

WYMAGANIA EDUKACYJNE – INFORMATYKA, ZAKRES ROZAZERZONY, KLASY 1-4

Założenia podstawy programowej z informatyki mają charakter przyrostowy, sugerując w ten sposób spiralny rozwój wiedzy, umiejętności i kompetencji uczniów przez wszystkie lata nauki szkolnej. Zgodnie z podstawą programową uczeń realizujący informatykę w zakresie rozszerzonym powinien spełniać wymagania określone dla zakresu podstawowego. Takie założenia zostały również przyjęte w prezentowanych wymaganiach edukacyjnych. Treści podawane są w sposób systematyczny, przyrostowo, z zachowaniem spiralnego charakteru kształcenia, przy założeniu, że uczeń zrealizował podstawę programową z zakresu podstawowego.

Program, podobnie jak podręczniki z serii „Teraz bajty” dla szkół ponadpodstawowych ma budowę modułową. Pozwala to realizować materiał w podanym układzie lub dostosować kolejność realizacji modułów, a tym samym zawartych w nich tematów, do specyfiki szkoły i grupy uczniów. Każdy temat stanowi gotowy scenariusz lekcji o wyrazistej strukturze merytoryczno-dydaktycznej.

Program podzielono na trzy części, które odpowiadają odpowiednio częściom podręcznika. Program dla każdej części składa się z czterech modułów o podobnych treściach:

- Moduł A. Wokół komputera i sieci komputerowych
- Moduł B. Wokół dokumentów komputerowych
- Moduł C. Wokół algorytmiki i programowania
- Moduł D. Wokół Internetu i projektów

ZAKRES PODSTAWOWY

KLASA 1

Moduł A. Wokół komputera

Komputer i urządzenia peryferyjne				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia i omawia podstawowe elementy komputera; podaje przykłady urządzeń peryferyjnych; wymienia urządzenia peryferyjne	klasyfikuje środki technologii informacyjnej ze względu na przeznaczenie; charakteryzuje przykładowe urządzenia peryferyjne; określa własności i przeznaczenie dysku twardego	potrafi określić podstawowe elementy komputera (wartości podstawowych parametrów, ich wzajemne współdziałanie); wie, czym jest RAM i BIOS, określa ich funkcje; omawia dodatkowe urządzenia pamięci masowej, m.in.: napędy optyczne, pamięci flash, pamięci taśmowe (streamery)	wymienia podstawowe układy mieszczące się na płycie głównej i charakteryzuje ich parametry; wyjaśnia, czym jest karta rozszerzenia; wie, w jakim celu tworzy się partycje na dysku twardym; wyjaśnia pojęcia: <i>partycja dyskowa</i> , <i>formatowanie dysku</i>	potrafi dobrać pełną konfigurację sprzętu i oprogramowania do danego zastosowania; dba o prawidłowe funkcjonowanie komputera, przeprowadzając wszystkie niezbędne testy

Systemy operacyjne i inne oprogramowanie				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, co to jest system operacyjny; omawia podstawowy zestaw oprogramowania, który może być zainstalowany na komputerze	zna funkcje systemu operacyjnego; wymienia popularne systemy operacyjne; omawia rodzaje programów komputerowych i potrafi określić ich przeznaczenie	podaje podstawowe cechy systemu Windows; charakteryzuje narzędzia TI, w tym: oprogramowanie użytkowe, języki programowania, programy narzędziowe; zna podstawowe typy plików	omawia ogólną strukturę systemu operacyjnego; potrafi scharakteryzować różne systemy operacyjne (Windows, Linux, Unix); omawia zawartość plików w zależności od ich rozszerzenia	omawia drogę rozwoju systemu Windows; dokonuje analizy porównawczej różnych systemów operacyjnych

Moduł B. Wokół dokumentów komputerowych

Opracowywanie dokumentów tekstowych o rozbudowanej strukturze				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie do czego służy nagłówek i stopka dokumentu; rozróżnia style tekstu; zapisuje dokument w pliku we wskazanym folderze; zna i stosuje podstawowe zasady redagowania i formatowania tekstu; zna podstawowe zasady pracy z dokumentem wielostronicowym; wstawia tabelę i wykonuje podstawowe operacje na komórkach tabeli; stosuje numerację i wypunktowanie; dzieli dokument na strony	redaguje nagłówek i stopkę, wstawia numery stron; wie, w jakim celu stosuje się style tekstu; stosuje wbudowane style nagłówkowe; stosuje przypisy; właściwie dzieli tekst na akapity; poprawia tekst, wykorzystując możliwości wyszukiwania i zamiany znaków oraz słowniki: ortograficzny i synonimów; stosuje tabulację i wcięcia; wykorzystuje indeksy górny i dolny oraz symbole do pisania prostych wzorów i tekstów w języku obcym; wie do czego służy podział dokumentu na sekcje; rozmieszcza tekst w kolumnach	redaguje inną stopkę i inny nagłówek dla stron parzystych i nieparzystych; stosuje różne wbudowane style tekstu; wie, czym są odwołania w tekście; umieszcza podpisy pod rysunkami, tabelami i wykresami; tworzy spis treści; zmienia ustawienia strony – wielkość marginesów, orientację strony, rozmiar papieru; stosuje różne typy tabulatorów, potrafi zmienić ich ustawienia w całym tekście; dzieli dokument na sekcje; pracuje z dokumentem trybie recenzji; korzysta z opcji śledzenia zmian, wstawia komentarze	tworzy spis ilustracji, tabel i wykresów; stosuje różne sposoby wyświetlania dokumentu; przygotowuje poprawnie zredagowany i sformatowany tekst, dostosowując formę tekstu do jego przeznaczenia; redaguje złożone wzory matematyczne korzystając z edytora równań; samodzielnie wyszukuje opcje menu potrzebne do rozwiązania dowolnego problemu; wykonuje konwersję tekstu na tabelę i odwrotnie; korzysta z podziału tekstu na sekcje; pracuje z dokumentem trybie recenzji i porównuje dokumenty;	samodzielnie odkrywa nowe możliwości edytora tekstu, przygotowując dokumenty tekstowe; tworzy dokumenty tekstowe, stosując poprawnie wszystkie poznane zasady redagowania i formatowania tekstu; przygotowuje profesjonalny tekst – pismo, sprawozdanie, z zachowaniem wszystkich zasad redagowania i formatowania tekstów

Opracowywanie grafiki rastrowej				
2	3	4	5	6
Uczeń: z pomocą nauczyciela korzysta z wybranego programu do tworzenia grafiki rastrowej; wyszukuje potrzebne funkcje w menu programu; wymienia rodzaje grafiki komputerowej	Uczeń: zna formaty plików graficznych; opracowuje grafikę rastrową: stosuje warstwy i narzędzia selekcji, zmianę kontrastu i nasycenia kolorów, kadrowanie i skalowanie; wykonuje proste projekty w grafice wektorowej, korzystając z możliwości wstawiania Kształtów w edytorze tekstu	Uczeń: sprawnie korzysta z Pomocy wbudowanej do programów w celu znalezienia szczegółowych sposobów rozwiązania danego problemu; podaje różnice między grafiką rastrową i wektorową; opracowuje grafikę rastrową: uzyskuje efekty specjalne dzięki zastosowaniu tzw. filtrów; tworzy proste kompozycje, korzystając z wybranego programu do tworzenia grafiki wektorowej; podaje różnice między grafiką 2D i 3D	Uczeń: rozumie znaczenie zapisu pliku graficznego w danym formacie – zależnie od przeznaczenia; omawia zalety, wady i zastosowanie wybranych formatów plików grafiki rastrowej; potrafi zastosować odpowiedni format pliku graficznego; zapisuje pliki w różnych formatach; opracowuje grafikę wektorową: przekształca obraz (pochyla, obraca), grupuje obiekty	Uczeń: samodzielnie zapoznaje się z możliwościami wybranego programu graficznego, przygotowując złożone projekty z różnych dziedzin

Tworzenie prezentacji multimedialnej				
2	3	4	5	6
Uczeń: zna podstawowe typy i zasady tworzenia prezentacji multimedialnej; tworzy prezentację składającą się z kilku slajdów z zastosowaniem animacji niestandardowych; korzysta z szablonów slajdów; umieszcza na slajdach teksty i obrazy; zapisuje prezentację we wskazanym folderze docelowym; uruchamia pokaz slajdów	Uczeń: zna etapy tworzenia prezentacji multimedialnej; przygotowuje prezentację na zadany temat; wie, do czego służą poszczególne widoki slajdów; potrafi ustawić jednakowe tło dla wszystkich slajdów oraz zmienić tło dla wybranego slajdu; wstawia do slajdu wykresy, tabele, równania matematyczne, efekty dźwiękowe	Uczeń: potrafi właściwie zaplanować prezentację na zadany temat; wstawia dźwięki z plików spoza listy standardowej; zmienia tło, wstawia obiekty i hiperłącza; umieszcza przyciski akcji; dodaje animacje i efekty dźwiękowe do obiektów; dodaje narrację do prezentacji; prezentuje swoje prace przed klasą	Uczeń: wstawia podkład muzyczny odtwarzany podczas całej prezentacji; przygotowuje materiały informacyjne dla uczestników pokazu i przeprowadza pokaz; konwertuje przygotowaną prezentację do formatu umożliwiającego publikację w Internecie i otwiera ją lokalnie w przeglądarce internetowej	Uczeń: potrafi samodzielnie zaprojektować i przygotować multimedialną prezentację na wybrany temat, cechującą się ciekawym ujęciem zagadnienia, interesującym układem slajdów

Moduł C. Wokół algorytmiki i programowania

Programowanie i rozwiązywanie problemów z wykorzystaniem komputera				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wyjaśnia co to jest algorytm; podaje przykłady sytuacji problemowych; wyjaśnia pojęcie <i>specyfikacja problemu</i> ; wie, na czym polega programowanie; analizuje gotowe proste programy zapisane w wybranym języku programowania	wyjaśnia pojęcie algorytmu oraz zależności między problemem, algorytmem i programem; dobiera algorytm do rozwiązania problemu; formułuje specyfikację zadania; określa dane do zadania oraz wyniki; zna klasyfikację języków programowania; klasyfikuje języki programowania	omawia etapy rozwiązywania problemu (zadania); testuje rozwiązania; wyjaśnia, na czym polega prezentacja algorytmu w postaci programu; zna pojęcia: <i>program źródłowy, program wynikowy, implementacja, kompilacja, interpretacja, translacja</i> ; porównuje gotowe, proste programy zapisane w różnych językach programowania (wizualnych i tekstowych)	analizuje i porównuje gotowe, proste programy zapisane w języku C++ i/lub Python; odróżnia kompilację od interpretacji; wymienia i charakteryzuje języki programowania	potrafi samodzielnie napisać specyfikację określonego zadania; samodzielnie określa algorytm i narzędzia właściwe do rozwiązania danego problemu

Tworzenie programów w wybranym języku programowania				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
charakteryzuje środowisko programistyczne wybranego tekstowego języka programowania; analizuje gotowe proste programy zapisane w wybranym języku programowania	omawia etapy programowania w wybranym tekstowym języku programowania; wie, na czym polega iteracja; zna kryteria, jakie powinien spełniać poprawny program; wyjaśnia, co to jest iteracja	zna zasady stosowania zmiennych i wykonywania obliczeń w wybranym tekstowym języku programowania; realizuje prostą sytuację warunkową w wybranym języku programowania, stosuje proste warunki logiczne; sprawdza poprawność danych; zapisuje proste algorytmy iteracyjne w postaci listy kroków	zapisuje rozwiązanie problemu w wybranym tekstowym języku programowania; realizuje sytuację warunkową w wybranym języku programowania; stosuje złożone warunki logiczne; zapisuje rozwiązanie problemu iteracyjnego w postaci programu	zapisuje złożony algorytm w wybranym tekstowym języku programowania; samodzielnie pisze program realizujący algorytm z warunkami zagnieżdżonymi; stosuje zagnieżdżone instrukcje iteracyjne; uczestniczy w konkursach i olimpiadach informatycznych

Moduł D. Wokół Internetu i projektów

Internet i wyszukiwanie informacji w Internecie				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wyszukuje strony WWW poprzez proste hasło wpisywane do wyszukiwarki internetowej; zna zasady nawigacji po stronie WWW, poruszając się po wybranych stronach internetowych	wie, czym są Internet i strona WWW oraz zna genezę powstania Internetu; wymienia wybrane usługi Internetowe; podaje opisy i zastosowania wyszukiwarki internetowej; szuka informacji w Internecie, konstruuując złożone hasło	omawia rozwój usług internetowych, wskazując najważniejsze fakty; wyjaśnia, na czym polega przeglądanie strony internetowej; potrafi właściwie zawęzić obszar poszukiwań, aby szybko odszukać informacje; korzysta z encyklopedii i słowników w wersji elektronicznej; wyszukuje informacje zapisane w innych językach; korzysta z serwisu mapowego	omawia organizację informacji w WWW; wyjaśnia postać adresu URL; potrafi zastosować różne narzędzia do wyszukiwania informacji, usprawniając szukanie informacji; właściwie porządkuje informacje o stronach WWW; potrafi odpowiednio ocenić przydatność i wiarygodność informacji	potrafi formułować własne wnioski i spostrzeżenia dotyczące rozwoju Internetu, jego znaczenia dla różnych dziedzin gospodarki i dla własnego rozwoju; wyszukuje, gromadzi i właściwie selekcjonuje informacje, tworząc złożone projekty z różnych dziedzin

Usługi internetowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia przykładowe e-usługi, np. e-nauczanie, e-banki, e-sklepy, e-aukcje, e-podpis; wie, na czym polegają nauczanie i praca na odległość	omawia przykładowe e-usługi; korzysta z wybranych e-usług, np. e-learningu; jest świadomy istnienia zagrożeń wynikających z korzystania z e-usług	omawia zalety i wady poszczególnych e-usług; zna i stosuje zasady bezpiecznego korzystania z poszczególnych e-usług	wyjaśnia działanie e-banku; podaje metody zabezpieczeń; podaje zasady korzystania z poszczególnych e-usług; wie, czym jest podpis elektroniczny	potrafi przedstawić własne wnioski z analizy zalet i wad poszczególnych e-usług; korzystając z dodatkowych źródeł, znajduje najnowsze informacje na temat e-usług

KLASA 2

Moduł A. Wokół komputera i sieci komputerowych

Praca w sieci komputerowej				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, czym jest sieć komputerów i dlaczego komputery łączą się w sieć. korzysta z podstawowych usług sieci	wymienia podstawowe klasy sieci; rozumie pojęcie logowania się do sieci; omawia podstawowe sposoby łączenia komputerów w sieć; wymienia korzyści płynące z połączenia komputerów w sieć	zna podstawy konfiguracji sieci (protokoły sieciowe, identyfikacja sieciowa); wymienia elementy niezbędne do budowy sieci; potrafi udostępniać zasoby komputera; omawia korzyści płynące z połączenia komputerów w sieć	omawia przykładowe schematy sieci: domowej i szkolnej; udostępnia zasoby w sieci	potrafi mapować zasoby komputera; wie, czym jest maska podsieci

Bezpieczeństwo i ochrona danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia sposoby ochrony danych w komputerach i sieciach komputerowych	zna zasady ochrony danych w komputerach i sieciach komputerowych	rozumie potrzebę wykonywania podstawowych operacji porządkujących zasoby komputera oraz stosowania podstawowych zasad ochrony własnych dokumentów i zasobów komputera; zna sposoby ochrony przed utratą danych	podając przykłady, dyskutuje na temat odmian złośliwego oprogramowania i oprogramowania zabezpieczającego komputer	dzieli się własnymi doświadczeniami w zakresie bezpieczeństwa i ochrony danych w komputerach

Moduł B. Wokół dokumentów komputerowych

Formuły, funkcje i wykresy w arkuszu kalkulacyjnym				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna podstawowe zastosowania arkusza kalkulacyjnego; potrafi zaznaczyć zadany blok komórek; ustawia liczbowy format danych; samodzielnie pisze formułę wykonującą jedno z czterech podstawowych działań arytmetycznych (dodawanie, odejmowanie, mnożenie, dzielenie); potrafi zastosować kopiowanie i wklejanie formuł; tworzy prosty wykres; zapisuje utworzony skoroszyt we wskazanym folderze docelowym; zna i stosuje podstawowe funkcje arkusza kalkulacyjnego: SUMA, ŚREDNIA	rozdziela zasady adresowania w arkuszu kalkulacyjnym; stosuje adresowanie bezwzględne wtedy, gdy jest to uzasadnione; potrafi tworzyć formuły wykonujące bardziej zaawansowane obliczenia (potęgowanie, pierwiastkowanie, z zastosowaniem nawiasów); tworzy wykres składający się z wielu serii danych, dodając do niego odpowiednie opisy; ustawia inne formaty danych poza liczbowym; formatuje tabelę; korzysta z możliwości wstawiania funkcji; potrafi zastosować funkcję JEŻELI	poprawnie planuje tabelę w arkuszu kalkulacyjnym, umieszczając w niej dane liczbowe i opisy; stosuje adresowanie mieszane; stosuje formatowanie warunkowe tabeli arkusza; potrafi stosować wybrane funkcje arkusza kalkulacyjnego do rozwiązywania zadań z różnych przedmiotów; zna zastosowania różnych typów wykresów; potrafi narysować wykres wybranej funkcji matematycznej; tworzy wykres funkcji trygonometrycznej; wie, na czym polega myślenie komputacyjne	tworzy rozbudowane formuły z zastosowaniem funkcji arkusza kalkulacyjnego; stosuje wybrane funkcje arkusza kalkulacyjnego: statystyczne, logiczne, matematyczne, daty i czasu; panuje kolejne kroki rozwiązywania problemu, z uwzględnieniem podstawowych etapów myślenia komputacyjnego; przedstawia dane i wyniki w tabelach i na wykresach; dopasowuje wygląd arkusza kalkulacyjnego po wydruku - dobiera ustawienia strony, ustawia podział stron i obszar wydruku	potrafi przeprowadzić analizę przykładowego problemu i opracować właściwy algorytm obliczeń; potrafi samodzielnie planować kolejne kroki rozwiązywania problemu, z uwzględnieniem podstawowych etapów myślenia komputacyjnego; potrafi samodzielnie zrealizować rozwiązanie danego problemu; zna działanie i zastosowanie większości funkcji dostępnych w arkuszu kalkulacyjnym; samodzielnie opracowuje problemy zgodnie z kolejnymi krokami myślenia komputacyjnego

Filtry oraz tabele i wykresy przestawne w arkuszu kalkulacyjnym				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, do czego służą filtry; potrafi, z pomocą opisu w podręczniku i nauczyciela, wyświetlić dane według prostego kryterium	korzysta z możliwości ustawiania niestandardowych filtrów do filtrowania danych w arkuszu kalkulacyjnym	potrafi stosować filtry i selekcjonować dane na podstawie zaawansowanych kryteriów; wie, do czego służą tabele przestawne; tworzy tabele i wykresy przestawne, korzystając z przykładów z podręcznika	tworzy tabele i wykresy przestawne, analizując dane zgromadzone w arkuszu kalkulacyjnym; stosuje filtry w tabeli przestawnej	odszukuje w Pomocy informacje na temat tabel i wykresów przestawnych; potrafi samodzielnie określić dane, jakie można przedstawić i poddać analizie z wykorzystaniem tablic i wykresów przestawnych

Opracowywanie grafiki wektorowej				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
korzysta z podstawowych możliwości wybranego programu do tworzenia grafiki wektorowej; wykonuje proste projekty w grafice wektorowej, korzystając z możliwości wstawiania Kształtów w edytorze tekstu; tworzy i edytuje prosty rysunek w wybranym programie graficznym, korzystając z podstawowych narzędzi do rysowania figur	wykonuje projekty w grafice wektorowej, korzystając z możliwości wstawiania Kształtów w edytorze tekstu; grupuje obiekty; tworzy proste kompozycje, korzystając z wybranego programu do tworzenia grafiki wektorowej; zna podstawowe możliwości wybranego programu do edycji obrazu wektorowego; korzysta z narzędzi do rysowania figur i ścieżek; przekształca obraz –przeciąga, skaluje, obraca	wyszukuje potrzebne funkcje programu w menu programu graficznego; tworzy kompozycje z figur w grafice wektorowej; rysuje bryły, korzystając z możliwości rysowania przestrzennego	dostrzega różnice między grafiką rastrową i wektorową. opracowuje grafikę wektorową, rysując ciekawe kompozycje z figur, przy tym potrafi zmienić właściwości wybranego narzędzia; wyjaśnia, czym są ścieżki i rysuje je, używając odpowiednich narzędzi; wyjaśnia, czym są punkty węzłowe; wie, na czym polega praca z warstwami; wykonuje rysunki, korzystając z warstw	zapoznaje się z możliwościami wybranego programu graficznego, korzystając z Pomocy i innych źródeł; poznaje możliwości programów graficznych; przygotowuje złożone projekty z różnych dziedzin; tworzy obrazy, wykorzystując różne możliwości programu; przygotowuje grafikę do własnej strony internetowej lub prezentacji multimedialnej; uczestniczy w konkursach dotyczących grafiki komputerowej

Moduł C. Wokół algorytmiki i programowania

Tworzenie programów w wybranym języku programowania				
2	3	4	5	6
Uczeń: tworzy prosty program w języku wysokiego poziomu, np. wyświetlający napis na ekranie, wykonujący proste obliczenia; stosuje w programach zmienne i wykonuje proste obliczenia, np. oblicza sumę dwóch liczb, średnią z dwóch liczb; potrafi uruchomić utworzony program i wyprowadzić wyniki na ekran; analizuje i omawia działanie gotowych prostych programów zapisanych w wybranym języku programowania, zawierających instrukcję pętli for i/lub instrukcję warunkową if	Uczeń: zna postać i działanie instrukcji iteracyjnej while w wybranym języku programowania i stosuje ją w tworzonych programach komputerowych; analizuje (wspólnie z nauczycielem) programy, w których zastosowano funkcje; zapisuje je, uruchamia i wyjaśnia ich działanie; wie, co to są podprogramy i zna ich zastosowanie; definiuje (korzystając z podręcznika) funkcje bez parametrów i stosuje je w programach; korzystając z podręcznika, deklaruje tablice, wczytuje i wyprowadza elementy tablicy i/lub listy na ekran	Uczeń: zna postać i działanie instrukcji iteracyjnej do ... while w języku C++ i stosuje ją w tworzonych programach komputerowych; wyjaśnia pojęcia: <i>parametr formalny</i> , <i>parametr aktualny</i> ; definiuje funkcje z parametrami w wybranym języku wysokiego poziomu; zna sposób definiowania funkcji zwracającej wartość i niezwracającej wartości; zna pojęcia: <i>tablica</i> , <i>zmienna indeksowana</i> ; na bazie przykładów z podręcznika, deklaruje tablicę i/lub listę, wczytuje i wyprowadza elementy tablicy i/lub listy; definiując odpowiednie funkcje w wybranym języku programowania	Uczeń: wyjaśnia różnicę pomiędzy funkcją zwracającą wartość i niezwracającą wartości; stosuje funkcje bez parametrów i z parametrami w programach; potrafi zastosować tablicę i/lub listę w zadaniach; potrafi odwoływać się do dowolnego elementu tablicy i/lub listy; wykonuje operacje na elementach tablicy i/lub listy; potrafi modyfikować program, znaleźć błędy i je poprawić	Uczeń: potrafi samodzielnie zastosować odpowiedni rodzaj instrukcji pętli w tworzonym programie; omawia podobieństwa i różnice w działaniu wszystkich omówionych instrukcji pętli w dwóch różnych językach programowania; omawia podobieństwa i różnice w definiowaniu tablic i/lub list w dwóch różnych językach programowania; stosuje w programach tablice i/lub listy, odpowiednio dobierając określoną strukturę danych do algorytmu; pisze trudniejsze programy, w których stosuje funkcje i tablice

Programowanie wybranych algorytmów				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
analizuje opis jednego z wybranych algorytmów, np. porządkowania metodą bąbelkową; potrafi wyjaśnić na czym polega, zademonstrować z użyciem pomocy dydaktycznych, przetestować	analizuje opisane w podręczniku gotowe programy realizujące jeden wybrany algorytm z podanych w podręczniku, np. pierwszości liczby; przepisuje wybrane kody programów, uruchamia programy i wyjaśnia ich działanie	omawia dwa wybrane algorytmy, np. obliczania elementów ciągu liczb Fibonacciego i porządkowania przez wstawianie, demonstruje je przy użyciu pomocy dydaktycznych; korzystając z opisu w podręczniku zapisuje je w wybranym języku programowania i potrafi wyjaśnić ich działanie	omawia algorytmy badania pierwszości liczby i obliczania elementów ciągu liczb Fibonacciego, porządkowania metodą bąbelkową i przez wstawianie; zapisuje je w postaci programów, korzystając z podręcznika, rozumie działanie tych programów; stosuje funkcje i tablice w zapisie w/w algorytmów w postaci programów	samodzielnie zapisuje w postaci programów algorytmy badania pierwszości liczby i obliczania elementów ciągu liczb Fibonacciego, porządkowania metodą bąbelkową i przez wstawianie; definiuje odpowiednie i funkcje; rozwiązuje przykładowe zadania z konkursów informatycznych; bierze udział w konkursach

Moduł D. Wokół Internetu i projektów

Wybrane przepisy prawa				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna podstawowe przepisy prawa dotyczące korzystania z cudzych materiałów i stosuje je w praktyce; zna podstawowe zasady korzystania z programów komputerowych; rozumie konieczność posiadania licencji na programy komputerowe; jest świadomy istnienia przestępstw komputerowych	wie, co jest przedmiotem prawa autorskiego i co jemu nie podlega; zna pojęcie licencji; wymienia przykładowe rodzaje licencji; wymienia przykładowe rodzaje przestępstw komputerowych	wyjaśnia wybrane przepisy prawa autorskiego, m.in.: „dozwolony użytek utworów”, zasady korzystania z cudzego utworu bez pytania o zgodę, ochrona wizerunku; omawia przykładowe rodzaje licencji na programy komputerowe; omawia wybrane przykłady przestępstw komputerowych	potrafi uzasadnić zastosowanie wybranego przepisu prawa w konkretnym przypadku; podaje przykłady łamania wybranych przepisów prawa; omawia różnice pomiędzy różnymi rodzajami licencji; sprawdza, na podstawie jakiej licencji jest rozpowszechniany dany program i wyjaśnia zasady tej licencji	potrafi samodzielnie interpretować ważniejsze przepisy prawa autorskiego dotyczące korzystania z różnych źródeł informacji i ochrony programów komputerowych; wyszukuje dodatkowe informacje na temat przestępstw komputerowych

Komunikacja i wymiana informacji w Internecie				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia podstawowe zasady pisania listów elektronicznych; podaje przykładowe sposoby komunikacji i wymiany informacji z wykorzystaniem Internetu; zna i stosuje zasady netykiety; podaje przynajmniej dwie korzyści wynikające z rozwoju technologii informacyjno-komunikacyjnych (TIK); jest świadomy istnienia zagrożeń wynikających z rozwoju TIK	poprawnie redaguje listy elektroniczne, dbając o ich formę i treść; omawia wybrane formy komunikacji i wymiany informacji; zna i stosuje zasady netykiety dotyczące form komunikacji, z których korzysta; wyjaśnia, na czym polega komunikacja w czasie rzeczywistym; wie, czym jest chmura obliczeniowa;	rozdziela poszczególne formy komunikowania się przez Internet; rozdziela poszczególne sposoby komunikacji i wymiany informacji; omawia działanie poczty elektronicznej; zna i stosuje zasady netykiety dotyczące wszystkich form komunikacji; zna i omawia możliwości korzystania z chmury obliczeniowej;	potrafi dokonać analizy porównawczej różnych form komunikacji i wymiany informacji, podając opis poszczególnych form i niezbędne wymagania; porównuje metody dostępu do poczty elektronicznej; wyjaśnia, na czym polega korzystanie z oprogramowania w chmurze i na czym polega współdzielenie dokumentów umieszczonych w chmurze;	samodzielnie wyszukuje w różnych źródłach informacje na temat nowoczesnych możliwości korzystania z Internetu, np. za pomocą urządzeń mobilnych; wyszukuje w Internecie informacji na temat łączenia się z chmurą; potrafi samodzielnie ocenić znaczenie technologii informacyjno-komunikacyjnej w

	wymienia przynajmniej dwie możliwości korzystania z chmury; podaje zalety korzystania z komunikacji za pomocą Internetu; wymienia podstawowe zagrożenia wynikające z rozwoju technologii informacyjno-komunikacyjnej	omawia szczegółowo zagrożenia wynikające z rozwoju technologii informacyjno-komunikacyjnej (m.in.: cyberprzemoc, anonimowość kontaktów, uzależnienie od komputera); wie, czym jest fake news; omawia korzyści i zagrożenia dotyczące korzystania z różnych form komunikacji i wymiany informacji z wykorzystaniem Internetu	wyjaśnia, jakie korzyści daje rozwój technologii informacyjno-komunikacyjnej osobom z niepełnosprawnościami; bierze aktywny udział w debacie na temat szans i zagrożeń wynikających z rozwoju TIK.	komunikacji i wymianie informacji oraz zna najnowsze osiągnięcia w tej dziedzinie
--	--	---	---	---

Tworzenie stron internetowych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, w jaki sposób zbudowane są strony WWW; zna najważniejsze narzędzia do tworzenia stron internetowych; wie, na czym polega tworzenie strony internetowej; zapoznaje się z przykładowym źródłem strony internetowej, przeglądając strukturę pliku	potrafi przygotować prostą stronę internetową, używając dowolnego edytora tekstu; umie tworzyć akapity i wymuszać podział wiersza, dodawać nagłówki do tekstu, zmieniać krój i wielkość czcionki; wie, jak wstawiać linie rozdzielające; umie wstawiać hiperłącza, korzystać z kotwic; rozumie strukturę plików HTML; podaje przykłady stosowania stylów CSS	potrafi tworzyć proste strony w języku HTML, używając edytora tekstowego; zna funkcje i zastosowanie najważniejszych znaczników HTML; potrafi wstawiać grafikę do utworzonych stron; umie tworzyć listy wypunktowane i numerowane; zna nazewnictwo kolorów; zna najczęściej wykorzystywane atrybuty CSS i sposoby określania ich wartości; omawia sposoby publikowania strony w Internecie	umie wstawiać tabele do tworzonych stron i je formatować; koduje polskie znaki; tworzy podstronę; umieszcza łącza hipertekstowe, m.in. łącząc stronę główną z podstroną; stosuje kolory; analizuje wady i zalety różnych sposobów publikowania i promowania stron w Internecie; przygotowuje stronę do publikacji w Internecie i ją publikuje	zna zagadnienia dotyczące promowania stron WWW; potrafi stworzyć własny, rozbudowany serwis WWW i przygotować go w taki sposób, żeby wyglądał estetycznie i zachęcał do odwiedzin; zna większość znaczników HTML

KLASA III

Moduł A. Wokół komputera i sieci komputerowych

Reprezentacja danych w komputerze				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna pojęcie systemu pozycyjnego; wie, co to jest system binarny; analizuje gotowy przykład z podręcznika obliczający wartość dziesiętną liczby zapisanej w systemie dwójkowym i na tej podstawie wykonuje podobne ćwiczenie	definiuje pojęcie systemu pozycyjnego; wie na czym polega działanie procesora; potrafi dokonać konwersji liczby między systemem dziesiętnym a dwójkowym oraz dwójkowym a dziesiętnym	zna podwójne nazwy pozycyjnych systemów liczbowych tj. dwójkowy – binarny, dziesiętny – decymalny, szesnastkowy - heksadecymalny; wyjaśnia, w jaki sposób procesor dodaje liczby; dokonuje konwersji liczb między systemem dziesiętnym i szesnastkowym oraz szesnastkowym i dziesiętnym	zna elementy uproszczonego modelu komputera zgodny z ideą von Neumanna; dokonuje konwersji liczb między systemem szesnastkowym i binarnym	potrafi narysować uproszczony model komputera zgodny z ideą von Neumanna; samodzielnie potrafi dokonać zamiany między trzema systemami pozycyjnymi (dwójkowym, dziesiętnym i szesnastkowym) w jednym zadaniu

Wybrane urządzenia cyfrowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia urządzenia cyfrowe wykorzystywane w szkole podczas zajęć (np. drukarka, drukarka 3D, tablica interaktywna, monitor, kamera); podaje nazwy urządzeń cyfrowych wykorzystywane w domu i poza nim (np. płyta grzejna, okap kuchenny, odtwarzacze audio, system multiroom, system nawigacji, smartwatch)	omawia funkcje poznanych urządzeń używanych w szkole oraz w domu i poza nim; potrafi zaprezentować w klasie wybrane urządzenie cyfrowe i omówić jego działanie	korzysta z wyszukiwarki internetowej celem opracowania informacji na temat wybranego urządzenia cyfrowego; z pomocą nauczyciela przygotowuje model 3D do druku 3D, korzystając z odpowiedniego oprogramowania; uruchamia drukarkę 3D i wykonuje przykładowy wydruk (lub omawia sposób drukowania – w przypadku braku drukarki w szkole)	objaśnia funkcje poznanych urządzeń używanych w domu i poza nim; zna podstawowe możliwościami oprogramowania towarzyszącego wybranemu urządzeniu, np. drukarce 3D i przygotowuje model 3D do wydruku; samodzielnie potrafi uruchomić drukarkę 3D i wykonać przykładowy wydruk (w przypadku, gdy szkoła ma takie możliwości)	wymienia parametry techniczne urządzeń cyfrowych podanych w specyfikacji technicznej; potrafi posługiwać się instrukcją obsługi urządzeń cyfrowych i poznawać samodzielnie możliwości towarzyszącego im oprogramowania

Moduł B. Wokół dokumentów komputerowych

Tworzenie baz danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wyjaśnia, na czym polega przetwarzanie danych; definiuje pojęcie baza danych; na przykładzie gotowego pliku bazy danych potrafi omówić jej strukturę – określić, jakie informacje są w niej pamiętane podaje obszary zastosowań baz danych – na przykładach z najbliższego otoczenia – szkoły, instytucji naukowych, społecznych i gospodarczych; wyjaśnia pojęcia: <i>baza danych</i> , <i>rekord</i> i <i>pole</i> ;	rozumie organizację danych w relacyjnych bazach danych; potrafi przygotować schemat prostej relacyjnej bazy danych; tworzy prostą bazę danych, składającą się z dwóch tabel; planuje zawartość tabel; stosuje zasady tworzenia tabel; wie do czego służy import danych w bazie danych	omawia etapy przygotowania bazy danych; określa odpowiednio typy danych; rozumie pojęcia relacji i klucza podstawowego; przygotowuje projekt formularza i raportu; tworzy tabele i korzysta z Widoku projektu ; umie zaimportować dane z arkusza kalkulacyjnego do bazy danych	rozumie, co oznacza przetwarzanie danych w bazach danych; definiuje relacje między tabelami; potrafi uzasadnić, dlaczego warto umieszczać dane w kilku tabelach połączonych relacją; podczas rozwiązywania nowego problemu korzysta z doświadczeń zdobytych przy rozwiązaniu innego, podobnego problemu	analizuje problemy występujące w utworzonej bazie danych i znajduje ich rozwiązanie; samodzielnie przygotowuje projekt bazy danych (składającej z trzech tabel) i potrafi ją wykonać w programie do tworzenia baz danych

Wykonywanie podstawowych operacji na relacyjnej bazie danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna zastosowanie filtrów do wyszukiwania danych; potrafi wyświetlić wynik gotowego zapytania i omówić, czego zapytanie dotyczy; omawia zastosowanie korespondencji seryjnej	definiuje pojęcie kwerendy; tworzy kwerendę wybierającą w Widoku projektu ; zna sposób przygotowania korespondencji seryjnej z wykorzystaniem danych z bazy danych	modyfikuje gotowe zapytania; tworzy kwerendę parametryczną; umieszcza w korespondencji seryjnej pola z tabeli bazy danych; korzysta z gotowych szablonów listów seryjnych	samodzielnie modyfikuje i tworzy kwerendy; wie, jaka jest korelacja między edytorem tekstu a bazą danych podczas tworzenia korespondencji seryjnej; w edytorze tekstu przygotowuje listy seryjne i etykiety adresowe, korzystając z danych zapisanych w bazie danych	potrafi zaprojektować samodzielnie relacyjną bazę danych (składającą się z trzech tabel), ustala typy pól; potrafi budować złożone kwerendy z dwóch lub więcej tabel połączonych; opracowuje własny szablon listu seryjnego

Moduł C. Wokół algorytmiki i programowania

Algorytmy na tekstach w językach C++ i Python				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, że w edytorach tekstu wykorzystywane są algorytmy na tekstach – pokazuje przykłady wyszukiwania znaków w tekście, porównywania tekstów; potrafi omówić, posługując się przykładami i pomocami dydaktycznymi, wybrany algorytm na tekstach; analizuje gotowy program wykorzystujący dane tekstowe i objaśnia stosowanie zmiennych tekstowych; testuje gotowe programy dla różnych danych	planuje kolejne kroki rozwiązania problemu porównywania tekstów, szukając rozwiązania; deklaruje zmienne typu tekstowego char i string; realizuje algorytm porównania dwóch tekstów; wie, na czym polega algorytm szukania wzorca w tekście; wie, jak odwoływać się do pojedynczego znaku łańcucha, wie, jak wyznaczyć długość łańcucha – potrafi zastosować funkcję length() oraz strlen()(C++) i funkcję len() (Python); analizuje funkcję realizującą np. algorytm porównywania tekstów i omawia działanie funkcji; tworzy program wykorzystujący tę funkcję	przeprowadza analizę rozwiązania algorytmu szukania znaku w tekście; szuka wystąpień wzorca w tekście metodą naiwną, analizując i uzupełniając kolejne kroki algorytmu z wykorzystaniem podręcznika; definiuje pojęcie konkatencji; formułuje treść zadania do przedstawionego kodu źródłowego; definiuje funkcję szukającą i zliczającą wystąpienia znaków lub ciągów znaków w tekście; korzysta z instrukcji if...elif (Python) do utworzenia menu programu	opracowuje program zliczający wystąpienie znaku w tekście; korzysta z funkcji wyboru switch()(C++) do utworzenia menu programu; umieszcza w kodzie źródłowym funkcję getline() do wprowadzania napisów składających się z wyrazów oddzielonych spacjami	tworzy zaawansowane programy wykorzystujące dane tekstowe i poznane funkcje; potrafi samodzielnie utworzyć algorytm i program realizujący porównywanie tekstów; tworzy rozbudowane menu wyboru z wykorzystaniem funkcji switch()(C++); tworzy rozbudowane menu wyboru z wykorzystaniem z instrukcji if ... elif (Python)

Algorytmy szyfrowania w językach C++ i Python				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, czym jest szyfrowanie danych i w jakim celu się je stosuje; potrafi, korzystając z przykładu z podręcznika, przeanalizować prosty przykład szyfrowania	zna pojęcia: <i>szyfr, szyfrowanie, deszyfrowanie</i> ; omawia schemat procesu szyfrowania i deszyfrowania; zna przynajmniej jeden algorytm szyfrowania danych – szyfr Cezara; korzystając z podręcznika szyfruje i deszyfruje wiadomość, korzystając z szyfru Cezara; analizuje i rozumie działanie funkcji (podanych w podręczniku) realizujących wybrany algorytm szyfrowania	wie, czym zajmuje się kryptologia i kryptoanaliza; stosuje szyfrowanie tekstu metodą Cezara; tworzy program realizujący algorytm szyfrowania szyfrem Cezara; formułuje algorytm deszyfrowania podstawieniowego i pisze funkcję realizującą ten algorytm	pisze program realizujący algorytm szyfrowania podstawieniowego (opracowuje odpowiednie funkcje pomocnicze); pisze funkcję deszyfrowania podstawieniowego; objaśnia szyfrowanie symetryczne i asymetryczne	omawia dziedziny gospodarki, w których wykorzystywane jest szyfrowanie danych; wyszukuje dodatkowe informacje na temat szyfrowania danych; omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat

Algorytmy na liczbach w językach C++ i Python				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
przeprowadza konwersję liczby z systemu dwójkowego na dziesiętny oraz z dziesiętnego na dwójkowy (bazując na przykładach z tematu A1); wyjaśnia, na czym polega wyznaczanie NWD i NWW	korzystając z fragmentu programów z podręcznika, pisze programy w wybranym języku programowania (C++, Python): obliczające wartość dziesiętną liczby dwójkowej i wyznaczające rozwinięcie dwójkowe liczby dziesiętnej; testuje programy dla różnych danych; opracowuje funkcję w języku programowania wyznaczającą NWD; opracowuje funkcję w języku programowania wyznaczającą NWW	analizuje algorytm zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie wyróżniając podproblemy; testuje działanie algorytmów: obliczania wartości dziesiętnej liczby zapisanej w dowolnym systemie i zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie, korzystając z odpowiednich list kroków (z podręcznika); wie, jak można reprezentować ułamki zwykłe w języku C++ (deklaruje rekordy za pomocą słowa kluczowego struct) lub w języku Python; zna wzory na obliczanie sumy ułamków zwykłych oraz mnożenie i dzielenie ułamków zwykłych	na podstawie list kroków (podanych w podręczniku) tworzy programy: obliczania wartości dziesiętnej liczby zapisanej w dowolnym systemie i zapisujący liczbę dziesiętną w systemie liczbowym o określonej podstawie; potrafi zastosować schemat Hornera do obliczenia wartości wielomianu; analizuje funkcje skracającą ułamki zwykłe i dodającą ułamki zwykłe oraz stosuje je do napisania programów: skracającego ułamki zwykłe oraz dodającego ułamki zwykłe; definiuje funkcje: odejmującą, mnożącą oraz dzielącą ułamki zwykłe; wywołuje funkcje w programie głównym i testuje programy dla różnych danych	pisze program wykonujący konwersję liczb zapisanych w dowolnych podstawach; bierze udział w konkursach informatycznych i/lub olimpiadzie informatycznej

Moduł D. Wokół Internetu i projektów

Elementy robotyki – projekty				
2	3	4	5	6
Uczeń: wie czym zajmuje się robotyka; potrafi scharakteryzować funkcje mikrokontrolera Arduino; zna etapy tworzenia projektu grupowego i wykonuje proste zadania szczegółowe z projektu grupowego	Uczeń: zna podstawową strukturę programu dla mikrokontrolera Arduino; potrafi nazwać moduły składające się na system sterowania nawadnianiem ogrodu; wykonuje przydzielone zadania szczegółowe	Uczeń: na podstawie wytycznych wykonuje układ elektroniczny oraz pisze program obsługujący system „podlewania ogrodu”; prawidłowo zapisuje, przechowuje i udostępnia dokumenty potrzebne do realizacji projektu; prezentuje efekty wspólnej pracy	Uczeń: rozszerza system „podlewania ogrodu”, zgodnie z opisem w podręczniku; konstruuje robota, stosując elementy elektroniczne oraz programuje jego funkcje; pełni rolę koordynatora projektu grupowego, m.in.: określa i przydziela zadania szczegółowe; scala elementy projektu wykonane przez członków grupy w jeden projekt	Uczeń: samodzielnie tworzy programy sterujące mikrokontrolerem; wykonuje wybrany inny system, np. system „Wycisz radio” po otwarciu drzwi do pokoju; proponuje tematykę własnego projektu, samodzielnie wyznacza zadania szczegółowe i sposób ich realizacji

Więcej na temat tworzenia stron internetowych – projekt				
2	3	4	5	6
Uczeń: wie, czym są blogi; potrafi znaleźć blog o wybranej tematyce; zna najważniejsze narzędzia do tworzenia stron internetowych; wie na czym polega tworzenie strony internetowej; wykonuje proste zadania szczegółowe z projektu grupowego; przestrzega zasad korzystania z cudzych materiałów	Uczeń: wie, jak założyć blog; wie, czym jest system zarządzania treścią; omawia etapy tworzenia strony internetowej; uczestniczy w przygotowaniu projektu graficznego strony internetowej; wie, jak ustalić tło strony internetowej; wykonuje przydzielone zadania szczegółowe; prezentuje efekty wspólnej pracy	Uczeń: potrafi założyć prosty blog o wybranej tematyce; korzysta z szablonów do tworzenia stron; przygotowuje projekt graficzny strony internetowej; potrafi ustawić listy w wierszach i kolumnach na stronie internetowej; omawia wybrane atrybuty CSS i podaje przykłady ich stosowania; prawidłowo zapisuje, przechowuje i udostępnia	Uczeń: tworzy stronę internetową wzbogaconą o dodatkowe elementy; potrafi tworzyć przyciski na stronie internetowej z elementów listy poprzez dodanie obramowania i innych atrybutów; stosuje wybrane atrybuty CSS; testuje stronę internetową, określając czy, projekt został wykonany zgodnie ze specyfikacją;	Uczeń: wykonuje samodzielnie projekt graficzny strony internetowej na wybrany przez siebie temat; tworzy własną stronę internetową wzbogaconą o dodatkowe elementy, w tym tabelami, listami, elementy dynamiczne; posługuje się arkuszem stylów; publikuje stronę w Internecie; proponuje tematykę własnego projektu, samodzielnie wyznacza zadania szczegółowe i sposób ich realizacji

		dokumenty potrzebne do realizacji projektu	pełni rolę koordynatora projektu grupowego; przydziela zadania szczegółowe; scala dokumenty wykonane przez członków grupy	
--	--	--	---	--

ZAKRES ROZSZERZONY

KLASA I i II

Moduł A. Wokół komputera i sieci komputerowych

Kompresja danych				
2	3	4	5	6
Uczeń: rozumie, na czym polega kompresja danych i w jakim celu się ją stosuje; zna wzór na obliczenie współczynnika kompresji	Uczeń: wymienia i omawia rodzaje kompresji – bezstratną i stratną; potrafi obliczyć współczynnik kompresji; zna pojęcie <i>przepływności</i>	Uczeń: omawia algorytmy kompresji stratnej; omawia algorytmy kompresji bezstratnej (statyczne, słownikowe)	Uczeń: dokonuje kompresji informacji, m.in. stosuje w przykładowych ćwiczeniach algorytmy kompresji bezstratnej; objaśnia różnice między kompresją stratną i bezstratną tekstów, obrazów, dźwięków i filmów	Uczeń: wykonuje samodzielnie zadania dla zainteresowanych

Sztuczna inteligencja				
2	3	4	5	6
Uczeń: wie czym jest sztuczna inteligencja; potrafi wymienić obszary zastosowania sztucznej inteligencji (SI);	Uczeń: umie przeanalizować prosty system SI w arkuszu kalkulacyjnym; potrafi wytłumaczyć schemat sieci neuronowej; konstruuje zapytania w systemie ChatGPT (lub podobnym); tworzy proste grafiki w systemie tworzenia obrazu	Uczeń: konstruuje rozwinięte zapytania w systemie ChatGPT (lub podobnym); samodzielnie korzysta z systemów tworzenia obrazów	Uczeń: wskazuje zalety i wady z korzystania z systemów wykorzystujących SI; korzysta z różnych systemów tworzenia obrazów	Uczeń: wyszukuje w Internecie inne systemy wykorzystujące sztuczną inteligencję oraz ocenia ich przydatność w różnych dziedzinach; prowadzi dyskusję o zastosowaniu SI i potrafi prawidłowo uargumentować swoje zdanie

Moduł B. Wokół dokumentów komputerowych

Tworzenie i edytowanie obrazów rastrowych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna podstawowe możliwości programu do edycji obrazu rastrowego (programu GIMP); potrafi utworzyć prosty rysunek, stosując różne narzędzia malarskie i korekcyjne oraz wybrać odpowiedni tryb ich pracy	wykonuje operacje na obszarach selekcji, m.in.: wycinanie, kopiowanie, kadrowanie, obrysowywanie; stosuje poznane możliwości programu GIMP do edycji zdjęć	zna przeznaczenie i odpowiednio stosuje wybrane klawisze modyfikujące (i kombinacje klawiszy); potrafi wykonać proste modyfikacje barw: zmianę jasności, odwracanie kolorów, zmianę nasycenia, odcieni kolorów	zna pojęcia: <i>histogram</i> , <i>krzywa barw</i> ; tworzy własny pędzel; wykorzystuje tryb Kolor ; operuje na kanałach barw; używa krzywej jasności i funkcji histogramu; korzysta z narzędzia klonowania; w razie konieczności wyszukuje informacje w Pomocy	samodzielnie zapoznaje się z dodatkowymi możliwościami programu GIMP, w tym opcjami tworzenia pędzli; rozwiązuje trudniejsze zadanie, korzystając z narzędzia klonowania i innych; wyszukuje w Internecie informacje na temat tworzenia zdjęć panoramicznych; uczestniczy w konkursach dotyczących grafiki komputerowej

Praca z warstwami obrazu i animacje				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
poprawia jakość zdjęcia, stosując wybrane filtry; wie, na czym polega praca z warstwami	wykonuje przekształcenia obrazu (obroty, odbicia), tworząc obrazy w grafice rastrowej; stosuje filtry, m.in. poprawia ostrość obrazu, wykonuje proste fotomontaże, korzystając z warstw	stosuje filtry, m.in. wykonuje efekt zamglenia, usuwa szumy; wykonuje fotomontaże, korzystając z kilku warstw; wie, czym jest oderwane zaznaczenie i jak zakotwiczyć warstwę; przygotowuje animacje składające się z kilkunastu klatek, stosując pracę na warstwach; zapisuje animację w formacie GIF	samodzielnie tworzy rysunki, powiela warstwy i stosuje możliwości przekształceń, aby utworzyć kolejną klatkę animacji; wykorzystując dodatkowe możliwości programu GIMP, rozwiązuje różne zadania, edytując obrazy, tworząc fotomontaże korzystając z poznanych możliwości programu GIMP, projektuje dwuwymiarową wizualizację lub animację	wyszukuje w Internecie dodatkowe informacje na temat korzystania z filtrów w programie GIMP; wymyśla temat fotomontażu i przygotowuje animacje według własnych pomysłów, korzystając z różnych możliwości programu do tworzenia animacji; projektuje dwuwymiarową wizualizację lub animację, wykorzystując dodatkowe możliwości programu GIMP

Moduł C. Wokół Algorytmiki i Programowania

Prezentacja algorytmu liniowego w wybranej notacji				
2	3	4	5	6
Uczeń: zna sposoby prezentacji algorytmów; testuje działanie algorytmu liniowego zapisanego w postaci listy kroków; zapisuje prosty algorytm liniowy w wybranym języku programowania	Uczeń: zna pojęcie specyfikacji zadania i potrafi zapisać specyfikację zadania; zna i stosuje zasady tworzenia listy kroków; pisze programy komputerowe realizujące dane algorytmy na podstawie ich list kroków	Uczeń: przedstawia dokładną specyfikację dowolnego zadania; objaśnia dobrany algorytm, uzasadnia poprawność rozwiązania na wybranych przykładach danych; realizuje przykładowy algorytm liniowy w wybranym języku programowania na podstawie ich list kroków; testuje program dla różnych danych	Uczeń: potrafi przeprowadzić szczegółową analizę poprawności listy kroków; analizuje działanie algorytmu dla przykładowych danych; potrafi wskazać i poprawić błędy w liście kroków; samodzielnie pisze program realizujący algorytm liniowy	Uczeń: zapisuje specyfikację zadania i przedstawia w postaci listy kroków rozwiązanie trudniejszego zadania; w wybranym języku programowania (C++ lub Python) pisze trudniejsze programy realizujące algorytmy przedstawione w postaci list kroków; uczestniczy w konkursach/olimpiadach informatycznych

Prezentacja algorytmu z warunkami w wybranej notacji				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
określa sytuacje warunkowe; podaje przykłady zadań, w których występują sytuacje warunkowe; potrafi zapisać w liście kroków nietrudny algorytm z warunkiem prostym	analizuje działanie algorytmu z warunkami; zna i stosuje zasady tworzenia listy kroków algorytmu z warunkami; pisze program w wybranym języku programowania realizujący algorytm z warunkami prostymi; zna i stosuje instrukcję warunkową	przedstawia algorytm z warunkami w postaci listy kroków; analizuje listę kroków i schemat blokowy algorytmu z warunkami, testując go dla wybranych danych; potrafi zapisać warunek złożony; korzystając z przykładu, zapisuje w postaci programu algorytm z warunkami złożonymi; zna i omawia warunek istnienia trójkąta	zapisuje algorytmy z pętlą zagnieżdżoną; testuje działanie algorytmu z warunkami zagnieżdżonymi zapisanego w postaci listy kroków na wybranych przykładach danych; zapisuje w postaci programu algorytm z warunkami zagnieżdżonymi	wymyśla samodzielnie problem z warunkami zagnieżdżonymi, zapisuje jego specyfikację, listę kroków i program w wybranym języku programowania; korzystając z dodatkowych źródeł, znajduje inny, niż podany w podręczniku, sposób sprawdzenia, czy z danych trzech odcinków można zbudować trójkąt i zapisuje ten algorytm w postaci programu komputerowego; w wybranym języku programowania pisze trudniejsze programy realizujące algorytmy z warunkami (w tym zagnieżdżonymi) przedstawione w postaci list kroków

Prezentacja algorytmu iteracyjnego w wybranej notacji				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
analizuje listę kroków algorytmu iteracyjnego, testując go dla wybranych danych	zna pojęcie iteracji i rozumie pojęcie algorytmu iteracyjnego; podaje przykłady algorytmów iteracyjnych; tworzy algorytm z warunkiem prostym i pętlą; testuje rozwiązanie dla wybranych danych; zna i stosuje instrukcję iteracyjną for w wybranym języku programowania; zapisuje prosty algorytm iteracyjny w postaci programu w wybranym języku programowania	analizuje algorytmy, w których występują powtórzenia (iteracje); ocenia zgodność algorytmu ze specyfikacją; analizuje listę kroków algorytmu z pętlą zagnieżdżoną, testując go dla wybranych danych; zapisuje algorytm iteracyjny w postaci programu w wybranym języku programowania	zapisuje w postaci listy kroków algorytmów z warunkami i iteracyjne; pisze listę kroków algorytmu z pętlą zagnieżdżoną; ocenia zgodność algorytmu ze specyfikacją problemu; wie, kiedy należy zastosować pętlę zagnieżdżoną; zapisuje w postaci programu wybrany algorytm z pętlą zagnieżdżoną; testuje program dla wybranych danych	przestrzega zasad zapisu algorytmów w zadanej postaci (notacji); stosuje listy kroków w opisie zadań (problemów) z innych przedmiotów szkolnych oraz różnych dziedzin życia; wskazuje podobieństwa i różnice dotyczące działania instrukcji warunkowych i iteracyjnej for w różnych językach programowania; zapisuje trudniejsze algorytmy iteracyjne w wybranym języku programowania; zapisuje programy w czytelnej postaci – stosuje wcięcia, komentarze; uczestniczy w konkursach/olimpiadach informatycznych

Funkcje w wybranym języku programowania				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, czym są podprogramy; w wybranym języku programowania definiuje prostą funkcję niezwracającą wartości bez parametrów. wie, jak wywołać funkcję niezwracającą wartości bez parametrów w programie; wywołuje taką funkcję w programie	wie, na czym polega programowanie strukturalne; pisze programy, stosując funkcje; definiuje funkcje niezwracające wartości bez parametrów i z parametrami; stosuje te funkcje w programach; wyjaśnia pojęcia zmienna lokalna i zmienna globalna	wymienia modele programowania; rozumie i stosuje zasady programowania strukturalnego; wyjaśnia, czym są parametry funkcji; wyjaśnia, czym jest funkcja zwracająca wartość, porównuje do funkcji niezwracającej wartości i wskazuje różnice; wie, na czym polega wywołanie funkcji z parametrami i wywołuje taką funkcję w programach; definiuje funkcje zwracające wartości bez parametrów i z parametrami oraz stosuje je w programach; wie, co to jest zasięg zmiennej i jakie ma znaczenie; deklaruje odpowiednio zmienne lokalne i globalne w programach	omawia modele programowania; rozumie zasady postępowania przy rozwiązywaniu problemu metodą zstępującą; wyjaśnia, czym różni się programowanie zstępujące od wstępującego; definiuje funkcje niezwracające wartości i zwracające wartość bez parametrów i z parametrami oraz stosuje je w programach; wyjaśnia, na czym polega przesłanianie zmiennych globalnych; wyjaśnia, na czym polega przesłanianie parametrów funkcji; modyfikuje programy, stosuje zmienne globalne i lokalne, objaśnia otrzymane wyniki	porównuje modele programowania, wskazując różnice; sprawnie definiuje i stosuje funkcje niezwracające wartości i zwracające wartość bez parametrów i z parametrami w programach; rozwiązuje przykładowe zadania z matury i olimpiady informatycznej; potrafi, na przykładzie programu utworzonego według własnego pomysłu, wyjaśnić różnice w stosowaniu zmiennych lokalnych i globalnych, omówić zasięg zmiennych i przesłanianie zmiennych; potrafi, na przykładzie programu utworzonego według własnego pomysłu, wyjaśnić przesłanianie parametrów funkcji

Tablice w wybranym języku programowania				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
analizuje i omawia gotowe przykłady programów, w których są użyte tablice lub listy; potrafi wyjaśnić, dlaczego do rozwiązania niektórych zadań należy użyć tablic (list)	zna pojęcia: <i>tablica</i> , <i>lista</i> , <i>zmienna indeksowana</i> ; na bazie przykładów, deklaruje tablicę i/lub listę, wczytuje i wyprowadza elementy tablicy i/lub listy, definiując odpowiednie funkcje w wybranym języku programowania	rozdziela struktury danych proste i złożone oraz podaje ich przykłady; wczytuje i wyprowadza elementy tablicy i/lub listy; definiuje odpowiednie funkcje; wie, jak odwołać się do elementu tablicy i/lub listy; potrafi zastosować tablicę (listę) w zadaniach oraz modyfikować program, znaleźć błędy i je poprawić	rozumie, na czym polega dobór struktur danych do algorytmu i tworzy programy, dobierając odpowiednie struktury danych do programu; deklaruje tablicę dwuwymiarową w języku C++; definiuje odpowiednie funkcje; definiuje różne listy (w tym dwuwymiarowe) w języku Python; tworzy programy, dobierając odpowiednie struktury danych do programu	omawia podobieństwa i różnice w definiowaniu tablic w języku C++ i list w języku Python; stosuje w programach listy jednowymiarowe i dwuwymiarowe, odpowiednio dobierając określoną strukturę danych (tu: rodzaj listy) do algorytmu; rozwiązuje przykładowe zadania z konkursów i olimpiady informatycznej, w których należy zastosować tablice i/lub listy i funkcje; dobiera najlepszy algorytm i odpowiednie struktury danych do rozwiązania postawionego problemu

Stosowanie instrukcji iteracyjnych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
analizuje i omawia działanie gotowych programów zapisanych w wybranym języku programowania, zawierających instrukcję pętli for; analizuje i omawia działanie gotowych programów zapisanych w wybranym języku programowania, zawierających instrukcję pętli while	zna pojęcie iteracji i rozumie pojęcie algorytmu iteracyjnego, podaje ich przykłady; zna postać i działanie instrukcji iteracyjnej while w językach C++ i/lub Python i stosuje ją w tworzonych programach komputerowych	zna sposoby zakończenia iteracji, określa kroki iteracji; stosuje instrukcję while w programach komputerowych; w języku C++ stosuje instrukcję do...while w programach komputerowych; zna różne sposoby określania liczby iteracji w języku Python, np. poprzez powtarzanie poleceń zawartych wewnątrz pętli dla konkretnych wartości lub poprzez podanie liczby powtórzeń z zastosowaniem funkcji range()	rozumie różnicę pomiędzy instrukcją for a instrukcją while w wybranym języku programowania; modyfikuje programy, zamieniając pętlę for na while i odwrotnie; ocenia program po zmianie (styl, czytelność); pisze programy, stosując poznane instrukcje iteracyjne; dobiera odpowiednią technikę algorytmiczną i odpowiednie struktury danych do rozwiązywanego problemu	potrafi samodzielnie zastosować odpowiedni rodzaj instrukcji pętli w tworzonym programie; potrafi samodzielnie dobrać odpowiednią instrukcję while lub do...while; omawia podobieństwa i różnice w działaniu wszystkich omówionych instrukcji pętli w języku C++; sprawnie pisze trudniejsze programy, stosując poznane instrukcje iteracyjne; wymyśla samodzielnie problem iteracyjny, formułuje zadanie, pisze jego specyfikację i listę kroków oraz zapisuje w wybranym języku programowania

Iteracyjna realizacja wybranych algorytmów, w tym algorytmów wyszukiwania				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
potrafi odróżnić algorytm liniowy od algorytmu iteracyjnego; omawia algorytm znajdowania elementu najmniejszego oraz analizuje listę kroków; analizuje i omawia gotową listę kroków algorytmu Euklidesa w jednej z wersji, testuje algorytm na podstawie listy	zna przykładowe algorytmy na liczbach naturalnych; wie, na czym polega metoda wyszukiwania liniowego i przez połowienie; zapisuje algorytm znajdowania najmniejszego (największego) elementu w postaci programu; zna iteracyjną postać algorytmu Euklidesa w wersji z odejmowaniem; potrafi napisać listy kroków algorytmu Euklidesa w wersji z odejmowaniem; zna metodę „dziel i zwyciężaj”; zapisuje algorytm Euklidesa w wersji z odejmowaniem w postaci programu w wybranym języku programowania	potrafi omówić algorytm naiwny i optymalny jednoczesnego znajdowania największego i najmniejszego elementu w zbiorze; zna iteracyjną postać algorytmu Euklidesa z resztą z dzielenia, pisze listę kroków tego algorytmu; zapisuje algorytm Euklidesa w wersji z resztą z dzielenia w postaci programu w wybranym języku programowania; wyjaśnia na przykładzie różnicę między wersją algorytmu Euklidesa z odejmowaniem a wersją z resztą z dzielenia	określa liczbę porównań w algorytmie naiwnym i optymalnym znajdowania największego i najmniejszego elementu w zbiorze, porównuje otrzymane wyniki; pisze listę kroków algorytmu jednoczesnego znajdowania minimum i maksimum z wykorzystaniem metody dziel i zwyciężaj, zapisuje ten algorytm w postaci programu w języku C++ i/lub Python; omawia zastosowanie schematu Hornera do obliczania wartości wielomianu, pisze listę kroków tego algorytmu i pisze program realizujący algorytm obliczania wartości wielomianu według schematu Hornera; programując w/w algorytmy, definiuje odpowiednie funkcje, dobiera struktury danych, dba o stosowanie podstawowych zasad programowania	podaje przykłady problemów, w których można zastosować wyszukiwanie liniowe lub przez połowienie; pisze trudniejsze programy komputerowe, w których wykorzystuje poznane algorytmy; korzystając z dodatkowych źródeł, wyszukuje informacje o zastosowaniu metody „dziel i zwyciężaj” oraz pisze program według własnego pomysłu pokazujący zastosowanie tej metody; samodzielnie zapoznaje się ze schematem Hornera i zapisuje go w postaci programu, dobierając poprawne struktury danych; korzystając z dodatkowych źródeł, omawia przykłady zastosowań algorytmu Euklidesa i poznaje trudniejsze algorytmy, np. trwałego małżeństwa, problem ośmiu hetmanów, algorytm znajdowania liczb bliźniaczych, potrafi zapisać je w języku programowania

Elementy analizy algorytmów				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia własności algorytmów; potrafi przeanalizować przebieg prostego algorytmu zapisanego w postaci listy kroków dla przykładowych danych i ocenić w ten sposób jego poprawność	zna i omawia własności algorytmów; wie, kiedy algorytm jest poprawny; potrafi przeanalizować przebieg algorytmu (np. obliczania silni) zapisanego w postaci listy kroków dla przykładowych danych i ocenić w ten sposób jego poprawność; analizuje program wybranego algorytmu (np. obliczania silni) i ocenia jego poprawność	wie, jak sprawdzić, czy algorytm jest skończony; potrafi ocenić poprawność działania algorytmu i jego zgodność ze specyfikacją; określa liczbę prostych działań zawartych w algorytmie; określa liczbę prostych działań zawartych w algorytmie; potrafi poprawić program, który jest niepoprawny	wie, kiedy algorytm jest skończony; potrafi przeanalizować przebieg algorytmu zapisanego w postaci listy kroków dla przykładowych danych i ocenić w ten sposób jego skończoność; analizuje program realizujący wybrany algorytm i ocenia jego skończoność; modyfikuje program, aby działał poprawnie; sprawdza poznane własności algorytmów, rozwiązując zadania, m.in. uzasadnia skończoność algorytmu znajdowania największego wspólnego dzielnika (NWD) dwóch liczb naturalnych; oblicza liczbę operacji porównania w algorytmie wyboru minimum z tablicy zawierającej n losowo uporządkowanych liczb	potrafi samodzielnie ocenić poprawność i skończoność wybranych algorytmów; potrafi samodzielnie wymyśleć zadanie (problem), napisać specyfikację zadania, listę kroków i program realizujący to zadanie; korzysta samodzielnie z dodatkowej literatury fachowej; poznane dodatkowe możliwości wybranego języka programowania (np. standardowe funkcje) i stosuje w programach; rozwiązuje przykładowe zadania z olimpiady informatycznej

Moduł D. Wokół Internetu I Projektów

Praca w chmurze i 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; uczestniczy czynnie w projekcie grupowym, wykonując proste zadania, np. wprowadza dane do bazy i je aktualizuje	planuje temat projektu; bierze aktywny udział w pracy grupowej jako członek zespołu, gromadząc i selekcionując materiały do projektu; bierze udział w testowaniu projektu; uczestniczy w przygotowaniu dokumentacji projektu, korzystając z edytora tekstu, ewentualnie z wybranego szablonu	korzystając z chmury, potrafi udostępnić pliki, linki do folderu, umożliwić współdzielenie danego folderu; realizuje projekt na zadany (lub samodzielnie wybrany) temat zgodnie z etapami projektowania, przygotowuje dokumentację projektu; współpracuje w grupie, wykonując projekt na temat projektowania zakupu nowego zestawu komputerowego oraz oprogramowania dla ucznia szkoły ponadpodstawowej oraz wpływu trendów w historycznym rozwoju pojęć i metod informatyki oraz technologii na możliwości rozwoju grafiki komputerowej	korzysta z oprogramowania dostępnego w chmurze, m.in. tworzy dokumenty w edytorze tekstu i umieszcza w chmurze, współdzieli je z innymi współużytkownikami; przygotowuje wybrane zadanie szczegółowe zgodnie z etapami przygotowania projektu; gromadzi materiały i inne pomoce, opracowuje dokumentację projektu, wykorzystując m.in. możliwości pracy w chmurze; udostępnia pliki, linki do folderu; wyszukuje informacje na e-platformach do e-nauczania; prezentuje wykonane zadanie projektowe i inicjuje dyskusje	sprawnie posługuje się środowiskiem przeznaczonym do współpracy i realizacji projektów zespołowych, w tym środowiskiem w chmurze; potrafi pełnić funkcję koordynatora grupy; koordynuje wykonywanie zadań szczegółowych na poszczególnych etapach; zarządza folderami (współdzieleniem) i pracą nad dokumentami, w tym dokumentacją projektu; ustala sposób prezentacji projektu i wyznacza osobę (osoby) do prezentacji; w miarę możliwości współtworzy zasoby udostępniane na platformach do e-nauczania; przygotowuje projekt na wybrany przez siebie temat

KLASA III i IV

Moduł A. Wokół Komputera i Sieci Komputerowych

Sieci komputerowe – funkcje urządzeń i protokoły				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna terminologię sieciową (<i>sieć komputerowa, zasoby sieciowe, klient, serwer, switch, router, brama, konfiguracja sieciowa, protokoły sieciowe</i>); potrafi zamienić oktety zapisane w systemie dziesiętnym na system dwójkowy	potrafi wyznaczyć adresy hostów i maskę w podsieci; przedstawia strukturę adresu domenowego; zna podstawowe funkcje urządzeń sieciowych; wie do czego służy szafa rackowa	opisuje podstawowe funkcje urządzeń i protokoły stosowane w przepływie informacji, w tym DNS, DHCP; potrafi odnaleźć w systemie Windows plik <i>hosts</i> oraz dokonać jego modyfikacji; przeprowadza konfigurację routera wg instrukcji; potrafi dokonać montażu urządzeń w szafie rackowej	określa ustawienia sieciowe danego komputera i jego lokalizację w sieci (adres sieciowy, podsieci IPv4, protokoły DHCP, DNS i TCP, określa ustawienia sieciowe komputera); zna standardowe numery portów przypisane wybranym usługom; analizuje wydajność sieci komputerowej w tym bezprzewodowej korzystając z odpowiedniej do tego celu aplikacji	przygotowuje prezentację multimedialną z opisem podstawowych protokołów stosowanych w przepływie informacji i w zarządzaniu siecią; samodzielnie konfiguruje urządzenia sieciowe; zabezpiecza urządzenia sieciowe przed nieuprawnionym dostępem; samodzielnie wykonuje zadania dla zainteresowanych

Moduł B. Wokół Dokumentów Komputerowych

Stosowanie wybranych funkcji w arkuszu kalkulacyjnym				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
analizuje i realizuje algorytm liniowy w arkuszu kalkulacyjnym, stosując odpowiednie funkcje zna zastosowanie funkcji JEŻELI w arkuszu kalkulacyjnym; wymienia grupy funkcji arkusza kalkulacyjnego np. statystyczne, matematyczne, tekstowe; wie, czym są zależności funkcyjne w arkuszu kalkulacyjnym	potrafi podać i omówić przykłady prostych algorytmów liniowych; zapisuje prostą sytuację warunkową w arkuszu kalkulacyjnym; tworzy formułę w arkuszu kalkulacyjnym sprawdzając czy wprowadzona liczba jest liczbą pierwszą; stosuje wybrane funkcje arkusza (matematyczne, logiczne, statystyczne, tekstowe) do realizacji algorytmów; analizuje przykłady zastosowania funkcji SUMA, JEŻELI, SILNIA, PRAWY, FRAGMENT, TEKSTU, DŁ i stosuje je w zadaniach	analizuje realizację algorytmu z warunkami w arkuszu kalkulacyjnym; stosuje i analizuje formuły, sprawdzające czy dana liczba jest liczbą pierwszą; wymienia przykłady fraktali jak: śnieżynka Kocha, dywan i trójkąt Sierpińskiego, paprotka Bransleya; umie przeprowadzić kilka kroków „gry w chaos”; stosuje w zadaniach funkcje SUMA, JEŻELI i SILNIA, PRAWY, FRAGMENT, TEKSTU, DŁ w arkuszu kalkulacyjnym; tworzy formuły na obliczenie funkcji liniowej, kwadratowej, trygonometrycznej i logarytmicznej i tworzy odpowiednio wykresy tych funkcji	zapisuje warunki zagnieżdżone w arkuszu kalkulacyjnym; dobiera funkcje arkusza kalkulacyjnego w zależności od rozwiązywanego problemu, w tym rodzaju danych; samodzielnie szkicuje trójkąt Sierpińskiego; omawia algorytm „gry w chaos” w arkuszu kalkulacyjnym; samodzielnie realizuje ćwiczenia z wykorzystaniem wybranych wzorów matematycznych, np. silni i symbolu Newtona; wykorzystuje w rozwiązaniach funkcje matematyczne, tekstowe i statystyczne i sprawdza poprawność rozwiązania	stosuje zaawansowane funkcje arkusza do realizacji algorytmów liniowych, z warunkami i iteracyjnymi i rysowania fraktali; samodzielnie analizuje algorytm konstruowania paprotki Bransleya; samodzielnie tworzy arkusze kalkulacyjne i stosuje poznane funkcje zgodnie z ich przeznaczeniem; wykonuje zadania dla zainteresowanych

Więcej o tworzeniu i edytowaniu grafiki wektorowej				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna cechy grafiki wektorowej; wie, czym różni się grafika wektorowa od grafiki rastrowej; potrafi wymienić nazwy programów do tworzenia grafiki wektorowej; wyjaśnia czym jest ścieżka; tworzy proste kształty; zapisuje plik do podanego formatu	potrafi korzystać z narzędzi <i>Kaligrafia, Narzędzie węzłów</i> ; używa podstawowych skrótów klawiszowych; stosuje właściwe formaty do zapisu plików graficznych; wie, czym jest efekt ścieżki; wymienia znane operacje logiczne	korzystając z poznanych narzędzi, tworzy rysunek według podanej specyfikacji; zmienia kształty korzystając z tzw. węzłów edycyjnych; zamienia kształt w ścieżkę; stosuje efekty ścieżki do zmiany wyglądu obiektu; tworzy projekt graficzny według podanych założeń	samodzielnie modyfikuje kształt i zmienia jego właściwości; potrafi łączyć działanie kilku narzędzi w tworzeniu i modyfikacji obiektów; wykorzystuje efekty przekształceń do uzyskania kształtów według swojego pomysłu; wykonuje na obiektach operacje logiczne; samodzielnie tworzy projekt graficzny	tworzy i edytuje obrazy wektorowe z wykorzystaniem zaawansowanych funkcji programu; stosuje wybrane narzędzia edycji obrazu i narzędzia pomocnicze, m.in. wybrane operacje na ścieżkach, klonowanie i inne możliwości zaawansowanego modyfikowania obiektów; wykorzystuje program Inkscape i Tinkercad do drukowania modelu 3D

Moduł C. Wokół Algorytmiki I Programowania

Wybrane struktury danych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, czym są <i>typ danych</i> i <i>format danych</i>; wymienia podstawowe proste i złożone typy danych; zna sposoby wprowadzania i wyprowadzania danych w języku C++ i/lub Python; potrafi wytłumaczyć, czym są dane losowe; wyjaśnia, czym jest sprawdzanie poprawności danych w programie komputerowym	korzysta z wybranych zbiorów i podzbiorów typów w językach C++ i Python; potrafi określić do jakiego typu należą zadeklarowane zmienne w kodzie źródłowym; zna wybrane elementy formatowania wykorzystywane przez funkcję printf() i print; wymienia funkcje do generowania danych losowych w języku C++ i/lub Python; zna funkcje standardowe, które można użyć do sprawdzenia poprawności danych	stosuje operator sizeof() w języku C++ do określania liczby bajtów pamięci zajmowanych przez zmienne; stosuje funkcje sizeof() i type() w języku Python; używa wbudowanych funkcji; objaśnia sposób formatowania poszczególnych wyników wykorzystanych przez funkcję printf() i print; wie, do czego służą f-łańcuchy w języku Python; analizuje przykłady stosowania danych losowych; analizuje przykłady sprawdzające poprawność danych liczbowych w języku C++ i/lub Python	oblicza i wyświetla wielkości pamięci zajmowane przez zmienne wybranych typów; oblicza i wypisuje minimalne i maksymalne wartości dające się zapamiętać w zmiennej; samodzielnie tworzy program z wykorzystaniem formatowania wyprowadzanych wyników w języku C++ i/lub Python; korzysta z f-łańcuchów w języku Python; tworzy program symulacji rzutów sześcienną kostką do gry; tworzy programy sprawdzające poprawność wprowadzonych danych liczbowych w języku C++ i/lub Python	wykonuje zadania dla zainteresowanych; potrafi wybrać odpowiednie struktury danych do wykonania zadań; samodzielnie bez pomocy nauczyciela rozwiązuje ćwiczenia i zadania; przygotowuje program według własnego pomysłu wykorzystujący poznane struktury danych

Przetwarzanie plików				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wie, czym nazywamy strumień danych; wie, czym jest kod ASCII, zna liczbę pozycji w tabeli; wymienia nazwy innych stron kodowych; omawia przykłady błędów w programach komputerowych	zna cel wykorzystania metod seekp(), seekg() (w języku C++) oraz/lub metody seek() (w języku Python); zna funkcje do obsługi błędów w języku C++; zna konstrukcje do obsługi błędów w języku Python	wykorzystuje klasy i funkcje (zależnie od języka programowania) do odczytu i zapisu danych; analizuje gotowy program odczytujący dane z pliku i wyświetlający je na ekranie; analizuje gotowy program zapisujący dane do pliku; stosuje funkcje do obsługi błędów w języku C++ i/lub stosuje konstrukcje do obsługi błędów w języku Python	używa funkcji obsługi plików w programach komputerowych; analizuje gotowy program z obsługą błędów; samodzielnie modyfikuje programy zgodnie z treścią ćwiczeń	wykonuje zadania dla zainteresowanych; potrafi wybrać odpowiednie struktury danych do wykonania zadań; samodzielnie bez pomocy nauczyciela rozwiązuje ćwiczenia i zadania

Sito Eratostenesa, rozkład liczby na czynniki pierwsze oraz znajdowanie w ciągu podciągów				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna i potrafi wytłumaczyć algorytm badania pierwszości liczby metodą sita Eratostenesa; omawia rozkład liczby na czynniki pierwsze; zna definicję podciągu, podciągu spójnego i podaje przykłady	wskazuje i nazywa konstrukcje programistyczne wykorzystane w przykładach programów generowania liczb pierwszych metodą sita Eratostenesa; analizuje opis algorytmu rozkładu liczby na czynniki pierwsze; podaje przykłady podciągów spójnych; zna definicję liczby doskonałej	analizuje realizację algorytmu generowania liczb pierwszych metodą sita Eratostenesa w języku C++ i/lub Python; potrafi wytłumaczyć korzystając z listy kroków algorytm wyszukania liczby doskonałej; planuje znajdowanie najdłuższego spójnego podciągu niemalejącego; planuje znajdowanie spójnego podciągu o największej sumie	tworzy programy komputerowe w językach C++ i/lub Python wyznaczające liczby pierwsze metodą sita Eratostenesa; tworzy programy komputerowe rozkładu liczby na czynniki pierwsze w języku C++ i/lub Python; samodzielnie sprawdza, czy dana liczba jest liczbą doskonałą; pisze i modyfikuje programy znajdujące najdłuższy spójny podciąg niemalejący oraz o największej sumie	wykonuje zadania dla zainteresowanych, w tym znajdowanie liczb bliźniaczych; potrafi wybrać odpowiednie struktury danych do wykonania zadań; przygotowuje program według własnego pomysłu wykorzystujący poznane struktury danych

Stosowanie metody zachłannej i metody połowienia				
2	3	4	5	6
Uczeń: omawia definicję algorytmu zachłannego; podaje przykłady wykorzystania algorytmu zachłannego	Uczeń: tłumaczy algorytm wydawania reszty metodą zachłanną; wie, na czym polega algorytm połowienia i demonstruje ten algorytm na przykładzie	Uczeń: analizuje na konkretnym przykładzie algorytm wydawania reszty metodą zachłanną; analizuje funkcje realizujące algorytm wydawania reszty metodą zachłanną i wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia	Uczeń: analizuje programy realizujące algorytm zachłanny wydawania reszty i wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia w języku C++ i/lub Python; objaśnia oraz porównuje podstawowe metody i techniki algorytmiczne oraz struktury danych zastosowane w w/w programach	Uczeń: pisze program realizujący algorytm wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia w języku C++ i/lub Python; znajduje błędy w wykonanym programie i je poprawia wykonuje zadania dla zainteresowanych; przygotowuje program według własnego pomysłu wykorzystujący poznane techniki programistyczne

Szyfrowanie				
2	3	4	5	6
Uczeń: wie, czym jest szyfrowanie z kluczem; zna różnice między algorytmem szyfrowania symetrycznego a algorytmem szyfrowania asymetrycznego	Uczeń: omawia schemat szyfrowania z kluczem; potrafi scharakteryzować szyfr Vigenère'a; wymienia zastosowanie szyfrowania w życiu codziennym	Uczeń: analizuje przykład stosowania szyfru Vigenère'a; określa wady i zalety szyfrowania z kluczem jawnym	Uczeń: dokonuje szyfrowania szyfrem Vigenere'a; opisuje wykorzystanie algorytmów szyfrowania w podpisie elektronicznym; samodzielnie wykonuje zadania	Uczeń: przygotowuje prezentację multimedialną dotyczącą algorytmu RSA; wykonuje zadania dla zainteresowanych

Moduł D. Wokół Internetu i Projektów

Zadania projektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
potrafi wyjaśnić cel projektu; określa propozycję zadań szczegółowych; wykonuje proste elementy projektu	potrafi wyszukać informacje potrzebne do wykorzystania w projekcie; wykonuje poszczególne zadania szczegółowe; tworzy slajdy w prezentacji i uzupełnia treścią	dba o sprawdzenie części merytorycznej i technicznej projektu; wybiera odpowiednie narzędzia; rozszerza slajdy prezentacji o elementy multimedialne	koordynuje prace zespołu projektowego; zarządza czasem wykonania poszczególnych etapów projektu; łączy zadania szczegółowe w jeden plik; prezentuje fragment projektu	samodzielnie prezentuje cały projekt; odpowiada na pytania; prowadzi dyskusję