

INFORMATYKA

dla szkół ponadgimnazjalnych.
Zakres rozszerzony

wymagania na poszczególne oceny

4.1. 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.</p> <p>Potrafi go skompilować i uruchomić.</p>	<p>Określa zależności między problemem, algorytmem a programem komputerowym. Potrafi odpowiedzieć na pytanie, czy istnieją działania, które nie mają cech algorytmów, i podać przykłady.</p> <p>Przedstawia dokładną specyfikację dowolnego zadania. 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. 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.</p> <p>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.</p> <p>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: Określa sytuacje warunkowe. Podaje przykłady zadań, w których występują sytuacje warunkowe. Wie, na czym polegają powtarzanie tych samych operacji. Potrafi omówić na konkretnym przykładzie algorytm znajdowania najmniejszego z trzech elementów.	Uczeń: Potrafi odróżnić algorytm liniowy od algorytmu z warunkami (z rozgałęzieniami). Zna pojęcie iteracji i rozumie pojęcie algorytmu iteracyjnego. Podaje ich przykłady. Wie, od czego zależy liczba powtórzeń. Tworzy schemat blokowy algorytmu z warunkiem prostym i pętlą. Testuje rozwiązanie dla wybranych danych. Określa problemy, w których występuje rekurencja i podaje przykłady „zjawisk rekurencyjnych” - wziętych z życia i zadań szkolnych. Potrafi omówić algorytm porządkowania elementów (metodą przez wybór) na praktycznym przykładzie, np. wybierając najwyższego ucznia z grupy. Omawia wybrane algorytmy sortowania. Omawia wybrany algorytm na tekstach (np. tworzenie anagramów).	Uczeń: Analizuje algorytmy, w których występują powtórzenia (iteracje). Ocenia zgodność algorytmu ze specyfikacją. Zna sposoby zakończenia iteracji. Określa kroki iteracji. Potrafi zapisać w wybranej notacji np. algorytm sumowania n liczb, algorytm obliczania silni, znajdowania minimum w ciągu n liczb, algorytm rozwiązywania równania liniowego. Potrafi zapisać algorytm z warunkami zagnieżdżonymi i pętlą w wybranej postaci. Zna iteracyjną postać algorytmu Euklidesa. Zna rekurencyjną realizację wybranego algorytmu, np. silni. Zna przynajmniej dwie techniki sortowania (np. bąbelkowe, przez wybór) i zapisuje wybrany algorytm w postaci programu komputerowego. Omawia wybrane algorytmy na tekstach. Potrafi wyjaśnić, na czym polega wydawanie reszty metodą zachłanną i napisać listę kroków tego algorytmu.	Uczeń: Zapisuje algorytmy z pętlą zagnieżdżoną. Zna metodę „dziel i zwyciężaj”, algorytm generowania liczb Fibonacciego, schemat Hornera. Omawia ich iteracyjną realizację i potrafi przedstawić jeden z nich w wybranej notacji. Zna inne algorytmy sortowania, np. pozycyjne, przez wstawianie. Wskazuje różnicę między rekurencją a iteracją. Zna rekurencyjną realizację wybranych algorytmów, np. obliczania silni i algorytm Euklidesa. Potrafi zamienić algorytm zapisany iteracyjnie na postać rekurencyjną. Zapisuje wybrany algorytm na tekstach (np. tworzenie anagramów, zliczanie znaków w tekście, sprawdzanie, czy dany ciąg jest palindromem) w postaci programu komputerowego. Zapisuje algorytm wydawania reszty metodą zachłanną w postaci programu komputerowego.	Uczeń: Rozumie dokładnie technikę rekurencji (znaczenie stosu). Potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję. 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). Korzysta samodzielnie z dodatkowej literatury fachowej.

Elementy analizy algorytmów

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

4.2. Komputer i sieci komputerowe

Reprezentacja danych w komputerze				
2	3	4	5	6
Uczeń: Zna pojęcie systemu pozycyjnego. Wie, co to jest system binarny. Korzystając z przykładów, potrafi obliczyć wartość dziesiętną liczby zapisanej w systemie dwójkowym.	Uczeń: Wyjaśnia, co to jest system binarny, i potrafi dokonać zamiany liczby z systemu dziesiętnego na binarny i odwrotnie.	Uczeń: Zna system szesnastkowy i potrafi wykonać konwersję liczb dziesiętnych na liczby w systemie szesnastkowym i odwrotnie. Zna zależność między systemem binarnym i szesnastkowym.	Uczeń: Potrafi wykonać dowolną konwersję pomiędzy systemem dziesiętnym, dwójkowym i szesnastkowym. Potrafi napisać program obliczający wartość dziesiętną liczby dwójkowej.	Uczeń: Potrafi napisać program (w wersji iteracyjnej i rekurencyjnej) realizujący algorytm zamiany liczby dziesiętnej na postać binarną. Potrafi napisać program realizujący algorytm umożliwiający zamianę liczb z systemu szesnastkowego na dziesiętny i odwrotnie.
Kompresja i szyfrowanie danych				
2	3	4	5	6
Uczeń: Wie, co to jest kompresja danych. Zna przynajmniej jeden algorytm kompresji danych. Wie, czym jest szyfrowanie danych. Zna przynajmniej jeden algorytm szyfrowania danych.	Uczeń: Rozumie, na czym polega kompresja danych i w jakim celu się ją wykonuje. Wymienia rodzaje kompresji. Omawia jeden przykładowy algorytm kompresji. Koduje tekst, używając alfabetu Morse'a. Wymienia przykładowe algorytmy szyfrowania. Potrafi zaszyfrować i odszyfrować prosty tekst.	Uczeń: Wie, co to jest współczynnik kompresji. Omawia rodzaje kompresji: kompresję stratną i bezstratną. Podaje przykłady algorytmów kompresji stratnej i bezstratnej. Omawia algorytm statyczny i słownikowy. Omawia przynajmniej dwa algorytmy szyfrowania: szyfr podstawieniowy i przestawieniowy.	Uczeń: Potrafi policzyć współczynnik kompresji. Wyjaśnia różnicę pomiędzy algorytmem statycznym a słownikowym. Stosuje algorytm słownikowy do kompresji ciągu znaków. Omawia przykładowy szyfr z kluczem. Stosuje szyfr Vigenere'a do zaszyfrowania ciągu znaków. Omawia wykorzystanie algorytmów szyfrowania w podpisie elektronicznym.	Uczeń: 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. Omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat. Wyjaśnia, w jaki sposób tworzy się podpis elektroniczny.

Komputer i system operacyjny				
2	3	4	5	6
Uczeń: Wymienia części składowe zestawu komputerowego, podaje ich parametry i przeznaczenie. Rozróżnia rodzaje pamięci komputera, określa ich własności i przeznaczenie. Wie, co to jest bit i bajt. Wie, co to jest system operacyjny, wymienia i omawia jego podstawowe funkcje oraz z nich korzysta. Podaje przykłady systemów operacyjnych.	Uczeń: Potrafi sklasyfikować środki (urządzenia) i narzędzia (oprogramowanie) technologii informacyjnej. Wie, jak działa komputer. Wyjaśnia rolę procesora. Rozumie sposób organizacji pamięci komputerowej. Zna jednostki pamięci, pojemności nośników i programów. Omawia dwa przykładowe systemy operacyjne.	Uczeń: Analizuje model komputera zgodny z ideą von Neumanna. Potrafi wymienić i omówić rodzaje aktualnie używanych komputerów. Omawia, jak działa procesor. Wymienia i omawia popularne systemy operacyjne: Microsoft Windows, Unix, Linux, Mac Os.	Uczeń: Omawia szczegółowo model komputera zgodny z ideą von Neumanna. Wyjaśnia, w jaki sposób procesor wykonuje dodawanie liczb. Porównuje cechy różnych systemów operacyjnych, np. Microsoft Windows, Unix, Linux, Mac Os. Omawia przykładowe systemy operacyjne dla urządzeń mobilnych. Samodzielnie zapoznaje się z możliwościami nowych urządzeń związanych z TIK.	Uczeń: Omawia szczegółowo system Linux, porównując go do systemu Microsoft Windows. Korzystając z dodatkowych źródeł, omawia kierunek rozwoju systemów operacyjnych. Korzystając z dodatkowych źródeł, omawia najnowsze osiągnięcia dotyczące systemów operacyjnych stosowanych w urządzeniach mobilnych.
Sieci komputerowe				
2	3	4	5	6
Uczeń: Wyjaśnia pojęcia: <i>sieć komputerowa, zasoby sieciowe, klient, serwer</i> . Podaje podział sieci ze względu na wielkość, ze względu na model funkcjonowania i na topologię. Potrafi wymienić kilka cech pracy w sieci, odróżniających ją od pracy na autonomicznym komputerze.	Uczeń: Wymienia korzyści płynące z korzystania z sieci. Wyjaśnia, na czym polega wymiana informacji w sieci. Omawia podstawowe klasy i topologie sieciowe. Potrafi wymienić urządzenia i elementy sieciowe oraz omówić ich ogólne przeznaczenie. Zna cechy systemu działającego w szkolnej pracowni. Wie, co to jest protokół komunikacyjny. Zna zasady pracy w sieci, m.in. zasady udostępniania zasobów. Wie, z jakich warstw składa się warstwowy model sieci. Ogólnie je omawia.	Uczeń: Postępuje się terminologią sieciową. Potrafi wymienić zalety i wady różnych topologii sieci. Charakteryzuje topologie gwiazdy, magistrali i pierścienia. Zna znaczenie protokołu w sieciach (w tym TCP/IP). Definiuje funkcje i usługi poszczególnych warstw modelu warstwowego sieci. Wie, co to jest adres sieciowy. Wyjaśnia, co to jest adres domenowy i omawia jego strukturę. Potrafi omówić ogólne zasady administrowania siecią komputerową w architekturze „klient-serwer”.	Uczeń: Swobodnie posługuje się terminologią sieciową. Zna schemat działania sieci komputerowych. Omawia ogólnie określanie ustawień sieciowych danego komputera i jego lokalizacji w sieci (podsieci IPv4, protokół DHCP, DNS, TCP). Wie, co określa maska podsieci. Potrafi z pomocą nauczyciela zrealizować małą sieć komputerową - skonfigurować jej składniki, udostępnić pliki, dyski, drukarki, dodać użytkowników.	Uczeń: Omawia szczegółowo model warstwowy sieci. Omawia różne systemy sieciowe. Dokonuje ich analizy porównawczej.

Zadania projektowe				
2	3	4	5	6
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.	Uczeń: 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.	Uczeń: Bierze aktywny udział w pracy grupowej jako członek zespołu, gromadząc i selekcjonując 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.	Uczeń: 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.	Uczeń: 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.

4.3. Opracowywanie informacji za pomocą komputera

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

Multimedia i grafika komputerowa

2	3	4	5	6
<p>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.</p>	<p>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.</p>	<p>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.</p>	<p>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 Beziery i rysuje je. Opracowuje samodzielnie krótki film.</p>	<p>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.</p>

Opracowywanie tekstu i prezentacji multimedialnej

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna i stosuje podstawowe zasady redagowania i formatowania tekstu.	Przygotowuje poprawnie zredagowany i sformatowany tekst, dostosowując formę tekstu do jego przeznaczenia.	<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. 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 zasady przygotowania prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</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. Posługuje się widokiem sortowania slajdów.</p> <p>Stosuje zasady referowania konkretnego tematu wspomaganego prezentacją.</p> <p>Zna i stosuje zasady przygotowania prezentacji do samodzielnego przeglądania przez odbiorcę oraz prezentacji samouruchamiającej się.</p> <p>Dodaje efekty multimedialne: 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>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 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>	<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.</p> <p>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>

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

Realizacja projektu programistycznego

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

4.5. Algorytmika i programowanie - problemy zaawansowane

Wprowadzanie danych i wyprowadzanie wyników				
2	3	4	5	6
Uczeń: Zna i stosuje podstawowe sposoby wprowadzania danych i wyprowadzania wyników w wybranym języku programowania.	Uczeń: Potrafi poprawnie zadeklarować dane potrzebne do rozwiązania zadania. Wie, co określa typ danych i format danych. Potrafi odpowiednio sformatować wyprowadzane wyniki.	Uczeń: Tworzy własne typy danych (typ okrojony, typ wyliczeniowy) i stałe. Zna procedury i funkcje umożliwiające generowanie danych losowych. Stosuje w programach generowanie losowych danych.	Uczeń: 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).	Uczeń: 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: 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.	Uczeń: Tworzy prosty program, w którym deklaruje dane typu rekordowego (strukturalnego). Wczytuje dane do rekordu. Potrafi omówić ogólnie zasady przetwarzania plików w wybranym języku programowania.	Uczeń: 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.	Uczeń: 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 dynamiczne. Zna wybrane struktury dynamiczne. Analizuje gotowe programy, w których zastosowano listę jednokierunkową i binarne drzewo poszukiwań; uruchamia je i testuje dla wybranych danych.	Uczeń: 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 dynamicznych. Rozwiązuje zadania z matury i olimpiady informatycznej i bierze w niej udział.

Programowanie modułarne i obiektowe

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

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.</p> <p>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. wyznaczenie 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>

4.6. 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 prawną 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 książki gości, dodaje wpisy do książki 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. Potrafi posługiwać się językiem skryptowym PHP (lub JavaScript) do tworzenia stron dynamicznych.</p> <p>Tworzy samodzielnie rozbudowaną witrynę internetową opartą na bazach danych.</p>