowe i stosowanie struktur dynamicznych w programach (tablicę dynamiczną i listę jednokierunkową).</p> <p>Zainteresowanym uczniom poleca zadania, w których wymagane jest zastosowanie innych dynamicznych struktur danych, np. listy dwukierunkowej.</p>	<p>Tworzą programy, w których deklarują dane typu rekordowego (strukturalnego). Wczytują dane do rekordu.</p> <p>Zapoznają się z zasadami przetwarzania plików w wybranym języku programowania (w językach programowania). Poznają odpowiednie procedury i funkcje. Deklarują zmienne typu plikowego. Odczytują dane z pliku. Zapisują dane do pliku. Obsługują błędy. Przetwarzają pliki tekstowe. Analizują gotowe programy i tworzą własne.</p> <p>Poznają dynamiczne struktury danych i typ wskaźnikowy danych. Deklarują zmienne typu wskaźnikowego. Stosują w programach zmienne wskaźnikowe. Tworzą zmienne dynamiczne.</p> <p>Stosują wybrane struktury dynamiczne w programach. Tworzą listę jednokierunkową. Tworzą i przeglądają binarne drzewo poszukiwań. Wyszukują wartości w binarnym drzewie uporządkowanym.</p> <p>Zainteresowani uczniowie tworzą programy z zastosowaniem innych struktur dynamicznych.</p>

Programowanie modułowe i obiektowe	
Czynności nauczyciela	Czynności uczniów
<p>Poleca przypomnienie klasyfikacji modeli programowania.</p> <p>Dokładniej wyjaśnia, na czym polega programowanie modułowe i w jaki sposób korzysta się ze standardowych modułów wybranego języka programowania (języków programowania). Omawia tworzenie własnych modułów.</p> <p>Omawia istotę programowania zorientowanego obiektowo. Wskazuje różnice w stosunku do programowania strukturalnego.</p> <p>Wyjaśnia pojęcia <i>klasy</i> i <i>obiektu</i>. Definiuje podstawowe pojęcia programowania obiektowego i omawia cechy programowania obiektowego: dziedziczenie i polimorfizm. Omawia na przykładzie, czym są metody wirtualne.</p> <p>Zainteresowanym uczniom poleca dodatkową literaturę dotyczącą zaawansowanych metod programowania i proponuje trudniejsze zadania.</p>	<p>Omawiają modele programowania.</p> <p>Korzystając z przykładowych programów, zapoznają się z zasadami programowania modułowego. Tworzą własne moduły.</p> <p>Poznają podstawowe pojęcia programowania obiektowego: <i>klasa</i>, <i>obiekt</i>, <i>pola</i>, <i>metody</i>. Definiują klasy. Deklarują pola prywatne.</p> <p>Analizując gotowe programy, poznają cechy programowania obiektowego: dziedziczenie i polimorfizm. Dowiadują się, czym są metody wirtualne.</p> <p>Zainteresowani uczniowie tworzą własne programy, stosując poznane zasady programowania modułowego i obiektowego.</p> <p>Wykonują wspólny projekt programistyczny, stosując zasady pracy zespołowej.</p>

Wybrane algorytmy	
Czynności nauczyciela	Czynności uczniów
<p>Poleca powtórzenie zagadnień związanych z zapisem binarnym i szesnastkowym liczb naturalnych.</p> <p>Omawia reprezentację binarną liczb ujemnych oraz reprezentacje: stałopozycyjną i zmiennopozycyjną liczb.</p> <p>Omawia wybrane algorytmy na liczbach naturalnych, algorytmy sortowania, algorytmy numeryczne i badające własności geometryczne.</p> <p>Poleca zadania, w których można zastosować omówione algorytmy.</p> <p>Zwraca uwagę na stosowanie wybranych struktur danych i właściwy ich dobór do rozwiązywanego problemu.</p> <p>Uczniom zainteresowanym poleca zadania z matury, olimpiad i konkursów informatycznych, do których rozwiązania potrzebna jest m.in. znajomość omawianych algorytmów.</p>	<p>Zapoznają się z łamigłówką Wież Hanoi. Wykonują praktyczne ćwiczenie – korzystając z pomocy dydaktycznych, przekładają krążki tak, by rozwiązać łamigłówkę.</p> <p>Poznają algorytmy sortowania: sortowanie przez wstawianie, przez scalanie, metodą szybką. Analizują gotowe listy kroków tych algorytmów, wykonując je dla wybranych danych. Sprawdzają liczbę porównań elementów w przypadku każdego z algorytmów.</p> <p>Zapoznają się z listą kroków algorytmu przeszukiwania binarnego.</p> <p>Uczniowie zainteresowani zapisują algorytmy sortowania oraz przeszukiwania binarnego w postaci programów komputerowych.</p> <p>Zapoznają się z algorytmami zamiany liczb między dowolnymi systemami pozycyjnymi: obliczania wartości dziesiętnej liczby, zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie. Zapisują te algorytmy w postaci programów komputerowych.</p> <p>Poznają przykładowe algorytmy na liczbach naturalnych: generowanie liczb pierwszych (podejście naiwne, sito Eratostenesa), rozkład liczb na czynniki pierwsze, liczby doskonałe.</p> <p>Zapoznają się z algorytmem szybkiego podnoszenia do potęgi i przykładowymi algorytmami badającymi własności geometryczne (przynależność punktu do odcinka, badanie położenia punktu względem prostej).</p> <p>Poznają reprezentację danych numerycznych w komputerze: reprezentację binarną liczb ujemnych, reprezentację stałopozycyjną liczb, reprezentację zmiennopozycyjną liczb.</p> <p>Dowiadują się, jakie są właściwości arytmetyki komputerowej. Na konkretnych przykładach obserwują, jak zmienia się wartość błędu względnego.</p> <p>Poznają przykłady algorytmów numerycznych: wyznaczanie miejsca zerowego funkcji, obliczanie wartości pierwiastka kwadratowego, obliczanie pola obszaru ograniczonego.</p> <p>Tworzą programy komputerowe, stosując poznane algorytmy.</p>

4.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci

Aplikacje bazodanowe dostępne za pośrednictwem Sieci	
Czynności nauczyciela	Czynności uczniów
<p>Poleca powtórzenie materiału o tworzeniu stron internetowych omawianego w gimnazjum i na informatyce realizowanej w zakresie podstawowym, ze szczególnym uwzględnieniem języka HTML.</p> <p>Wyjaśnia, na czym polega technologia dynamicznego generowania stron internetowych – czym różni się strona statyczna od dynamicznej.</p> <p>Przygotowuje odpowiednie narzędzia (instalując odpowiednie oprogramowanie), aby umożliwić dynamiczne przetwarzanie stron, tworzenie skryptów PHP i baz danych w MySQL.</p> <p>Omawia, na czym polega pisanie skryptów w PHP. Poleca napisanie kilku prostych skryptów.</p> <p>Omawia przesyłanie danych za pomocą formularzy (wyświetlanie danych z formularza i odbieranie danych z formularza).</p>	<p>Przypominają podstawy języka znaczników HTML, wykonując nieskomplikowaną stronę na zadany przez nauczyciela temat.</p> <p>Uczniowie, którzy posiadają już własne strony internetowe, prezentują je pozostałym kolegom z grupy. Przedstawiają także narzędzia i metody, jakimi posługiwali się przy ich wykonaniu.</p> <p>Poznają zasady dynamicznego przetwarzania stron.</p> <p>Piszą proste skrypty w języku PHP. Wyświetlają dane instrukcją <code>echo</code>. Stosują kodowanie UTF-8. Stosują zmienne i operatory.</p> <p>Tworzą skrypty przesyłające dane za pomocą formularzy HTML. Piszą kod wyświetlający prosty formularz i odbierający dane z formularza.</p>
<p>Omawia zasady tworzenia sieciowej aplikacji bazodanowej oraz korzystanie z baz danych z poziomu PHP.</p> <p>Wyjaśnia, w jaki sposób utworzyć konta użytkownika i bazy danych na serwerze MySQL. Pokazuje sposób łączenia się z bazą MySQL z poziomu PHP oraz wykonywanie zapytań do bazy danych z poziomu PHP.</p> <p>Poleca utworzenie prostej księgi gości.</p> <p>Zwraca uwagę na dbałość o redakcyjną i merytoryczną poprawność oraz zgodność z prawem umieszczanych w Internecie tekstów i materiałów.</p>	<p>Tworzą witrynę internetową opartą na bazie danych. Korzystają z baz danych z poziomu PHP. Tworzą konta użytkownika i bazy danych na serwerze MySQL. Łączą się z bazą MySQL z poziomu PHP. Wykonują zapytania do bazy danych z poziomu PHP.</p> <p>Tworzą prostą księgę gości: tworzą tabelę na wpisy z księgi gości, dodają wpisy do księgi gości za pomocą instrukcji <code>INSERT</code>, tworzą formularz dodający wpisy do bazy danych, odczytują dane z bazy za pomocą instrukcji <code>SELECT</code> (wyświetlają listy wpisów z księgi gości).</p> <p>Zainteresowani uczniowie tworzą samodzielnie rozbudowaną witrynę internetową opartą na bazach danych.</p>

5. Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne

5.1. Podstawy algorytmiki i programowania

Prezentacja algorytmu liniowego w wybranej notacji				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wie, co to jest algorytm.</p> <p>Określa dane do zadania oraz wyniki.</p> <p>Zna podstawowe zasady graficznego prezentowania algorytmów: podstawowe rodzaje bloków, ich przeznaczenie i sposoby umieszczania w schemacie blokowym.</p> <p>Potrafi narysować (odręcznie) schemat blokowy algorytmu liniowego.</p> <p>Potrafi napisać prosty program, wyświetlający napis na ekranie monitora.</p>	<p>Wymienia przykłady czynności i działań w życiu codziennym oraz zadań szkolnych, które uważa się za algorytmy.</p> <p>Zna pojęcie specyfikacji zadania.</p> <p>Zna wybrane sposoby prezentacji algorytmów.</p> <p>Przedstawia algorytm liniowy w postaci listy kroków.</p> <p>Podczas rysowania schematów blokowych potrafi wykorzystać Autokształty z edytora tekstu.</p> <p>Określa pojęcia <i>program komputerowy, język programowania</i>.</p> <p>Zapisuje prosty algorytm liniowy w wybranym języku programowania. Potrafi go skompilować i uruchomić.</p>	<p>Określa zależności między problemem, algorytmem a programem komputerowym.</p> <p>Potrafi odpowiedzieć na pytanie, czy istnieją działania, które nie mają cech algorytmów, i podać przykłady.</p> <p>Przedstawia dokładną specyfikację dowolnego zadania.</p> <p>Analizuje poprawność budowy schematu blokowego.</p> <p>Wyjaśnia pojęcia: <i>program wynikowy, kompilacja, translacja, interpretacja</i>.</p> <p>Realizuje przykładowy algorytm liniowy w wybranym języku programowania.</p> <p>Wykonuje program i testuje go, podstawiając różne dane.</p>	<p>Zapisuje dowolny algorytm w wybranej przez siebie postaci (notacji).</p> <p>Potrafi samodzielnie zapoznać się z nowym programem edukacyjnym przeznaczonym do konstrukcji schematów blokowych.</p> <p>Potrafi przeprowadzić szczegółową analizę poprawności konstrukcji schematu blokowego.</p> <p>Analizuje działanie algorytmu dla przykładowych danych.</p> <p>Potrafi posłużyć się kompilatorem danego języka.</p> <p>Potrafi wskazać i poprawić błędy w programie.</p>	<p>Przestrzega zasad zapisu algorytmów w zadanej postaci (notacji).</p> <p>Stosuje poznane metody prezentacji algorytmów w opisie zadań (problemów) z innych przedmiotów szkolnych oraz różnych dziedzin życia.</p> <p>Potrafi samodzielnie zapoznać się z kompilatorem wybranego języka programowania.</p> <p>Samodzielnie pisze program realizujący algorytm liniowy.</p>

Podstawowe zasady programowania				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna klasyfikację języków programowania.</p> <p>Zna ogólną budowę programu i najważniejsze elementy języka programowania – słowa kluczowe, instrukcje, wyrażenia, zasady składni.</p> <p>Potrafi zrealizować prosty algorytm liniowy i z warunkami w języku wysokiego poziomu; potrafi skompilować i uruchomić program.</p>	<p>Zapisuje program w czytelnej postaci – stosuje wcięcia, komentarze.</p> <p>Rozumie znaczenie i działanie podstawowych instrukcji (m.in. iteracyjnych, warunkowych) wybranego języka programowania wysokiego poziomu.</p> <p>Wie, na czym polega programowanie strukturalne.</p> <p>Rozróżnia i poprawia błędy kompilacji i błędy wykonania.</p> <p>Potrafi zrealizować algorytmy iteracyjne w języku wysokiego poziomu.</p>	<p>Wymienia i omawia modele programowania.</p> <p>Potrafi prezentować złożone algorytmy (z podprogramami) w wybranym języku programowania.</p> <p>Zna i stosuje instrukcje wyboru.</p> <p>Prezentuje wybrane algorytmy iteracyjne w postaci programu komputerowego.</p> <p>Zna rekurencyjne realizacje prostych algorytmów.</p> <p>Rozumie i stosuje zasady programowania strukturalnego.</p> <p>Deklaruje procedury i funkcje bez parametrów.</p> <p>Wie, na czym polega różnica pomiędzy przekazywaniem parametrów przez zmienną i przez wartość w procedurach i funkcjach.</p> <p>Rozumie zasady postępowania przy rozwiązywaniu problemu metodą zstępującą.</p>	<p>Wie, jaka jest różnica między językiem wysokiego poziomu a językiem wewnętrznym; potrafi określić rolę procesora i pamięci operacyjnej w działaniu programów.</p> <p>Deklaruje procedury i funkcje z parametrami.</p> <p>Wie, jakie znaczenie ma zasięg zmiennej.</p> <p>Definiuje funkcje rekurencyjne. Potrafi prezentować algorytmy rekurencyjne w postaci programu.</p> <p>Zapisuje w postaci programu wybrane algorytmy sortowania, algorytmy na tekstach, definiując odpowiednie procedury lub funkcje.</p>	<p>Ocenia efektywność działania programu.</p> <p>Wskazuje podobieństwa i różnice dotyczące tworzenia programów zapisanych w różnych językach programowania; wyjaśnia działanie poszczególnych instrukcji, sposób deklaracji zmiennych.</p> <p>Sprawnie definiuje i stosuje procedury i funkcje w programach.</p> <p>Sprawnie korzysta z dodatkowej, fachowej literatury.</p> <p>Rozwiązuje przykładowe zadania z matury i olimpiady informatycznej.</p>
<p>Wymienia przykłady prostych struktur danych.</p> <p>Potrafi zadeklarować zmienne typu liczbowego (całkowite, rzeczywiste) i stosować je w zadaniach.</p>	<p>Wie, czym jest zmienna w programie i co oznacza przypisanie jej konkretnej wartości.</p> <p>Rozróżnia struktury danych: proste i złożone. Podaje przykłady.</p>	<p>Potrafi zastosować łańcuchowy i tablicowy typ danych w zadaniach.</p> <p>Deklaruje typ tablicowy i łańcuchowy.</p>	<p>Rozumie, na czym polega dobór struktur danych do algorytmu. Wczytuje i wyprowadza elementy tablicy. Wprowadza dane tekstowe.</p> <p>Tworzy programy, dobierając odpowiednie struktury danych do programu.</p>	<p>Dobiera najlepszy algorytm i odpowiednie struktury danych do rozwiązania postawionego problemu.</p>

Techniki algorytmiczne i wybrane algorytmy				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Określa sytuacje warunkowe.</p> <p>Podaje przykłady zadań, w których występują sytuacje warunkowe.</p> <p>Wie, na czym polega powtarzanie tych samych operacji.</p> <p>Potrafi omówić na konkretnym przykładzie algorytm znajdowania najmniejszego z trzech elementów.</p>	<p>Potrafi odróżnić algorytm liniowy od algorytmu z warunkami (z rozgałęzieniami).</p> <p>Zna pojęcie iteracji i rozumie pojęcie algorytmu iteracyjnego. Podaje ich przykłady.</p> <p>Wie, od czego zależy liczba powtórzeń.</p> <p>Tworzy schemat blokowy algorytmu z warunkiem prostym i pętlą. Testuje rozwiązanie dla wybranych danych.</p> <p>Określa problemy, w których występuje rekurencja i podaje przykłady „zjawisk rekurencyjnych” – wziętych z życia i zadań szkolnych.</p> <p>Potrafi omówić algorytm porządkowania elementów (metodą przez wybór) na praktycznym przykładzie, np. wybierając najwyższego ucznia z grupy.</p> <p>Omawia wybrane algorytmy sortowania.</p> <p>Omawia wybrany algorytm na tekstach (np. tworzenie anagramów).</p>	<p>Analizuje algorytmy, w których występują powtórzenia (iteracje).</p> <p>Ocenia zgodność algorytmu ze specyfikacją.</p> <p>Zna sposoby zakończenia iteracji. Określa kroki iteracji.</p> <p>Potrafi zapisać w wybranej notacji np. algorytm sumowania n liczb, algorytm obliczania silni, znajdowania minimum w ciągu n liczb, algorytm rozwiązywania równania liniowego.</p> <p>Potrafi zapisać algorytm z warunkami zagnieżdżonymi i pętlą w wybranej postaci.</p> <p>Zna iteracyjną postać algorytmu Euklidesa.</p> <p>Zna rekurencyjną realizację wybranego algorytmu, np. silni.</p> <p>Zna przynajmniej dwie techniki sortowania (np. bąbelkowe, przez wybór) i zapisuje wybrany algorytm w postaci programu komputerowego.</p> <p>Omawia wybrane algorytmy na tekstach.</p> <p>Potrafi wyjaśnić, na czym polega wydawanie reszty metodą zachłanną i napisać listę kroków tego algorytmu.</p>	<p>Zapisuje algorytmy z pętlą zagnieżdżoną.</p> <p>Zna metodę „dziel i zwyciężaj”, algorytm generowania liczb Fibonacciego, schemat Hornera. Omawia ich iteracyjną realizację i potrafi przedstawić jeden z nich w wybranej notacji.</p> <p>Zna inne algorytmy sortowania, np. pozycyjne, przez wstawianie.</p> <p>Wskazuje różnicę między rekurencją a iteracją.</p> <p>Zna rekurencyjną realizację wybranych algorytmów, np. obliczania silni i algorytm Euklidesa.</p> <p>Potrafi zamienić algorytm zapisany iteracyjnie na postać rekurencyjną.</p> <p>Zapisuje wybrany algorytm na tekstach (np. tworzenie anagramów, zliczanie znaków w tekście, sprawdzanie, czy dany ciąg jest palindromem) w postaci programu komputerowego.</p> <p>Zapisuje algorytm wydawania reszty metodą zachłanną w postaci programu komputerowego.</p>	<p>Rozumie dokładnie technikę rekurencji (znaczenie stosu).</p> <p>Potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję.</p> <p>Zna trudniejsze algorytmy, np. trwałego małżeństwa, problem ośmiu hetmanów, szukanie wzorca w tekście. Potrafi zapisać je w różnych notacjach (również w języku programowania wysokiego poziomu).</p> <p>Korzysta samodzielnie z dodatkowej literatury fachowej.</p>

Elementy analizy algorytmów				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wymienia własności algorytmów.</p> <p>Potrafi przeanalizować przebieg algorytmu zapisanego w postaci listy kroków lub w postaci schematu blokowego dla przykładowych danych i ocenić w ten sposób jego poprawność.</p>	<p>Zna i omawia własności algorytmów.</p> <p>Potrafi ocenić poprawność działania algorytmu i jego zgodność ze specyfikacją.</p> <p>Określa liczbę prostych działań zawartych w algorytmie.</p>	<p>Rozumie, co to jest złożoność czasowa algorytmu i potrafi określić liczbę operacji wykonywanych na elementach zbioru w wybranym algorytmie sortowania.</p> <p>Rozróżnia złożoność czasową i pamięciową.</p>	<p>Wie, jak ocenić złożoność pamięciową algorytmu.</p> <p>Potrafi porównać złożoność obliczeniową różnych algorytmów tego samego zadania dla tych samych danych.</p> <p>Wie, kiedy algorytm jest efektywny.</p>	<p>Określa złożoność czasową i pamięciową wybranych algorytmów. Zna odpowiednie wzory.</p> <p>Określa efektywność algorytmów.</p>

5.2. Komputer i sieci komputerowe

Reprezentacja danych w komputerze				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna pojęcie systemu pozycyjnego.</p> <p>Wie, co to jest system binarny.</p> <p>Korzystając z przykładów, potrafi obliczyć wartość dziesiętną liczby zapisanej w systemie dwójkowym.</p>	<p>Wyjaśnia, co to jest system binarny, i potrafi dokonać zamiany liczby z systemu dziesiętnego na binarny i odwrotnie.</p>	<p>Zna system szesnastkowy i potrafi wykonać konwersję liczb dziesiętnych na liczby w systemie szesnastkowym i odwrotnie. Zna zależność między systemem binarnym i szesnastkowym.</p>	<p>Potrafi wykonać dowolną konwersję pomiędzy systemem dziesiętnym, dwójkowym i szesnastkowym.</p> <p>Potrafi napisać program obliczający wartość dziesiętną liczby dwójkowej.</p>	<p>Potrafi napisać program (w wersji iteracyjnej i rekurencyjnej) realizujący algorytm zamiany liczby dziesiętnej na postać binarną.</p> <p>Potrafi napisać program realizujący algorytm umożliwiający zamianę liczb z systemu szesnastkowego na dziesiętny i odwrotnie.</p>

Kompresja i szyfrowanie danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wie, co to jest kompresja danych. Zna przynajmniej jeden algorytm kompresji danych.</p> <p>Wie, czym jest szyfrowanie danych. Zna przynajmniej jeden algorytm szyfrowania danych.</p>	<p>Rozumie, na czym polega kompresja danych i w jakim celu się ją wykonuje. Wymienia rodzaje kompresji.</p> <p>Omawia jeden przykładowy algorytm kompresji.</p> <p>Koduje tekst, używając alfabetu Morse'a.</p> <p>Wymienia przykładowe algorytmy szyfrowania. Potrafi zaszyfrować i odszyfrować prosty tekst.</p>	<p>Wie, co to jest współczynnik kompresji.</p> <p>Omawia rodzaje kompresji: kompresję stratną i bezstratną. Podaje przykłady algorytmów kompresji stratnej i bezstratnej.</p> <p>Omawia algorytm statyczny i słownikowy.</p> <p>Omawia przynajmniej dwa algorytmy szyfrowania: szyfr podstawieniowy i przestawieniowy.</p>	<p>Potrafi policzyć współczynnik kompresji.</p> <p>Wyjaśnia różnicę pomiędzy algorytmem statycznym a słownikowym.</p> <p>Stosuje algorytm słownikowy do kompresji ciągu znaków.</p> <p>Omawia przykładowy szyfr z kluczem. Stosuje szyfr Vigenère'a do zaszyfrowania ciągu znaków.</p> <p>Omawia wykorzystanie algorytmów szyfrowania w podpisie elektronicznym.</p>	<p>Wyszukuje dodatkowe informacje na temat kompresji i szyfrowania danych. Omawia inne algorytmy kompresji i szyfrowania. Potrafi zapisać wybrany algorytm kompresji lub szyfrowania w postaci programu. Zapoznaje się samodzielnie z kodem Huffmana i pokazuje na przykładzie jego zastosowanie.</p> <p>Omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat. Wyjaśnia, w jaki sposób tworzy się podpis elektroniczny.</p>

Komputer i system operacyjny				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wymienia części składowe zestawu komputerowego, podaje ich parametry i przeznaczenie.</p> <p>Rozróżnia rodzaje pamięci komputera, określa ich własności i przeznaczenie.</p> <p>Wie, co to jest bit i bajt.</p> <p>Wie, co to jest system operacyjny, wymienia i omawia jego podstawowe funkcje oraz z nich korzysta.</p> <p>Podaje przykłady systemów operacyjnych.</p>	<p>Potrafi sklasyfikować środki (urządzenia) i narzędzia (oprogramowanie) technologii informacyjnej.</p> <p>Wie, jak działa komputer. Wyjaśnia rolę procesora.</p> <p>Rozumie sposób organizacji pamięci komputerowej.</p> <p>Zna jednostki pamięci, pojemności nośników i programów.</p> <p>Omawia dwa przykładowe systemy operacyjne.</p>	<p>Analizuje model komputera zgodny z ideą von Neumanna.</p> <p>Potrafi wymienić i omówić rodzaje aktualnie używanych komputerów.</p> <p>Omawia, jak działa procesor.</p> <p>Wymienia i omawia popularne systemy operacyjne: Microsoft Windows, Unix, Linux, Mac Os.</p>	<p>Omawia szczegółowo model komputera zgodny z ideą von Neumanna.</p> <p>Wyjaśnia, w jaki sposób procesor wykonuje dodawanie liczb.</p> <p>Porównuje cechy różnych systemów operacyjnych, np. Microsoft Windows, Unix, Linux, Mac Os.</p> <p>Omawia przykładowe systemy operacyjne dla urządzeń mobilnych.</p> <p>Samodzielnie zapoznaje się z możliwościami nowych urządzeń związanych z TIK.</p>	<p>Omawia szczegółowo system Linux, porównując go do systemu Microsoft Windows.</p> <p>Korzystając z dodatkowych źródeł, omawia kierunek rozwoju systemów operacyjnych.</p> <p>Korzystając z dodatkowych źródeł, omawia najnowsze osiągnięcia dotyczące systemów operacyjnych stosowanych w urządzeniach mobilnych.</p>

Sieci komputerowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wyjaśnia pojęcia: <i>sieć komputerowa, zasoby sieciowe, klient, serwer</i>.</p> <p>Podaje podział sieci ze względu na wielkość, ze względu na model funkcjonowania i na topologię.</p> <p>Potrafi wymienić kilka cech pracy w sieci, odróżniających ją od pracy na autonomicznym komputerze.</p>	<p>Wymienia korzyści płynące z korzystania z sieci. Wyjaśnia, na czym polega wymiana informacji w sieci.</p> <p>Omawia podstawowe klasy i topologie sieciowe.</p> <p>Potrafi wymienić urządzenia i elementy sieciowe oraz omówić ich ogólne przeznaczenie. Zna cechy systemu działającego w szkolnej pracowni.</p> <p>Wie, co to jest protokół komunikacyjny.</p>	<p>Posługuje się terminologią sieciową.</p> <p>Potrafi wymienić zalety i wady różnych topologii sieci. Charakteryzuje topologie gwiazdy, magistrali i pierścienia.</p> <p>Zna znaczenie protokołu w sieciach (w tym TCP/IP).</p> <p>Definiuje funkcje i usługi poszczególnych warstw modelu warstwowego sieci.</p> <p>Wie, co to jest adres sieciowy. Wyjaśnia, co to jest adres</p>	<p>Swobodnie posługuje się terminologią sieciową.</p> <p>Zna schemat działania sieci komputerowych.</p> <p>Omawia ogólnie określanie ustawień sieciowych danego komputera i jego lokalizacji w sieci (podsieci IPv4, protokół DHCP, DNS, TCP).</p> <p>Wie, co określa maska podsieci.</p> <p>Potrafi z pomocą nauczyciela zrealizować małą sieć komputerową – skonfigurować jej</p>	<p>Omawia szczegółowo model warstwowy sieci.</p> <p>Omawia różne systemy sieciowe. Dokonuje ich analizy porównawczej.</p>

	Zna zasady pracy w sieci, m.in. zasady udostępniania zasobów. Wie, z jakich warstw składa się warstwowy model sieci. Ogólnie je omawia.	domenowy i omawia jego strukturę. Potrafi omówić ogólne zasady administrowania siecią komputerową w architekturze „klient-serwer”.	składniki, udostępnić pliki, dyski, drukarki, dodać użytkowników.	
--	--	---	---	--

Zadania projektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna etapy pracy nad projektem i bierze udział w pracy grupowej jako członek zespołu. Potrafi omówić historię komputerów. Umie wskazać ogólny kierunek zmian w technologiach komputerowych. Zna i stosuje podstawowe zasady netykiety.	Omawia etapy pracy nad projektem i bierze aktywny udział w pracy grupowej jako członek zespołu. Potrafi określić nowoczesne trendy w zastosowaniu urządzeń komputerowych. Omawia wybrane normy etyczne i prawne, m.in.: zasady korzystania z programów komputerowych, rodzaje licencji, rozpowszechnianie programów komputerowych. Omawia społeczne aspekty zastosowania informatyki.	Bierze aktywny udział w pracy grupowej jako członek zespołu, gromadząc i selekcionując materiały do projektu. Potrafi wskazać nowości w zakresie usług internetowych oraz odszukać informacje na temat najnowszych pomysłów na komputery. Omawia wybrane normy etyczne i prawne, m.in. korzystanie z cudzych materiałów. Omawia szanse i zagrożenia związane z rozwojem informatyki i TIK, m.in.: uzależnienie od komputera i Internetu.	Pełni funkcje koordynatora w grupie. Komunikuje się z innymi członkami grupy z wykorzystaniem nowoczesnych technologii, m.in. za pomocą urządzeń mobilnych. Omawia zagadnienia bezpieczeństwa i ochrony danych oraz informacji w komputerze i w sieciach komputerowych (rodzaje zagrożeń, sposoby ochrony). Omawia rozwój informatyki i technologii informacyjno-komunikacyjnych, m.in. najważniejsze elementy procesu rozwoju informatyki i TIK.	Przygotowuje analizę porównawczą, pokazującą na przestrzeni lat rozwój informatyki, w tym sieci komputerowych, oraz multimediiów. Wskazuje tendencje w rozwoju informatyki i jej zastosowań, dostrzegając przeobrażenia w tej dziedzinie w kraju i na świecie. Przygotowuje indywidualny projekt na wybrany przez siebie temat.

5.3. Opracowywanie informacji za pomocą komputera

Algorytmy i zależności funkcyjne w arkuszu kalkulacyjnym				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna zasady tworzenia formuł i stosowania funkcji arkusza kalkulacyjnego.</p> <p>Zna i stosuje zasady adresowania względnego i bezwzględnego w arkuszu kalkulacyjnym.</p> <p>Zapisuje w arkuszu kalkulacyjnym algorytm liniowy i z warunkami.</p> <p>Stosuje wybrane funkcje arkusza kalkulacyjnego.</p> <p>Potrafi utworzyć wykres w arkuszu kalkulacyjnym.</p>	<p>Zapisuje w arkuszu kalkulacyjnym algorytm z warunkami zagnieżdżonymi.</p> <p>Zna i stosuje zasady adresowania mieszanego w arkuszu kalkulacyjnym.</p> <p>Potrafi zrealizować iterację w arkuszu kalkulacyjnym.</p> <p>Rysuje wykres funkcji liniowej i kwadratowej.</p>	<p>Potrafi zrealizować pętlę zagnieżdżoną w arkuszu kalkulacyjnym.</p> <p>Wie, co to jest fraktal i w jaki sposób się go tworzy. Podaje przykłady fraktali.</p> <p>Rysuje wykres wybranej funkcji trygonometrycznej.</p> <p>Dobiera odpowiedni typ wykresu do prezentowanych danych.</p>	<p>Rysuje wykres funkcji liniowej, wielomianu, wybranej funkcji trygonometrycznej i funkcji logarytmicznej.</p> <p>Zna możliwości zastosowania algorytmów iteracyjnych w rysowaniu fraktali, m.in.: śnieżynki Kocha, dywanu i trójkąta Sierpińskiego.</p> <p>Stosuje wybrane możliwości arkusza kalkulacyjnego do rozwiązywania zadań z różnych dziedzin.</p>	<p>Potrafi samodzielnie zobrazować wybraną zależność funkcyjną w arkuszu kalkulacyjnym, np. algorytm rozwiązywania układu równań liniowych metodą wyznaczników.</p> <p>Rozumie, w jaki sposób narysować paprotkę Barnsleya w arkuszu kalkulacyjnym.</p> <p>Korzystając z Internetu i innych źródeł, wyszukuje dodatkowe informacje na temat geometrii fraktalnej, m.in. dotyczące jej zastosowań.</p>

Multimedia i grafika komputerowa				
2	3	4	5	6
Uczeń: Zna urządzenia multimedialne, wymienia przykładowe nazwy, określa ogólnie przeznaczenie urządzeń multimedialnych. Posługuje się drukarką i skanerem. Po zeskanowaniu zapisuje obraz w pliku w formacie domyślnym. Wymienia programy do tworzenia i obróbki grafiki. Posługuje się jednym z nich w celu tworzenia własnych rysunków. Tworzy i edytuje obrazy w wybranym programie graficznym, korzystając z podstawowych narzędzi do edycji obrazu. Wie, czym różni się grafika rastrowa od wektorowej.	Uczeń: Korzysta z różnych urządzeń multimedialnych, zna ich działanie, podaje ich przeznaczenie. Definiuje pojęcie <i>komputer multimedialny</i> . Zna sposoby reprezentacji obrazu i dźwięku w komputerze. Zna modele barw. Zapisuje plik graficzny w różnych formatach. Zna zastosowanie poszczególnych formatów, ich zalety i wady. Wymienia formaty zapisu dźwięku. Zna podstawowe możliwości wybranych programów do edycji obrazu rastrowego i wektorowego. Potrafi wybrać fragmenty obrazu i wykonać na nich różne operacje, np. selekcje, przekształcenia (obroty, odbicia). Potrafi, korzystając z gotowego pliku video, wykonać proste operacje, tj. podzielić film na fragmenty, przyciąć film, dodać efekty.	Uczeń: Posługuje się sprawnie wybranymi urządzeniami multimedialnymi. Zna różne możliwości komputera w zakresie edycji obrazu, dźwięku, animacji i wideo. Omawia model barw. Zna pojęcia: RGB i CMYK. Omawia formaty plików dźwiękowych. Potrafi stosować różne narzędzia malarskie i korekcyjne oraz wybrać odpowiedni tryb ich pracy. Wie, na czym polega praca z warstwami. Wykonuje rysunki, korzystając z warstw. Potrafi tworzyć przykładowe fotomontaże. Tworzy obraz w grafice wektorowej, rysuje figury, ścieżki. Wykonuje przekształcenia obrazu (obroty, odbicia), tworząc obrazy w grafice rastrowej i wektorowej.	Uczeń: Zna i stosuje w praktyce zaawansowaną obróbkę grafiki rastrowej i wektorowej. Przy użyciu odpowiednich narzędzi potrafi zaznaczyć fragmenty obrazu nawet o skomplikowanym kształcie. Potrafi zapisywać pliki multimedialne w różnych formatach, ze szczególnym uwzględnieniem formatów internetowych. Zna pojęcia: <i>filtr</i> , <i>histogram</i> , <i>krzywa barw</i> . Potrafi zdefiniować barwy i wykonać na nich operacje. Stosuje filtry. Wykonuje ćwiczenia z maskami (wybiera fragmenty obrazu). Potrafi retuszować obraz. Wie, czym są krzywe Béziera i rysuje je. Opracowuje samodzielnie krótki film.	Uczeń: Samodzielnie zapoznaje się z programami do obróbki grafiki rastrowej i wektorowej. Tworzy obrazy, wykorzystując różne możliwości programów. Korzystając z Pomocy i innych źródeł, poznaje możliwości programów graficznych. Przygotowuje grafikę do własnej strony internetowej lub prezentacji multimedialnej. Uczestniczy w konkursach dotyczących grafiki komputerowej.

Opracowywanie tekstu i prezentacji multimedialnej				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna i stosuje podstawowe zasady redagowania i formatowania tekstu.</p>	<p>Przygotowuje poprawnie zredagowany i sformatowany tekst, dostosowując formę tekstu do jego przeznaczenia.</p>	<p>Tworzy dokumenty tekstowe, stosując poprawnie wszystkie poznane zasady redagowania i formatowania tekstu.</p> <p>Zna możliwości śledzenia zmian w dokumencie tekstowym. Potrafi śledzić zmiany w dokumencie tekstowym</p> <p>Zapisuje dokument tekstowy w formacie PDF.</p>	<p>Tworzy wielostronicowe dokumenty tekstowe, stosując poprawnie wszystkie poznane zasady pracy z tekstem wielostronicowym.</p> <p>Potrafi korzystać z możliwości śledzenia zmian w dokumencie, wstawiać komentarze, porównywać dokumenty.</p>	<p>Samodzielnie odkrywa nowe możliwości edytora tekstu, przygotowując dokumenty tekstowe.</p> <p>Korzysta z możliwości śledzenia zmian w dokumencie, pracując w grupie kilku osób nad jednym dokumentem.</p>
<p>Tworzy prezentację składającą się z kilku slajdów. Wstawia teksty i obrazy, stosuje animacje.</p> <p>Zna ogólne zasady tworzenia prezentacji.</p> <p>Potrafi dobrać tło, atrybuty czcionek, odpowiednio rozmieścić tekst i grafikę na slajdzie.</p>	<p>Rozróżnia sposoby przygotowania prezentacji wspomagającej wystąpienie prelegenta oraz prezentacji typu kiosk.</p> <p>Wie, na czym polega dostosowanie treści i formy do rodzaju prezentacji.</p> <p>Zna i stosuje poprawne zasady tworzenia prezentacji wspomagającej wystąpienie prelegenta.</p> <p>Potrafi zaprojektować prezentację wspomagającą własne wystąpienie. Posługuje się szablonem projektu.</p> <p>Wyszukuje i gromadzi gotowe materiały (teksty i obrazy, dźwięk). Komponuje układ slajdów i ich animację.</p> <p>Stosuje zasady prezentowania pokazu slajdów.</p>	<p>Zna i stosuje metody projektowania różnych rodzajów prezentacji.</p> <p>Potrafi zaprojektować prezentację wspomagającą własne wystąpienie.</p> <p>Wybiera temat, przygotowuje scenariusz, wyszukuje oraz tworzy własne materiały (teksty i obrazy, dźwięk). Komponuje układ slajdów i ich animacje.</p> <p>Posługuje się widokiem sortowania slajdów.</p> <p>Stosuje zasady referowania konkretnego tematu wspomaganej prezentacją.</p> <p>Zna i stosuje zasady przygotowania prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</p> <p>Dodaje efekty multimedialne:</p>	<p>Na gotowym, poprawnie wykonanym przykładzie przedstawia zasady tworzenia prezentacji multimedialnych.</p> <p>Wyjaśnia, na czym polega dostosowanie treści i formy do rodzaju prezentacji.</p> <p>Wyjaśnia różnice w zasadach projektowania prezentacji wspomagającej wystąpienie prelegenta, prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</p> <p>Potrafi organizować prezentację w widoku konspektu.</p> <p>Wykorzystuje możliwości tworzenia schematu organizacyjnego oraz możliwości tworzenia wykresów.</p> <p>Nagrywa narrację.</p> <p>Poprawnie ustawia i testuje</p>	<p>Przygotowuje profesjonalnie prezentacje dowolnego typu.</p> <p>Potrafi, korzystając z prezentacji wspomagającej wystąpienie prelegenta, przeprowadzić profesjonalny pokaz.</p> <p>Dodaje do prezentacji materiały ze skanera, aparatu cyfrowego i kamery cyfrowej.</p> <p>Publikuje prezentację w Internecie. Dopasowuje parametry konwersji do formatu HTML.</p> <p>Przygotowuje materiały ułatwiające opracowanie prezentacji, np. wydruk miniaturki slajdów wraz z notatkami.</p> <p>Nagrywa narrację i dodaje ją do prezentacji.</p>

	<p>Zna zasady przygotowania prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</p>	<p>animacje, grafikę, dźwięki, podkład muzyczny. Ustawia i testuje chronometraż. Stosuje hiperłącza.</p> <p>Zapisuje prezentację w formacie PDF.</p> <p>Aktywnie współpracuje z grupą przy projektowaniu prezentacji.</p>	<p>chronometraż, stosuje hiperłącza.</p> <p>Uczestniczy w przygotowaniu w formie projektów grupowych: prezentacji do samodzielnego przeglądania przez odbiorcę oraz samouruchamiającej się.</p> <p>Zna sposoby umieszczania prezentacji w Internecie.</p>	
--	---	---	---	--

5.4. Bazy danych

Projektowanie relacyjnej bazy danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Podaje obszary zastosowań baz danych – na przykładach z najbliższego otoczenia – szkoły, instytucji naukowych, społecznych i gospodarczych.</p> <p>Podaje przykłady programów do tworzenia baz danych.</p> <p>Potrafi wykonać podstawowe operacje na bazie danych przygotowanej w jednej tabeli (wprowadzanie, redagowanie, sortowanie, wyszukiwanie, prezentacja).</p> <p>Potrafi utworzyć prostą kwerendę, jeden formularz i raport.</p>	<p>Rozumie metody przetwarzania danych na przykładzie gotowej bazy danych.</p> <p>Określa podstawowe pojęcia (<i>rekord, pole, typ pola, relacja, klucz podstawowy</i>).</p> <p>Tworzy bazę danych składającą się z dwóch tabel, w każdej po kilka pól różnych typów.</p> <p>Projektuje formularze i raporty.</p> <p>Tworzy proste kwerendy wybierające.</p> <p>Potrafi wykonywać operacje przetwarzania danych w bazie składającej się z kilku rekordów.</p> <p>Zna zasady przygotowania korespondencji seryjnej.</p>	<p>Projektuje relacyjną bazę danych (na zadany temat) składającą się z trzech tabel połączonych relacją.</p> <p>Omawia typy relacji w bazie danych.</p> <p>Zna zasady definiowania kluczy podstawowych.</p> <p>Projektuje formularze i raporty według wskazówek nauczyciela.</p> <p>Potrafi utworzyć formularz z podformularzem. Umieszcza przyciski nawigacyjne.</p> <p>Tworzy kwerendy wybierające.</p> <p>Importuje dane z tabel arkusza kalkulacyjnego i dokumentu tekstowego do tabel bazy danych.</p> <p>Eksportuje dane z tabel bazy danych do tabel arkusza kalkulacyjnego i do dokumentu tekstowego.</p>	<p>Potrafi wytłumaczyć pojęcie relacji.</p> <p>Projektuje relacyjną bazę danych składającą się z trzech lub większej liczby tabel.</p> <p>Samodzielnie ustala zawartość bazy (rodzaj informacji).</p> <p>Zna kilka rodzajów formularzy i raportów.</p> <p>Umie zaprojektować samodzielnie wygląd formularza i raportu.</p> <p>Na formularzach umieszcza pola kombi, ogranicza wartości, wstawia (gdy jest taka potrzeba) bieżącą datę, umieszcza przyciski poleceń.</p> <p>Stosuje funkcje standardowe w kwerendach i standardowe operatory w kryteriach wyszukiwania. Korzysta z parametrów w kwerendzie.</p>	<p>Zna dokładnie wybrany program do projektowania baz danych.</p> <p>Potrafi samodzielnie zaprojektować bazę danych, korzystając z wybranego narzędzia (programu). Projekt bazy opiera na rzeczywistych informacjach, aby można było wykorzystać ją w praktyce, np. w szkole czy w domu.</p> <p>Korzysta z dodatkowej, fachowej literatury.</p>

Tworzenie kwerend z wykorzystaniem języka SQL				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wyszukuje informacje w bazie, korzystając wyłącznie z gotowych kwerend i narzędzi wbudowanych do programu.	<p>Tworzy samodzielnie kwerendy (proste i złożone), korzystając z wbudowanych do programu narzędzi.</p> <p>Wie, co to jest język SQL. Potrafi przeanalizować przykład zapytania utworzonego w języku SQL.</p>	<p>Zna zasady wyszukiwania informacji w bazie z wykorzystaniem języka zapytań.</p> <p>Zna składnię i działanie podstawowych instrukcji.</p> <p>Potrafi zapisać prostą kwerendę, korzystając z języka zapytań.</p>	<p>Potrafi zapisać złożone kwerendy, korzystając z wybranej instrukcji.</p> <p>Stosuje instrukcję <code>SELECT</code> i jej główne klauzule, by wybrać kolumny z tabel bazy danych.</p> <p>Wykorzystuje klauzulę <code>JOIN</code> do łączenia informacji z wielu tabel i kwerend oraz przedstawiania wyników jako jednego logicznego połączenia rekordów.</p> <p>Stosuje instrukcje <code>INSERT</code> do dopisywania rekordów i <code>UPDATE</code> do modyfikowania rekordów w bazie. Usuwa rekordy, korzystając z instrukcji <code>DELETE</code>.</p>	Opierając się na profesjonalnej literaturze, potrafi samodzielnie zapisywać złożone kwerendy z wykorzystaniem języka zapytań SQL.

Realizacja projektu programistycznego				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Uczestniczy czynnie w projekcie grupowym, wykonując proste zadania, np. wprowadza dane do bazy i je aktualizuje.</p> <p>Bierze udział w testowaniu projektu.</p>	<p>Zna wszystkie etapy projektowania systemów informatycznych.</p> <p>Uczestniczy czynnie w poszczególnych etapach projektu, wykonując zleczone zadania szczegółowe.</p> <p>Planuje temat projektu.</p>	<p>Wie, co to jest system informatyczny. Potrafi omówić zakres prac na każdym etapie.</p> <p>Realizuje projekt na zadany (lub samodzielnie wybrany) temat zgodnie z etapami projektowania.</p> <p>Uczestniczy czynnie w analizie systemu informacyjnego, przygotowuje dokumentację.</p> <p>Przygotowuje założenia w postaci dokumentów edytora tekstu, korzystając z szablonów.</p> <p>Współpracuje przy projektowaniu tabel, formularzy i raportów.</p>	<p>Realizuje projekt zgodnie z zamierzoną organizacją pracy zespołowej i wytyczonymi wcześniej etapami projektowania.</p> <p>Wykonuje trudniejsze prace związane z projektowaniem systemu.</p> <p>Przeprowadza analizę systemu informacyjnego.</p> <p>Projektuje złożone kwerendy, formularze, raporty.</p> <p>Uczestniczy we wdrażaniu systemu informatycznego.</p>	<p>Potrafi pełnić funkcję koordynatora projektu.</p> <p>Przydziela zadania szczegółowe, dba o ich prawidłowe wykonanie, nadzoruje pracę innych, dba o dobrą atmosferę w grupie.</p>

5.5. Algorytmika i programowanie – problemy zaawansowane

Wprowadzanie danych i wyprowadzanie wyników				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna i stosuje podstawowe sposoby wprowadzania danych i wyprowadzania wyników w wybranym języku programowania.	Potrafi poprawnie zadeklarować dane potrzebne do rozwiązania zadania. Wie, co określa typ danych i format danych. Potrafi odpowiednio sformatować wyprowadzane wyniki.	Tworzy własne typy danych (typ okrojony, typ wyczerpujący) i stałe. Zna procedury i funkcje umożliwiające generowanie danych losowych. Stosuje w programach generowanie losowych danych.	Poprawnie formatuje wyprowadzane wyniki. Omawia stos jako przykład struktury danych. Wykonuje operacje na stosie. Implementuje stos z wykorzystaniem tablicy. Zna i omawia zastosowanie stosu (odwrotną notację polską – ONP).	Dobiera sposób wprowadzania danych do rozwiązywanego zadania. Potrafi sprawdzić poprawność danych wprowadzanych do programu. Definiuje własne typy danych, potrzebne do rozwiązania danego zadania. Korzystając z możliwości generowania danych losowych, tworzy własne gry komputerowe.

Wybrane typy i struktury danych, w tym dynamiczne				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna i omawia strukturę typów danych w wybranym języku programowania. Wie, jakiego typu dane zapisuje się w rekordzie. Wie, czym charakteryzują się dane typu tablicowego.	Tworzy prosty program, w którym deklaruje dane typu rekordowego (strukturalnego). Wczytuje dane do rekordu. Potrafi omówić ogólnie zasady przetwarzania plików w wybranym języku programowania.	Potrafi zastosować rekordowy typ danych do przetwarzania danych różnego rodzaju. Zna zasady przetwarzania plików w wybranym języku programowania (w językach programowania). Stosuje odpowiednie procedury i funkcje. Deklaruje zmienne typu plikowego. Korzystając z przykładów, odczytuje dane z pliku i zapisuje dane do pliku.	Wykonuje operacje na plikach w wybranym języku programowania (w językach programowania). Przetwarza pliki tekstowe. Tworzy własne programy, w których wykorzystuje przetwarzanie plików. Zna dynamiczne struktury danych i typ wskaźnikowy danych. Deklaruje zmienne typu wskaźnikowego. Stosuje w programach zmienne wskaźnikowe. Tworzy zmienne	Stosuje w programach wybrane struktury dynamiczne. Tworzy listę jednokierunkową. Tworzy i przegląda binarne drzewo poszukiwań. Pisze program wyszukujący wartości w binarnym drzewie uporządkowanym. Korzystając z dodatkowej literatury, zapoznaje się z innymi strukturami dynamicznymi, np. z listą dwukierunkową. Tworzy programy z zastosowaniem struktur

			<p>dynamiczne.</p> <p>Zna wybrane struktury dynamiczne.</p> <p>Analizuje gotowe programy, w których zastosowano listę jednokierunkową i binarne drzewo poszukiwań; uruchamia je i testuje dla wybranych danych.</p>	<p>dynamicznych.</p> <p>Rozwiązuje zadania z matury i olimpiady informatycznej i bierze w niej udział.</p>
--	--	--	---	--

Programowanie modułowe i obiektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wymienia modele programowania.</p> <p>Omawia model programowania strukturalnego. Wie, jakie są korzyści z definiowania procedur i funkcji.</p> <p>Potrafi zdefiniować procedury i funkcje w wybranym języku programowania.</p> <p>Wie, co to jest moduł.</p>	<p>Omawia szczegółowo modele programowania.</p> <p>Zna zasady programowania modułowego.</p> <p>Analizując przykładowe programy, tworzy własne moduły.</p>	<p>Wie, na czym polega programowanie obiektowe.</p> <p>Zna podstawowe pojęcia programowania obiektowego: <i>klasa, obiekt, pola, metody</i>.</p>	<p>Rozumie i potrafi zastosować typ obiektowy.</p> <p>Definiuje klasy. Deklaruje pola prywatne.</p> <p>Omawia cechy programowania obiektowego: dziedziczenie i polimorfizm. Wie, czym są metody wirtualne. Poznaje zastosowanie tych cech, analizując gotowe programy. Modyfikuje programy według wskazówek nauczyciela.</p>	<p>Tworzy własne programy, stosując poznane zasady programowania modułowego i obiektowego.</p> <p>Korzysta z fachowej literatury.</p> <p>Wie, na czym polega programowanie zdarzeniowe.</p> <p>Rozwiązuje zadania z olimpiady informatycznej i bierze w niej udział.</p>

Wybrane algorytmy				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna i potrafi omówić wybrane algorytmy sortowania.</p> <p>Analizuje gotowe listy kroków wybranych algorytmów sortowania, wykonuje algorytmy dla wybranych danych.</p> <p>Potrafi omówić algorytm rozkładu liczby na czynniki pierwsze.</p> <p>Zna łamigłówkę Wież Hanoi. Wykonuje praktyczne ćwiczenie, odpowiednio przekłada trzy krążki.</p>	<p>Zna jeden z wybranych algorytmów sortowania: np. przez wstawianie. Sprawdza liczbę porównań elementów w tym algorytmie.</p> <p>Zna algorytmy zamiany liczb między dowolnymi systemami pozycyjnymi: obliczania wartości dziesiętnej liczby, zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie.</p> <p>Wie, jak sprawdzić, czy dana liczba jest liczbą doskonałą.</p> <p>Zna reprezentację danych numerycznych w komputerze: reprezentację binarną liczb ujemnych, reprezentację stałopozycyjną liczb, reprezentację zmiennopozycyjną liczb.</p>	<p>Zna algorytmy sortowania: np. przez wstawianie, przez scalanie, metodą szybką. Sprawdza liczbę porównań elementów w przypadku każdego z algorytmów.</p> <p>Zna algorytm przeszukiwania binarnego. Potrafi utworzyć listę kroków tego algorytmu.</p> <p>Zna przykładowe algorytmy na liczbach naturalnych: generowanie liczb pierwszych (podejście naiwne, sito Eratostenesa).</p> <p>Omawia wybrany algorytm numeryczny, np. obliczanie wartości pierwiastka kwadratowego.</p> <p>Zna sposób zapisu liczby całkowitej i rzeczywistej (zmiennoprzecinkowej).</p>	<p>Omawia i stosuje w zadaniach algorytmy na liczbach naturalnych: generowanie liczb pierwszych (podejście naiwne, sito Eratostenesa).</p> <p>Omawia algorytm szybkiego podnoszenia do potęgi i algorytmy badające własności geometryczne (np. przynależność punktu do odcinka, badanie położenia punktu względem prostej).</p> <p>Omawia wybrany algorytm numeryczny, np. wyznaczanie miejsca zerowego funkcji, obliczanie wartości pierwiastka kwadratowego, obliczanie pola obszaru ograniczonego.</p> <p>Zna właściwości arytmetyki komputerowej. Na konkretnych przykładach potrafi sprawdzić, jak zmienia się wartość błędu względnego.</p> <p>Tworzy program komputerowy, stosując wybrany algorytm.</p> <p>Wykonuje projekt programistyczny, stosując zasady pracy zespołowej.</p>	<p>Zapisuje wybrane algorytmy sortowania (np. przez wstawianie, przez scalanie, metodą szybką) w postaci programu komputerowego.</p> <p>Potrafi zapisać w języku programowania wysokiego poziomu algorytm konwersji liczb z dowolnego systemu pozycyjnego na inny.</p> <p>Tworzy programy komputerowe, stosując wybrane algorytmy.</p> <p>Rozwiązuje zadania z matury i olimpiady informatycznej.</p>

5.6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci

Aplikacje bazodanowe dostępne za pośrednictwem Sieci				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Wymienia przykładowe programy do projektowania i tworzenia stron internetowych.</p> <p>Potrafi wymienić podstawowe elementy, z których składa się strona WWW.</p> <p>W stopniu podstawowym posługuje się wybranym programem do tworzenia stron.</p> <p>Zna podstawowe znaczniki języka HTML.</p> <p>Tworzy nieskomplikowaną stronę internetową. Wstawia tytuł, formatuje tekst, umieszcza obraz.</p> <p>Wie, na czym polega technologia dynamicznego generowania stron internetowych.</p>	<p>Potrafi omówić strukturę pliku w języku HTML.</p> <p>Zna podstawy języka znaczników HTML i potrafi wykonać prostą stronę na zadany przez nauczyciela temat. Projektuje wygląd strony. Planuje jej zawartość (teksty, rysunki, dźwięki, animacje) i umieszcza na niej ww. elementy.</p> <p>Zna zasady dynamicznego przetwarzania stron internetowych.</p> <p>Potrafi napisać proste skrypty w języku PHP.</p>	<p>Potrafi samodzielnie zaprojektować wygląd strony.</p> <p>Zna reguły poprawnego projektowania układu strony, m.in. dba o jej czytelność i przejrzystość, o poprawność redakcyjną, i merytoryczną oraz prawidłową umieszczanych na niej tekstów i materiałów.</p> <p>Zna zaawansowane możliwości języka HTML: tabele, ramki, style.</p> <p>Zna sposoby publikowania stron w Internecie.</p> <p>Tworzy skrypty w języku PHP. Wyświetla dane instrukcją echo. Stosuje kodowanie UTF-8. Stosuje zmienne i operatory.</p> <p>Wie, jak utworzyć witrynę internetową opartą na bazie danych, m.in. w jaki sposób utworzyć prostą księgę gości w MySQL.</p>	<p>Potrafi samodzielnie wykorzystać poznane funkcje języka HTML do udoskonalenia własnych stron internetowych</p> <p>Włącza licznik odwiedzin na stronie. Dodaje inne typowe elementy: forum, księgę gości.</p> <p>Zna podstawy języka skryptowego PHP (lub JavaScript). Używa go dla osiągnięcia nieskomplikowanych efektów wizualnych na stronie.</p> <p>Tworzy skrypty przesyłające dane za pomocą formularzy HTML. Pisze kod wyświetlający prosty formularz i odbierający dane z formularza.</p> <p>Tworzy witrynę internetową opartą na bazie danych. Tworzy konta użytkownika i bazy danych na serwerze MySQL. Wykonuje zapytania do bazy danych z poziomu PHP.</p> <p>Tworzy prostą księgę gości: tworzy tabelę na wpisy z księgi gości, dodaje wpisy do księgi gości za pomocą instrukcji INSERT, tworzy formularz dodający wpisy do bazy danych, odczytuje dane z bazy za pomocą instrukcji SELECT.</p>	<p>Potrafi samodzielnie zapoznać się z nowym programem do tworzenia stron internetowych.</p> <p>Potrafi posługiwać się językiem skryptowym PHP (lub JavaScript) do tworzenia stron dynamicznych.</p> <p>Tworzy samodzielnie rozbudowaną witrynę internetową opartą na bazach danych.</p>

6. Propozycje metod sprawdzania osiągnięć ucznia

Należy tak organizować pracę na lekcji (przygotowywać odpowiednie ćwiczenia), aby również uczniowie mniej zaawansowani mieli szansę zdobycia najwyższej oceny i nie czuli się dyskryminowani. Wszystkim uczniom należy zapewnić sprawiedliwy sposób oceniania. Należy zwracać szczególną uwagę na postępy w zdobywaniu wiedzy.

Dla uczniów zainteresowanych należy przygotować zadania trudniejsze (np. z matury i olimpiady informatycznej), aby mogli wykazać się swoimi umiejętnościami i wiedzą, biorąc udział w konkursach i olimpiadach informatycznych.

Uczniom mającym trudności z wykonywaniem ćwiczeń i poleceń należy pomagać bezpośrednio podczas zajęć, udzielać wyjaśnień i dodatkowych wskazówek, motywować ich i zachęcać do dalszej pracy.

Należy uczniów wcześniej poinformować o wymaganiach na poszczególne oceny.

Uczniowie powinni otrzymywać dużo ocen cząstkowych. Należy oceniać np.:

- ćwiczenia wykonywane podczas lekcji,
- odpowiedzi na pytania,
- udział w projekcie grupowym,
- zadania domowe,
- aktywność na lekcji,
- ćwiczenia sprawdzające.

Ocena ćwiczeń wykonywanych podczas lekcji

W trakcie zajęć nauczyciel powinien zwrócić szczególną uwagę na sposób wykonywania ćwiczeń przez uczniów, korzystanie z podręcznika oraz wyjaśnień nauczyciela, ewentualnie pomocy innego ucznia. Nie należy tolerować wykonywania ćwiczeń przez „sąsiada”.

Należy obserwować, czy działania podejmowane przez uczniów w celu rozwiązania zadania wynikają z wiedzy na dany temat i nabytych umiejętności, czy są to działania świadome oraz czy uczeń wykonuje wszystkie czynności planowo i nie działa chaotycznie lub przypadkowo.

Można premiować uczniów, którzy wykonają zadanie całkowicie samodzielnie i poprawnie.

Gdy wykonywane ćwiczenie ma być podsumowaniem większego działu, w ocenie należy uwzględnić opanowanie wszystkich umiejętności przewidzianych w programie dla danego tematu.

Podczas wykonywania ćwiczeń zasadne jest ocenienie na tej samej lekcji wszystkich uczniów.

Ocena odpowiedzi na pytania

W trakcie wykonywania przez uczniów ćwiczeń można zadawać pytania o zastosowaną metodę lub sposób otrzymania danego rozwiązania. Należy zwrócić uwagę na sposób formułowania odpowiedzi: czy uczeń posługuje się słownictwem potocznym, czy też używa określeń fachowych i rozumie ich znaczenie.

Ocena udziału w projekcie grupowym

Wykonanie przez uczniów projektu grupowego pozwala na sprawdzenie i ocenę nabytych kompetencji oraz podsumowanie omówionych treści nauczania. W pracy grupowej każdy uczeń powinien być oceniany za wykonanie częściowego zadania, składającego się na cały projekt. Trzeba uwzględnić jego wkład pracy, zaangażowanie i umiejętność pracy w zespole.

Ocena zadań domowych

Uczniowie powinni mieć systematycznie zadawane zadania domowe, np. udzielanie odpowiedzi na pytania i wykonywanie zadań dotyczących tematu omawianego na lekcji, samodzielne zapoznanie się z danym tematem lub jego częścią. Zadania domowe należy regularnie sprawdzać, na każdej lekcji wybranym uczniom lub całej grupie.

Ocena aktywności na lekcji

Uczeń poprawnie odpowiadający na lekcji na pytania dodatkowe, wykonujący dodatkowe ćwiczenia, powinien być za to oceniany.

Ocena zadań sprawdzających

Zadania sprawdzające powinny być bardzo precyzyjnie określone i dokładnie przygotowane (np. w postaci wypunktowanych poleceń), w formie zrozumiałej dla ucznia i ułatwiającej jednoznaczną ocenę. Forma zadań nie powinna odbiegać od ćwiczeń, które uczniowie wykonują podczas zajęć. Należy unikać tzw. zaliczania przedmiotu pod koniec półrocza czy roku szkolnego.

W ocenie ćwiczenia należy uwzględnić wykonanie wszystkich poleceń zgodnie z treścią. Warto opracować odpowiednią punktację za wykonanie każdego polecenia.