

Przedmiotowy system oceniania z informatyki

Klasa III i IV Technikum

I. Zasady ogólne

1. Przedmiotowy System Oceniania (PSO) jest zgodny z Wewnątrzszkolnym Systemem Oceniania (WSO), który stanowi załącznik do Statutu Szkoły.
2. W ramach oceniania przedmiotowego nauczyciel rozpoznaje poziom i postępy w opanowaniu przez ucznia wiadomości i umiejętności w stosunku do wymagań edukacyjnych wynikających z podstawy programowej danego etapu edukacyjnego i realizowanego przez nauczyciela programu nauczania uwzględniającego tą podstawę.
3. Oceniani podlegają osiągnięcia edukacyjne ucznia, tj. stan wiedzy i umiejętności uczniów oraz postępy czynione przez ucznia.
4. O zakresie wymagań edukacyjnych, kryteriach i sposobach oceniania oraz trybie poprawiania oceny nauczyciel informuje uczniów na pierwszej lekcji informatyki.
5. Wymagania edukacyjne są dostosowane do indywidualnych potrzeb rozwojowych i edukacyjnych oraz możliwości psychofizycznych ucznia (m.in. na podstawie orzeczeń, opinii ppp oraz w wyniku rozpoznania indywidualnych potrzeb przez pracowników placówki).
6. Oceny cząstkowe wyraża się w skali od 1 do 6, można stosować znaki + i -. Ocenę śródroczną i roczną wyraża się w sześciostopniowej skali - od 1 do 6.
7. Główną funkcją oceniania bieżącego jest monitorowanie pracy ucznia i przekazywanie mu informacji o jego osiągnięciach edukacyjnych pomagających w uczeniu się, poprzez wskazanie, co uczeń robi dobrze, co i jak wymaga poprawy oraz jak powinien dalej się uczyć.
Ocenianie bieżące ma za zadanie umożliwić:
 - a) informowanie ucznia, rodzica i nauczyciela o poziomie osiągnięć edukacyjnych oraz postępach ucznia,
 - b) udzielanie uczniowi pomocy w nauce poprzez przekazanie mu informacji o tym, co zrobił dobrze i jak powinien się dalej uczyć;
 - c) wskazywanie uczniowi mocnych (uzdolnień) i słabych stron, a przede wszystkim sposobów pracy nad nimi,
 - d) planowanie rozwoju ucznia, rozwijania jego uzdolnień, pokonywania ewentualnych trudności,
 - e) motywowanie ucznia do dalszych postępów w nauce.
8. Ustalenie śródrocznej i rocznej oceny klasyfikacyjnej odbywa się w trybie ustalonym w WSO.
9. Jeśli wynik klasyfikacji śródrocznej ucznia wskazuje na to, że poziom osiągnięć edukacyjnych ucznia uniemożliwi bądź utrudni mu kontynuowanie nauki w klasie programowo wyższej, szkoła umożliwia uczniowi uzupełnienie braków w następujący sposób:
 - a) szczegółowe przedstawienie przez nauczyciela przedmiotu braków, wskazanie treści, które są niezbędne do opanowania przez ucznia,
 - b) oferta dodatkowych zadań i ćwiczeń pozwalających na przewyżczenie trudności,
 - c) konsultacje indywidualne z nauczycielem przedmiotu.
10. Wszystkie oceny są dla ucznia i jego rodziców jawne, a sprawdzone i ocenione pisemne prace ucznia są udostępniane na zasadach określonych w WSO.
11. Oceny podlegają uzasadnieniu przez nauczyciela (w sposób określony w Statucie szkoły).

II. Sposoby sprawdzania osiągnięć edukacyjnych

1. Nauczyciel sprawdza osiągnięcia edukacyjne ucznia możliwie często. Im większa liczba ocen cząstkowych, tym mniejszy błąd pomiaru, którym są obarczone powszechnie stosowane testy nauczycielskie.
2. Do sprawdzania wiedzy, umiejętności i postępów edukacyjnych ucznia stosuje się takie narzędzia jak: obserwacja ucznia w trakcie zajęć edukacyjnych – udział ucznia w zajęciach, udział w ćwiczeniach, testy, sprawdziany, prace pisemne, kartkówki, wypowiedzi ustne, prace domowe.
3. Uzyskane oceny są jawne, podlegają uzasadnieniu, a ocenione prace pisemne wglądowi.
4. Każdą oceną można poprawić w trybie określonym w WSO. Poprawa pracy jest dobrowolna.
5. Sprawdziany i ich zakres są zapowiadane z co najmniej tygodniowym wyprzedzeniem, kartkówki z bieżącego materiału nie podlegają tej zasadzie.
6. Sprawdziany, kartkówki i prace pisemne zapowiadane przez nauczyciela są obowiązkowe.
7. O terminach i zakresie prac domowych nauczyciel informuje na bieżąco.
8. Uczeń może być nieprzygotowany do zajęć z przyczyn losowych i jest zobowiązany zgłosić ten fakt nauczycielowi na początku lekcji (przez nieprzygotowanie do lekcji rozumiane jest: nieprzygotowanie do kartkówki, brak pracy domowej, brak ćwiczeń).
9. Uczeń ma prawo do uzyskania pomocy nauczyciela w nadrobieniu zaległości wynikających z długotrwałej nieobecności w szkole. Termin nadrobienia zaległości podlega indywidualnym ustaleniom (adekwatnym do długości i przyczyny nieobecności).

10. Udział i osiągnięcia w konkursach przedmiotowych mogą spowodować podniesienie oceny śródrocznej lub rocznej.
11. W przypadku prac pisemnych punktowanych stosuje się następującą skalę :
- celujący 100% maksymalnej liczby punktów
 - bardzo dobry 90-99% maksymalnej liczby punktów
 - dobry 75-89% maksymalnej liczby punktów
 - dostateczny 51-74% maksymalnej liczby punktów
 - dopuszczający 31-50% maksymalnej liczby punktów
 - niedostateczny 0-30% maksymalnej liczby punktów
12. Ocena końcoworoczna zostaje ustalona zgodnie z WSO.
13. Wyodrębnia się następujące poziomy wymagań edukacyjnych:

WIADOMOŚCI np. znajomość definicji, pojęć informatycznych, itp.	Poziom I Kategoria A	ZAPAMIĘTYWANIE – uczeń nazywa, definiuje, wylicza, wymienia, zna
	Poziom I Kategoria B	ZROZUMIENIE – uczeń rozumie, rozróżnia, streszcza, wyjaśnia, rozróżnia, ilustruje
UMIĘJĘTNOŚCI np. analiza problemów, łączenie teorii z praktyką, itp.	Poziom II Kategoria C	ZASTOSOWANIE INFORMACJI W SYTUACJACH TYPOWYCH – uczeń reaguje, stosuje, informuje, odpowiada, pyta itd
	Poziom II Kategoria D	ZASTOSOWANIE INFORMACJI W SYTUACJACH PROBLEMOWYCH – proponuje, planuje, ocenia, przekonuje, argumentuje itp.

Dla uzyskania poszczególnych stopni, uczeń powinien wykazać się odpowiednią wiedzą i umiejętnościami.

III. Wymagania edukacyjne niezbędne do uzyskania poszczególnych ocen

Kryteria oceniania ogólne

Ocena	Wymagania
Dopuszczający	Znajomość podstawowych wiadomości i najważniejszych pojęć informatycznych, uczeń ma trudności w posługiwaniu się komputerem, wykonuje ćwiczenia z pomocą nauczyciela.
Dostateczny	Opanowanie przewidzianego materiału w stopniu zadawalającym, przeciętna znajomość omawianych narzędzi pracy informatyka, uczeń bez problemu radzi sobie z podstawowymi ćwiczeniami i problemami, rozwiązywanie trudniejszych ćwiczeń z pomocą nauczyciela.
Dobry	Uczeń dysponuje wiedzą wymaganą w programie, jest aktywny w czasie lekcji, rozwiązuje samodzielnie zadania o podwyższonym stopniu trudności, analizuje napotkane problemy, potrafi łączyć teorię z praktyką.
Bardzo dobry	Pełna znajomość materiału z programu nauczania, aktywne uczestnictwo w lekcji, umiejętność samodzielnego korzystania z dodatkowej literatury, uczeń bardzo dobrze zna wykorzystywane w czasie lekcji środowiska, samodzielnie analizuje i rozwiązuje problemy, uczestniczy w szkolnych i poza szkolnych konkursach.
Celujący	Uczeń w wysokim stopniu opanował wiedzę i umiejętności określone programem nauczania, biegła znajomość narzędzi, programów, systemów, rozwiązywanie skomplikowanych problemów, uczeń uczestniczy i osiąga sukcesy w szkolnych i poza szkolnych olimpiadach i konkursach.

Uczeń otrzymuje ocenę *niedostateczną*, jeśli nie spełnia kryteriów na ocenę dopuszczającą, czyli nie ma opanował podstawowej wiedzy i umiejętności określonych w nowej podstawie programowej, a braki w wiadomościach i umiejętnościach uniemożliwiają dalszą naukę, oraz nie potrafi wykonać zadań o elementarnym stopniu trudności.

Wymagania edukacyjne w klasie 3

Tematyka zajęć	Wymagania na ocenę dopuszczającą. Uczeń:	Wymagania na ocenę dostateczną. Uczeń:	Wymagania na ocenę dobrą. Uczeń:	Wymagania na ocenę bardzo dobrą. Uczeń:	Wymagania na ocenę celującą. Uczeń:
Techniki algorytmiczne i wybrane algorytmy	<p>Określa sytuacje warunkowe. 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	<p>Wymienia własności algorytmów.</p> <p>Potrafi przeanalizować przebieg algorytmu zapisanego w postaci listy kroków lub w postaci</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</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</p>	<p>Wie, jak ocenić złożoność pamięciową algorytmu.</p> <p>Potrafi porównać złożoność obliczeniową różnych algorytmów tego samego zadania dla tych samych</p>	<p>Określa złożoność czasową i pamięciową wybranych algorytmów. Zna odpowiednie wzory.</p> <p>Określa efektywność algorytmów.</p>

	schematu blokowego dla przykładowych danych i ocenić w ten sposób jego poprawność.	zawartych w algorytmie.	i pamięciową.	danych. Wie, kiedy algorytm jest efektywny.	
Wybrane algorytmy	<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. 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>

Wymagania edukacyjne w klasie 4

Tematyka	Wymagania na ocenę dopuszczającą. Uczeń:	Wymagania na ocenę dostateczną. Uczeń:	Wymagania na ocenę dobrą. Uczeń:	Wymagania na ocenę bardzo dobrą. Uczeń:	Wymagania na ocenę celującą. Uczeń:
Szyfrowanie danych Certyfikaty Podpis elektroniczny	- wie, na czym polega szyfrowanie danych - umie odczytać zakodowany tekst na podstawie opisanej zasady dekodowania i klucza	- rozumie pojęcie certyfikatu publicznego - opisuje jego zastosowanie - zna pojęcie klucz publiczny	- opisuje różnice pomiędzy kluczem publicznym symetrycznym a asymetrycznym i zna schemat ich użycia - zabezpiecza pliki tekstowe w czasie ich zachowywania na dysku	- wie, czym jest podpis elektroniczny i w jakich okolicznościach się go stosuje - definiuje certyfikat klucza publicznego	- opisuje sposób zdobycia podpisu elektronicznego i przypadki, w których może on być stosowany
Projektowanie relacyjnej bazy danych	Podaje obszary zastosowań baz danych – na przykładach z najbliższego otoczenia – szkoły, instytucji naukowych, społecznych i gospodarczych. Podaje przykłady programów do tworzenia baz danych. Potrafi wykonać podstawowe operacje na bazie danych przygotowanej w jednej tabeli (wprowadzanie, redagowanie, sortowanie, wyszukiwanie, prezentacja). Potrafi utworzyć prostą kwerendę, jeden formularz i raport.	Rozumie metody przetwarzania danych na przykładzie gotowej bazy danych. Określa podstawowe pojęcia (<i>rekord, pole, typ pola, relacja, klucz podstawowy</i>). Tworzy bazę danych składającą się z dwóch tabel, w każdej po kilka pól różnych typów. Projektuje formularze i raporty. Tworzy proste kwerendy wybierające. Potrafi wykonywać operacje przetwarzania danych w bazie składającej się z kilku rekordów. Zna zasady przygotowania korespondencji seryjnej.	Projektuje relacyjną bazę danych (na zadany temat) składającą się z trzech tabel połączonych relacją. Omawia typy relacji w bazie danych. Zna zasady definiowania kluczy podstawowych. Projektuje formularze i raporty według wskazówek nauczyciela. Potrafi utworzyć formularz z podformularzem. Umieszcza przyciski nawigacyjne. Tworzy kwerendy wybierające. Importuje dane z tabel arkusza kalkulacyjnego i dokumentu tekstowego do tabel bazy danych. Eksportuje dane z tabel bazy danych do tabel arkusza kalkulacyjnego i do dokumentu tekstowego.	Potrafi wytłumaczyć pojęcie relacji. Projektuje relacyjną bazę danych składającą się z trzech lub większej liczby tabel. Samodzielnie ustala zawartość bazy (rodzaj informacji). Zna kilka rodzajów formularzy i raportów. Umie zaprojektować samodzielnie wygląd formularza i raportu. Na formularzach umieszcza pola kombi, ogranicza wartości, wstawia (gdzie jest taka potrzeba) bieżącą datę, umieszcza przyciski poleceń. Stosuje funkcje standardowe w kwerendach i standardowe operatory w kryteriach wyszukiwania. Korzysta z parametrów w kwerendzie.	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. Korzysta z dodatkowej, fachowej literatury.
Tworzenie kwerend z wykorzystaniem języka SQL	Wyszukuje informacje w bazie, korzystając wyłącznie z gotowych kwerend i narzędzi wbudowanych do programu.	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.	Zna zasady wyszukiwania informacji w bazie z wykorzystaniem języka zapytań. Zna składnię i działanie podstawowych instrukcji. Potrafi zapisać prostą kwerendę, korzystając z języka zapytań.	Potrafi zapisać złożone kwerendy, korzystając z wybranej instrukcji. Stosuje instrukcję SELECT i jej główne klauzule, by wybrać kolumny z tabel bazy danych. Wykorzystuje klauzulę JOIN do łączenia informacji z wielu tabel i kwerend oraz przedstawiania wyników jako jednego logicznego połączenia rekordów. Stosuje instrukcje INSERT do dopisywania rekordów i UPDATE do	Opierając się na profesjonalnej literaturze, potrafi samodzielnie zapisywać złożone kwerendy z wykorzystaniem języka zapytań SQL.

				modyfikowania rekordów w bazie. Usuwa rekordy, korzystając z instrukcji DELETE.	
Komputer i system operacyjny	<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.</p> <p>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	<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.</p> <p>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>Zna zasady pracy w sieci, m.in. zasady udostępniania zasobów.</p> <p>Wie, z jakich warstw składa się warstwowy model sieci. Ogólnie je omawia.</p>	<p>Postępuje 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.</p> <p>Wyjaśnia, co to jest adres domenowy i omawia jego strukturę.</p> <p>Potrafi omówić ogólne zasady administrowania siecią komputerową w architekturze „klient-serwer”.</p>	<p>Swobodnie postępuje 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 składniki, udostępnić pliki, dyski, drukarki, dodać użytkowników.</p>	<p>Omawia szczegółowo model warstwowy sieci.</p> <p>Omawia różne systemy sieciowe. Dokonuje ich analizy porównawczej.</p>
Algorytmy i zależności funkcyjne w arkuszu kalkulacyjnym	<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</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>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>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>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</p>

	<p>kalkulacyjnym algorytm liniowy i z warunkami.</p> <p>Stosuje wybrane funkcje arkusza kalkulacyjnego.</p> <p>Potrafi utworzyć wykres w arkuszu kalkulacyjnym.</p>	<p>Rysuje wykres funkcji liniowej i kwadratowej.</p>	<p>Dobiera odpowiedni typ wykresu do prezentowanych danych.</p>	<p>Stosuje wybrane możliwości arkusza kalkulacyjnego do rozwiązywania zadań z różnych dziedzin.</p>	<p>kalkulacyjnym.</p> <p>Korzystając z Internetu i innych źródeł, wyszukuje dodatkowe informacje na temat geometrii fraktalnej, m.in. dotyczące jej zastosowań.</p>
<p>Wybrane typy i struktury danych, w tym dynamiczne</p>	<p>Zna i omawia strukturę typów danych w wybranym języku programowania.</p> <p>Wie, jakiego typu dane zapisuje się w rekordzie.</p> <p>Wie, czym charakteryzują się dane typu tablicowego.</p>	<p>Tworzy prosty program, w którym deklaruje dane typu rekordowego (strukturalnego).</p> <p>Wczytuje dane do rekordu.</p> <p>Potrafi omówić ogólnie zasady przetwarzania plików w wybranym języku programowania.</p>	<p>Potrafi zastosować rekordowy typ danych do przetwarzania danych różnego rodzaju.</p> <p>Zna zasady przetwarzania plików w wybranym języku programowania (w językach programowania). Stosuje odpowiednie procedury i funkcje.</p> <p>Deklaruje zmienne typu plikowego.</p> <p>Korzystając z przykładów, odczytuje dane z pliku i zapisuje dane do pliku.</p>	<p>Wykonuje operacje na plikach w wybranym języku programowania (w językach programowania).</p> <p>Przetwarza pliki tekstowe.</p> <p>Tworzy własne programy, w których wykorzystuje przetwarzanie plików.</p> <p>Zna dynamiczne struktury danych i typ wskaźnikowy danych.</p> <p>Deklaruje zmienne typu wskaźnikowego.</p> <p>Stosuje w programach zmienne wskaźnikowe. Tworzy zmienne 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>Stosuje w programach wybrane struktury dynamiczne. Tworzy listę jednokierunkową.</p> <p>Tworzy i przegląda binarne drzewo poszukiwań.</p> <p>Pisze program wyszukujący wartości w binarnym drzewie uporządkowanym.</p> <p>Korzystając z dodatkowej literatury, zapoznaje się z innymi strukturami dynamicznymi, np. z listą dwukierunkową.</p> <p>Tworzy programy z zastosowaniem struktur dynamicznych.</p> <p>Rozwiązuje zadania z matury i olimpiady informatycznej i bierze w niej udział.</p>
<p>Programowanie modułowe i obiektowe</p>	<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.</p> <p>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>