

Informatyka inżynierska

Tabela wymagań na poszczególne oceny

Spis treści

1. Podstawy algorytmiki i programowania	2
2. Komputer i sieci komputerowe	6
3. Opracowywanie informacji za pomocą komputera	9
4. Bazy danych	11
5. Algorytmika i programowanie – problemy zaawansowane	14
6. Aplikacje bazodanowe dostępne za pośrednictwem Sieci	17

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>Ocenią 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>

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.</p> <p>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 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.	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.

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:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
<p>Zna urządzenia multimedialne, wymienia przykładowe nazwy, określa ogólnie przeznaczenie urządzeń multimedialnych.</p> <p>Wymienia programy do tworzenia i obróbki grafiki. Posługuje się jednym z nich w celu tworzenia własnych rysunków.</p> <p>Tworzy i edytuje obrazy w wybranym programie graficznym, korzystając z podstawowych narzędzi do edycji obrazu.</p> <p>Wie, czym różni się grafika rastrowa od wektorowej.</p>	<p>Korzysta z różnych urządzeń multimedialnych, zna ich działanie, podaje ich przeznaczenie.</p> <p>Zna sposoby reprezentacji obrazu i dźwięku w komputerze.</p> <p>Zna modele barw.</p> <p>Zapisuje plik graficzny w różnych formatach. Zna zastosowanie poszczególnych formatów, ich zalety i wady.</p> <p>Wymienia formaty zapisu dźwięku.</p> <p>Zna podstawowe możliwości wybranych programów do edycji obrazu.</p> <p>Potrafi wybrać fragmenty obrazu i wykonać na nich różne operacje, np. selekcje, przekształcenia (obroty, odbicia).</p> <p>Potrafi, korzystając z gotowego pliku video, wykonać proste operacje, tj. podzielić film na fragmenty, przyciąć film, dodać efekty.</p> <p>Potrafi wstawiać i modyfikować tekst do ruchomego obrazu.</p> <p>Potrafi wstawić pola formularza.</p>	<p>Posługuje się sprawnie wybranymi urządzeniami multimedialnymi.</p> <p>Zna różne możliwości komputera w zakresie edycji obrazu, dźwięku, animacji i wideo.</p> <p>Omawia model barw. Zna pojęcia: RGB i CMYK.</p> <p>Omawia formaty plików dźwiękowych.</p> <p>Potrafi stosować różne narzędzia graficzne oraz wybrać odpowiedni tryb ich pracy.</p> <p>Wie, na czym polega praca z warstwami. Wykonuje rysunki, korzystając z warstw.</p> <p>Potrafi tworzyć animacje poklatkowe oraz z wykorzystaniem gotowych efektów.</p> <p>Tworzy obraz w grafice wektorowej, rysuje figury, ścieżki.</p> <p>Wykonuje przekształcenia obrazu (obroty, odbicia), tworząc obrazy w grafice wektorowej.</p> <p>Potrafi wstawiać obiekty do biblioteki i tworzyć instancje.</p> <p>Potrafi napisać prosty skrypt.</p>	<p>Zna i stosuje w praktyce zaawansowaną obróbkę grafiki rastrowej i wektorowej.</p> <p>Przy użyciu odpowiednich narzędzi potrafi zaznaczyć fragmenty obrazu nawet o skomplikowanym kształcie.</p> <p>Potrafi zapisywać pliki multimedialne w różnych formatach, ze szczególnym uwzględnieniem formatów internetowych.</p> <p>Potrafi samodzielnie wykonać animację po ścieżce ruchu.</p> <p>Wykonuje ćwiczenia z maskami (wybiera fragmenty obrazu).</p> <p>Wie, czym są krzywe Béziera i rysuje je.</p> <p>Opracowuje samodzielnie krótki film.</p> <p>Potrafi napisać skrypt umożliwiający sterowanie animacją za pomocą myszy i klawiatury.</p> <p>Umie wykorzystać dane z formularzy.</p>	<p>Samodzielnie zapoznaje się z programami do obróbki grafiki rastrowej i wektorowej.</p> <p>Tworzy obrazy, wykorzystując różne możliwości programów.</p> <p>Korzystając z Pomocy i innych źródeł, poznaje możliwości programów graficznych.</p> <p>Przygotowuje grafikę do własnej strony internetowej lub prezentacji multimedialnej.</p> <p>Uczestniczy w konkursach dotyczących grafiki komputerowej.</p> <p>Samodzielnie tworzy zaawansowane skrypty.</p> <p>Projektuje i wykonuje gry z wykorzystaniem technologii flash.</p>

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. 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 zlecone 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. 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 wycięziowy) 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. Pozna je 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>

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 umieszczonych 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>