

KLASA 1

2	3	4	5	6
Urządzenia komputerowe w sieci				
<ul style="list-style-type: none"> wymienia urządzenia mobilne zaliczane do systemów komputerowych, wymienia elementy budowy systemu operacyjnego, rozumie pojęcie „ścieżka dostępu” w kontekście systemów plików, sprawdza i wymienia atrybuty pliku, opisuje, jak uruchomić system BIOS na komputerze, wyjaśnia konieczność tworzenia bezpiecznych haseł, wymienia metody zabezpieczania danych na komputerze, uruchamia Menedżera zadań w systemie Windows, wymienia problemy, jakie można napotkać podczas korzystania z komputera, wyjaśnia pojęcie sztucznej inteligencji, opisuje, czym jest chmura obliczeniowa, wymienia zastosowania automatów i robotów, podaje przykłady wykorzystania druku 3D, 	<ul style="list-style-type: none"> wymienia urządzenia wchodzące w skład sieci komputerowej, identyfikuje wersję systemu operacyjnego swojego smartfona (komputera), wyjaśnia różnicę między bezwzględną i względną ścieżką dostępu, określa różnicę pomiędzy BIOS a UEFI, rozumie pojęcie serwera, opisuje zasady bezpiecznego korzystania z systemu operacyjnego, wyjaśnia, jak założyć konto użytkownika w używanym przez siebie systemie operacyjnym, konstruuje bezpieczne hasła, kopiuje dane, aby wykonać kopię zapasową zewnętrznym nośniku, uruchamia komputer w trybie awaryjnym, sprawdza obciążenie procesora, wyjaśnia pojęcia fragmentacji i defragmentacji dysku, 	<ul style="list-style-type: none"> opisuje, czym jest model warstwowy systemu komputerowego, wymienia i wyjaśnia zadania systemu operacyjnego, określa różnicę pomiędzy trybem jądra a trybem użytkownika, instaluje i aktualizuje oprogramowanie, podczas zamykania aplikacji umiejętnie korzysta z Menedżera zadań w systemie Windows, korzysta z narzędzi oczyszczania dysku, opisuje procedurę wykonywania kopii zapasowej dla systemu operacyjnego w szkolnej pracowni, opisuje zastosowania rzeczywistości wirtualnej i rozszerzonej dobiera kryteria wyboru elementów zestawu komputerowego w zależności od jego przeznaczenia, wymienia i opisuje urządzenia sieciowe, 	<ul style="list-style-type: none"> opisuje każdą z warstw modelu systemu komputerowego, charakteryzuje poszczególne elementy systemu operacyjnego, opisuje działanie systemu operacyjnego, modyfikuje uprawnienia konta użytkownika systemu operacyjnego, wykonuje defragmentację dysku, wymienia i opisuje zastosowania sieci internet, projektuje rozbudowę zestawu komputerowego zależnie od jego przeznaczenia i możliwości finansowych, charakteryzuje różne topologie sieci komputerowych, wyjaśnia pojęcie i budowę ramki jako porcji informacji w transmisji danych, opisuje sposób adresowania urządzeń w sieci internet, wyjaśnia sposób komunikacji między 	<ul style="list-style-type: none"> obsługuje różne systemy operacyjne, korzysta z poleceń trybu tekstowego Windows, kopiuje pliki w trybie tekstowym Windows za pomocą ścieżek względnych i bezwzględnych, dokonuje istotnych zmian w BIOS, wyjaśnia zasadę działania sztucznego neuronu i sieci neuronowej, projektuje modele warstwowe skomplikowanych procesów życia codziennego,

<ul style="list-style-type: none"> • zna i opisuje zagrożenia wynikające z rozwoju technologii, • określa przeznaczenie projektowanego zestawu komputerowego, • wyjaśnia pojęcia: „sieci komputerowe” i „urządzenia sieciowe”, • opisuje przeznaczenie warstwowych modeli sieci, • wyjaśnia przeznaczenie protokołu IP, • wyjaśnia pojęcie sztucznej inteligencji, • opisuje, czym jest chmura obliczeniowa, • wymienia zastosowania automatów i robotów, • podaje przykłady wykorzystania druku 3D, • zna i opisuje zagrożenia wynikające z rozwoju technologii, 	<ul style="list-style-type: none"> • wyjaśnia różnicę pomiędzy systemami plików FAT32 oraz NTFS, • definiuje pojęcie systemu operacyjnego, • wyjaśnia różnicę pomiędzy wirtualną a rozszerzoną rzeczywistością • projektuje zestaw komputerowy według podanych kryteriów, • nazywa różne porty urządzeń sieciowych, • wymienia korzyści wynikające z korzystania z warstwowych modeli sieci, • opisuje budowę adresu IPv4 w wersjach dziesiętnej i binarnej, • rozróżnia typy domen (krajowe, funkcjonalne), • wyjaśnia pojęcie systemu DNS, • opisuje budowę adresu URL, • wyjaśnia, czym są e-usługi, 	<ul style="list-style-type: none"> • opisuje sieci komputerowe ze względu na zasięg ich działania, • wyjaśnia budowę adresów MAC i sprawdza je na komputerze z systemem Windows, • wyjaśnia pojęcia: adres IP, maska podsieci, • opisuje modele klient–serwer oraz peer-to-peer, • schematycznie przedstawia i omawia model warstwowy TCP/IP, • schematycznie przedstawia i omawia model warstwowy OSI, • określa relacje między podmiotami rynku e-usług, • korzysta z wybranych e-usług, 	<p>urządzeniami tej samej oraz różnych sieci,</p> <ul style="list-style-type: none"> • omawia procesy enkapsulacji i dekapulacji danych w transmisji sieciowej, • opisuje protokoły wykorzystywane podczas transmisji danych w sieci, • podłącza i konfiguruje urządzenia sieciowe, • projektuje domową sieć komputerową, • opisuje sposób tworzenia i budowę domeny internetowej, • konfiguruje urządzenie do pracy w internecie i omawia ten proces, • wymienia i omawia protokoły usług internetowych, • diagnozuje stan połączeń internetowych, 	
Edytor tekstu i prezentacje				
<ul style="list-style-type: none"> • korzysta w podstawowym zakresie z formatowania tekstów w edytorze tekstowym, • wymienia etapy pracy nad dobrym wystąpieniem publicznym, 	<ul style="list-style-type: none"> • korzysta z szablonów w edytorze tekstów, • poprawnie stosuje style nagłówkowe, • generuje losowe bloki tekstowe, 	<ul style="list-style-type: none"> • tworzy i modyfikuje własne szablony oraz style tekstowe, • dzieli tekst na kolumny, • pracuje z wielostronicowym dokumentem w widoku konspektu, 	<ul style="list-style-type: none"> • korzysta z automatycznej numeracji tytułów oraz tworzy spis treści, • tworzy spisy ilustracji i tabel, • pracuje z dokumentem wspólnie z innymi osobami, 	<ul style="list-style-type: none"> • korzysta z różnych narzędzi (w tym mobilnych) podczas prezentacji, • bierze udział w projektach zespołowych

<ul style="list-style-type: none"> wymienia programy komputerowe do tworzenia prezentacji, wyjaśnia pojęcia: wykluczenie i włączenie cyfrowe, 	<ul style="list-style-type: none"> ustawia marginesy w dokumencie 	<ul style="list-style-type: none"> wymienia cechy dobrej prezentacji, tworzy ciekawe przejścia między slajdami, 	<p>korzystając z narzędzi pracy grupowej,</p> <ul style="list-style-type: none"> wykorzystuje opcje recenzji dokumentu, wygłasza prelekcję na wybrany temat zgodnie z zasadami dobrego wystąpienia, tworzy dokładny plan wystąpienia na dowolny temat, stosuje efekty na slajdach prezentacji, umieszcza filmy i ścieżki audio w prezentacji, prezentuje kompletny projekt na forum klasy, 	<p>jako odpowiedzialny lider projektu,</p> <ul style="list-style-type: none"> wypełnia wszystkie zadania wynikające z roli powierzonej mu w projekcie,
---	--	---	--	---

Społeczeństwo w internecie

<ul style="list-style-type: none"> wyjaśnia pojęcie cyfrowej tożsamości, wymienia sposoby uwierzytelniania użytkowników e-usług, wskazuje miejsca występowania e-zasobów, rozdziela wyszukiwarki od przeglądarek internetowych, podaje przykłady negatywnych zachowań w sieci internet, wyjaśnia znaczenie kryptografii dla bezpieczeństwa danych, 	<ul style="list-style-type: none"> wyjaśnia pojęcia: prawo autorskie, licencja, rozdziela i definiuje pojęcia wolnego i otwartego oprogramowania, wyjaśnia pojęcie licencji Creative Commons, wymienia wiarygodne źródła informacji w sieci internet, wyjaśnia, jak sprawdzić właściciela serwisu internetowego, omawia pojęcia związane z kryptografią, 	<ul style="list-style-type: none"> podaje cechy różnych rodzajów licencji oprogramowania, stosuje symbole i wyrażenia w wyszukiwarkach internetowych, wymienia zasady ochrony danych osobowych, opisuje zastosowania technologii komputerowej w różnych dziedzinach życia, opisuje rodzaje ataków sieciowych, omawia działanie protokołu SSL, 	<ul style="list-style-type: none"> wyjaśnia zasady stosowania prawa autorskiego, wykorzystuje narzędzia współpracy zdalnej, wyjaśnia, jak zwiększyć swoje bezpieczeństwo w sieci poprzez stosowanie różnych technik, omawia kryptoanalizę na wybranym przez siebie przykładzie, 	<ul style="list-style-type: none"> wyjaśnia, czym są: steganografia, kryptologia, kryptografia, kryptogram, kryptoanaliza prezentuje przykład zaawansowanej kryptoanalizy
--	--	---	---	---

	<ul style="list-style-type: none"> • opisuje pojęcie cyfrowej tożsamości, • wymienia zasady komunikacji w sieci internet (netykieta), • wymienia zagrożenia wynikające ze złej komunikacji w sieci, • opisuje wpływ rozwoju technologii na zmiany w społeczeństwie, • wymienia i opisuje rodzaje szkodliwego oprogramowania 			
Strony WWW i grafika komputerowa				
<ul style="list-style-type: none"> • zapisuje plik, nadając mu rozszerzenie .html, • rozróżnia sekcje HEAD i BODY oraz opisuje różnicę między tymi częściami kodu, • wymienia podstawowe znaczniki formatowania tekstu w języku HTML, • opisuje budowę znacznika HTML, • wyjaśnia pojęcie responsywności strony WWW, • uruchamia stronę WWW na smartfonie, • określa różnicę pomiędzy grafiką rastrową a wektorową, 	<ul style="list-style-type: none"> • wyjaśnia, czym są e-zasoby, • tworzy stronę tytułową w dokumencie tekstowym, • wyjaśnia, jak przygotować dobre wystąpienie, • zna narzędzia, dzięki którym można dobrać zestaw pasujących do siebie kolorów, • opisuje podstawową strukturę strony w języku HTML, • tworzy nagłówki w języku HTML, • wstawia komentarze w kodzie HTML, • tworzy listy uporządkowane i nieuporządkowane, • rozumie cel pozycjonowania stron WWW, 	<ul style="list-style-type: none"> • tworzy modele 3D z prostych brył 3D i ich przekształceń, • umieszcza zdjęcia na stronie WWW, • tworzy linki do zasobów zewnętrznych oraz miejsc w obrębie jednej strony, • poprawnie i na różne sposoby korzysta z opisu kolorów w języku HTML, • wyjaśnia działanie hostingu stron internetowych, • wykorzystuje język JavaScript podczas tworzenia stron internetowych • wymienia podstawowe narzędzia programu GIMP, • korzysta z warstw podczas pracy z programem GIMP, 	<ul style="list-style-type: none"> • korzysta ze ścieżek względnych i bezwzględnych w kodzie HTML, • poprawnie tworzy tabele o dowolnej strukturze, • dołącza style kaskadowe do dokumentu HTML, • tworzy ciekawą stronę WWW i publikuje ją w internecie, • poprawnie używa narzędzia do rysowania krzywych Béziera, • wycina dowolne elementy z obrazu rastrowego, • tworzy w programach do grafiki wektorowej infografiki według wzoru, 	<ul style="list-style-type: none"> • tworzy style opisujące wygląd strony WWW, • dodaje do strony elementy odpowiedzialne za jej responsywność, • buduje stronę z wykorzystaniem systemu CMS i publikuje ją w internecie, • tworzy złożone modele 3D.

<ul style="list-style-type: none">• zapisuje wynik swojej pracy w różnych formatach graficznych,• wyjaśnia, jak uruchomić środowisko do grafiki 3D online,	<ul style="list-style-type: none">• skaluje i kadruje obraz, dostosowując go do zadanego rozmiaru,• wymienia podstawowe narzędzia programu Inkscape,• tworzy dwuwymiarowe animacje,	<ul style="list-style-type: none">• pracuje na warstwach w programie do grafiki wektorowej,	<ul style="list-style-type: none">• tworzy bryły obrotowe 3D na podstawie ich przekroju,• tworzy trójwymiarowe animacje,	
---	---	---	---	--

KLASA 2

2	3	4	5	6
Arkusz kalkulacyjny				
<ul style="list-style-type: none"> wprowadza dane różnego typu do arkusza kalkulacyjnego, omawia zastosowania korespondencji seryjnej, wyjaśnia relacje w bazach danych. 	<ul style="list-style-type: none"> pobiera dane do arkusza kalkulacyjnego ze źródeł zewnętrznych, filtruje dane w arkuszu kalkulacyjnym, tworzy różne wykresy w arkuszu kalkulacyjnym w zależności od rodzaju danych, bierze udział w projektach informatycznych jako członek zespołu. 	<ul style="list-style-type: none"> przeprowadza analizę danych zgromadzonych w arkuszu kalkulacyjnym, omawia błąd zaokrąglenia i błąd przybliżenia w obliczeniach komputerowych, dobiera środowisko informatyczne do rodzaju rozwiązywanego problemu, wyszukuje informacje zgromadzone w bazach danych, w bazach danych wykorzystuje kwerendy, filtrowanie, formularze i raporty, usprawnia pracę, wykorzystując makropolecenia VBA. 	<ul style="list-style-type: none"> wykorzystuje zaawansowane formuły, opracowując dane w arkuszu kalkulacyjnym, stosuje funkcje zaokrąglające liczby, korzysta z możliwości obliczeń walutowych, 	
Algorytmy na liczbach całkowitych i tekstach				
<ul style="list-style-type: none"> definiuje podstawowe pojęcia z algorytmiki i programowania: algorytm, program, warunek, iteracja, rekurencja, wymienia sposoby reprezentacji algorytmów, 	<ul style="list-style-type: none"> przedstawia krótkie algorytmy w postaci listy kroków, opisu słownego, pseudokodu, schematu blokowego, dodaje liczbybinarne, 	<ul style="list-style-type: none"> określa specyfikację algorytmu (dane, wynik), pisze programy o różnym stopniu trudności, szacuje ich efektywność, przedstawia omawiane algorytmy w postaci opisu 	<ul style="list-style-type: none"> charakteryzuje sytuacje algorytmiczne, proponuje sposoby ich rozwiązania, pisze programy o podwyższonym stopniu trudności: oznaczone 	<ul style="list-style-type: none"> charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z

<ul style="list-style-type: none"> • korzysta ze środowiska programistycznego: pisze w nim kod, kompiluje i uruchamia program, odczytuje i zapisuje pliki, • pisze programy o niewielkim stopniu trudności, • omawia pojęcia: złożoność obliczeniowa algorytmu, algorytm naiwny, algorytm optymalny, złożoność pesymistyczna, złożoność oczekiwana (średnia), • korzysta z podstawowych funkcji języka: operacji wejścia i wyjścia, instrukcji warunkowych i iteracyjnych, gotowych funkcji bibliotecznych, • wymienia podstawowe typy danych, operacje arytmetyczne i logiczne, • definiuje pojęcie systemów liczbowych, • wyjaśnia, czym jest tablica kodów ASCII, • wymienia systemy liczbowe używane w informatyce, • konwertuje liczby między systemami binarnym i dziesiętnym, • dodaje pisemnie liczby binarne, • wyjaśnia, czym są palindrom i anagram, podaje przykłady, 	<ul style="list-style-type: none"> • konwertuje liczby między pozycyjnymi systemami liczbowymi, • wykonuje działania arytmetyczne na liczbach w systemach liczbowych o różnych podstawach, • przedstawia liczby w kodzie U2, • definiuje pojęcie zdania logicznego, charakteryzuje podstawowe operacje logiczne (koniunkcja, alternatywa, negacja) oraz operatory logiczne, • charakteryzuje wybrane typy zmiennych służących do zapisu liczb całkowitych w języku C++: shortint, int, longint, longlongint, • pisze programy wykonujące działania na liczbach całkowitych, • korzysta z biblioteki string do operacji na łańcuchach znaków, • wykonuje operacje na napisach, wykorzystując słowa kluczowe: size, find, substr, erase, toupper, tolower, • wczytuje napisy ze spacjami, wykorzystując słowo kluczowe getline, 	<p>słownego, listy kroków, schematu blokowego, pseudokodu,</p> <ul style="list-style-type: none"> • dobiera typy danych do realizacji problemu, • stosuje zmienne typu unsigned w tworzonych programach, • pisze programy konwertujące liczby między systemem dziesiętnym i binarnym, • implementuje w języku C++ algorytmy wykonujące działania arytmetyczne na liczbach w różnych systemach, • w algorytmach zamiany wykorzystuje zależności między systemami binarnym, ósemkowym i heksadecymalnym, • omawia sposób reprezentacji obrazów w komputerze, korzystając z takich pojęć jak: piksel, model RGB, kanał alfa, • wyjaśnia, na czym polega digitalizacja (dyskretyzacja) dźwięku, • wyjaśnia zasadę tworzenia animacji, • implementuje w języku C++ algorytmy 	<p>trzema gwiazdkami w podręczniku,</p> <ul style="list-style-type: none"> • optymalizuje rozwiązania, • pisze programy konwertujące liczby między różnymi systemami pozycyjnymi, • w programach wykonujących działania na liczbach w różnych systemach pozycyjnych wykorzystuje bibliotekę string i strukturalne typy danych, • wykorzystuje rozwinięcie binarne liczb dziesiętnych w algorytmie szybkiego podnoszenia do potęgi, • wykonuje operacje arytmetyczne na liczbach w różnych systemach, implementuje je w języku C++, • stosuje różne sposoby przekazywania parametrów do funkcji, uzasadnia ich użycie, • pisze funkcje typu logicznego, np. sprawdzającą, czy napis jest palindromem, • szyfruje dane wczytane z pliku z uwzględnieniem polskich znaków diakrytycznych, 	<p>zastosowaniem złożonych struktur danych i biblioteki STL języka C++,</p> <ul style="list-style-type: none"> • pisze programy o wysokim stopniu trudności: z olimpiad przedmiotowych, konkursów informatycznych lub oznaczone trzema gwiazdkami w podręczniku, • wyszukuje palindromy lub anagramy w plikach tekstowych, • tworzy palindromy z napisów, dopisując minimalną liczbę znaków, • pisze program rozkładający liczbę złożoną na dwie liczby pierwsze (hipoteza Goldbacha), • implementuje w języku C++ algorytm Euklidesa, stosując iterację i rekurencję, • pisze programy szyfrujące i deszyfrujące z wykorzystaniem zaawansowanych
--	--	--	--	---

<ul style="list-style-type: none"> • podaje definicje liczby pierwszej i liczby złożonej, • implementuje w języku C++ algorytm zliczający dzielniki danej liczby, • omawia geometryczną interpretację algorytmu Euklidesa, 	<ul style="list-style-type: none"> • tworzy algorytmy sprawdzające, czy napis jest palindromem, • wyjaśnia różnicę między parametrami formalnym i aktualnym, a także między zmiennymi lokalną i globalną, • implementuje w języku C++ algorytm naiwny sprawdzający, czy liczba jest pierwsza, • implementuje w języku C++ algorytm Euklidesa w wersjach z dzieleniem i odejmowaniem, 	<p>sprawdzające, czy napis jest palindromem,</p> <ul style="list-style-type: none"> • implementuje w języku C++ i optymalizuje algorytm sprawdzający, czy liczba jest pierwsza, • pisze program rozkładający liczby na czynniki pierwsze, • stosuje w programach algorytm Euklidesa do obliczenia NWD i NWW, • wykorzystuje algorytm Euklidesa do działań na ułamkach, • szyfruje dane wczytane z pliku tekstowego, • zapisuje w postaci programu rozszerzony algorytm Euklidesa, wyjaśnia jego działanie i zastosowanie, 	<ul style="list-style-type: none"> • pisze program odczytujący informację ukrytą za pomocą szyfru Cezara z wykorzystaniem analizy częstości znaków w tekście, • wykorzystuje algorytm Euklidesa do działań na ułamkach, stosując struktury lub pary (typ pair), 	<p>szyfrów (np. permutacyjny lub Vigenere'a) i różnych kluczy,</p>
<p>Rozwiązywanie problemów z wykorzystaniem struktur danych</p>				
<ul style="list-style-type: none"> • w pisanych programach korzysta ze strukturalnych typów danych: napisów, struktur, tablic, • definiuje pojęcia: kryptologia, kryptografia, kryptoanaliza, tekst jawny, klucz, szyfrogram, 	<ul style="list-style-type: none"> • przedstawia w postaci algorytmu problem wyszukiwania anagramów, • przy pisaniu programów stosuje własne funkcje różnych typów, w tym funkcję typu void, • pisze program szyfrujący napis szyfrem Cezara, 	<ul style="list-style-type: none"> • stosuje różne sposoby przekazywania parametrów do funkcji: przez wartość, referencję lub wskaźnik, • pisze programy sprawdzające, czy dwa napisy są anagramami, wykorzystując funkcję sort z biblioteki STL, 	<ul style="list-style-type: none"> • dobiera struktury danych i metody do rodzaju problemu, • sprawdza, czy napisy są anagramami, stosując sortowanie lub zliczanie znaków, • przy testowaniu liczby na pierwszość stosuje funkcję typu logicznego, 	<ul style="list-style-type: none"> • stosuje w programach algorytmy sortowania inne niż omawiane na lekcjach (np. heapsort), • w projektach zespołowych przyjmuje rolę lidera.

<ul style="list-style-type: none">• rozróżnia szyfry podstawieniowe i przestawieniowe,• omawia szyfr Cezara jako przykład szyfru podstawieniowego i szyfr kolumnowy jako przykład szyfru przestawieniowego,• wyjaśnia, na czym polega łamanie szyfru,• omawia algorytm zliczania znaków w tekście,	<ul style="list-style-type: none">• omawia algorytm sita Eratostenesa,• przedstawia algorytm znajdowania spójnych podciągów, wyznaczania najdłuższego z nich oraz podciągu o największej sumie elementów,• omawia algorytm znajdowania idola i lidera w zbiorze,	<ul style="list-style-type: none">• stosuje algorytm wyszukiwania binarnego i oszacowuje jego złożoność czasową,• pisze programy sortujące metodami prostymi z zastosowaniem funkcji typu void,• pisze program realizujący algorytm sita Eratostenesa,	<ul style="list-style-type: none">• wyszukuje liczby bliźniacze,	
---	--	--	--	--

KLASA 3

2	3	4	5	6
Rozwiązywanie problemów z wykorzystaniem struktur danych				
<ul style="list-style-type: none"> wypisuje liczby pierwsze z zadanego przedziału, stosując metodę sita Eratostenesa, wyszukuje w ciągu liczb spójne podciągi (nierosnący, niemalejący, stały), wskazuje najdłuższe, oblicza ich sumę, wskazuje idola i lidera w zbiorze danych, 	<ul style="list-style-type: none"> omawia algorytm zliczania znaków w tekście oraz wyszukujący maksimum z wykorzystaniem tablic, przedstawia w postaci listy kroków lub schematu blokowego algorytm sortowania prostego (bąbelkowe, przez wybieranie) oraz szybkiego i przez scalanie, określa operacje dominujące, implementuje w języku C++ algorytm rekurencyjne: obliczanie elementów ciągu Fibonacciego, wartości silni i potęgi, omawia rozszerzony algorytm Euklidesa, formułuje algorytm wydawania reszty minimalną liczbą monet, harmonogramu wykorzystania sali, pakowania plecaka, znajdowania drogi metodami zachłanną i dynamiczną, uczestniczy w realizacji projektu zespołowego, 	<ul style="list-style-type: none"> implementuje w języku C++ algorytm zliczania znaków w tekście oraz wyszukujący maksimum z wykorzystaniem tablic, implementuje w języku C++ algorytm wyszukujące spójne podciągi o różnych cechach, stosuje w programach algorytm wyszukiwania lidera i idola w zbiorze, 	<ul style="list-style-type: none"> stosuje zaawansowane funkcje środowiska i języka programowania (np. z biblioteki STL), optymalizuje program realizujący algorytm sita Eratostenesa i szacuje jego złożoność czasową, wyszukuje spójne podciągi w plikach tekstowych, stosując optymalne algorytm (w tym programowanie dynamiczne), wyjaśnia ich działanie, pisze programy wyszukujące lidera i idola w zbiorze, optymalizuje je, szacuje złożoność czasową, 	<ul style="list-style-type: none"> stosuje zaawansowane algorytm i struktury danych do wyszukiwania spójnych podciągów,

	wykonując powierzone mu zadania.			
Metody algorytmiczne				
<ul style="list-style-type: none"> • wyjaśnia, na czym polega metoda „dziel i zwyciężaj”, • wczytuje dane z pliku tekstowego, zapisuje wyniki w pliku, • omawia algorytmy wyszukiwania liczby w zbiorach uporządkowanym i nieuporządkowanym, • stosuje funkcję losującą w tworzonych programach, • omawia metody sortowania prostego (bąbelkowe, przez wybieranie) oraz szybkiego i przez scalanie na przykładowych danych, • definiuje pojęcia iteracji i rekurencji, • omawia zasadę złotego podziału, • opisuje rozszerzony algorytm Euklidesa, • omawia metody zachłanne na 	<ul style="list-style-type: none"> • implementuje w języku C++ algorytmy wyszukiwania liniowego i liniowego z wartownikami, porównuje ich efektywność, 	<ul style="list-style-type: none"> • stosuje algorytmy sortowania szybkiego i przez scalanie, • stosuje metodę zachłanną w programach – problem kasjera, harmonogram wykorzystania sali, wyszukiwanie drogi, pakowanie plecaka, • uczestniczy w realizacji projektu zespołowego na wszystkich jego etapach, bierze czynny udział w tworzeniu dokumentacji projektowej. • porównuje algorytmy iteracyjne i rekurencyjne (liczbę wykonywanych operacji), szacuje ich złożoność czasową, 	<ul style="list-style-type: none"> • pisze program wyszukiwujący jednocześnie minimum i maksimum w zbiorze z wykorzystaniem metody „dziel i zwyciężaj” oraz podaje wzór na liczbę wykonywanych operacji, • szacuje złożoność obliczeniową programów sortujących, modyfikuje funkcje sortujące, zmieniając porządek sortowania, • wykorzystuje poznane algorytmy do rozwiązywania problemów nieomawianych na lekcjach, • pisze programy obliczające liczbę operacji przenoszenia krążków w problemie wież Hanoi, stosując iterację i rekurencję, • do implementacji rozszerzonego algorytmu Euklidesa stosuje zarówno iterację, jak i rekurencję, • stosuje metody zachłanną i dynamiczną w problemach 	<ul style="list-style-type: none"> • implementuje w języku C++ algorytm wyszukiwania binarnego w wersji rekurencyjnej, • pisze programy sortujące dane różnego typu w plikach tekstowych (liczby, napisy, pary), • stosuje zaawansowane algorytmy wyszukiwania, np. najlepszego wyboru (trwałych par), stosując rekurencję, • pisze programy obliczające liczbę operacji przenoszenia krążków w problemie wież Hanoi, stosując iterację i rekurencję,

<p>przykładzie problemu kasjera, harmonogramu sali, pakowania plecaka i wyszukiwania drogi,</p> <ul style="list-style-type: none"> • porównuje metody zachłanną i dynamiczną, • uczestniczy w realizacji projektu zespołowego, wykonując powierzone mu zadania o niewielkim stopniu trudności. 			<p>kasjera, harmonogramu wykorzystania sali, pakowania plecaka i wyszukiwania drogi, wskazuje wady i zalety obu metod, szacuje złożoność czasową,</p> <ul style="list-style-type: none"> • aktywnie uczestniczy w realizacji projektu zespołowego na wszystkich jego etapach, prezentuje efekty wspólnej pracy. 	
<p>Rozwiązywanie problemów z wykorzystaniem dynamicznych struktur danych</p>				
<ul style="list-style-type: none"> • pisze programy o niewielkim stopniu trudności, • wyjaśnia, co to jest notacja infiksowa, notacja prefiksowa, odwrotna notacja polska, drzewo wyrażenia algebraicznego, • definiuje pojęcie dynamicznej struktury danych, • definiuje dynamiczne struktury danych takie jak: stos, kolejka, lista, vector, • wymienia rodzaje list, 	<ul style="list-style-type: none"> • wyróżnia operacje, które można wykonywać na dynamicznych strukturach danych (stosie, kolejce, liście, typie vector), • omawia zastosowanie dynamicznych struktur danych na różnych przykładach, • zapisuje wyrażenia algebraiczne bez użycia nawiasów, w tym w postaci odwrotnej notacji polskiej, • oblicza wartość wyrażenia arytmetycznego zapisanego w odwrotnej notacji polskiej, • omawia algorytmy znajdowania wyjścia z 	<ul style="list-style-type: none"> • pisze programy o różnym stopniu trudności, szacuje ich efektywność, • dobiera typy danych do rozwiązania problemu, • do przeglądania grafu stosuje algorytm przeszukiwania w głąb (DFS) oraz algorytm przeszukiwania grafu wszcz (BFS), • omawia algorytm Dijkstry, 	<ul style="list-style-type: none"> • charakteryzuje sytuacje algorytmiczne, proponuje sposoby ich rozwiązania, • pisze programy o podwyższonym stopniu trudności: rozwiązuje zadania oznaczone trzema gwiazdkami w podręczniku, • optymalizuje rozwiązania, • stosuje zaawansowane funkcje środowiska i języka programowania, • dobiera struktury danych i metody do rodzaju problemu, • szacuje złożoność algorytmów, 	<ul style="list-style-type: none"> • charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z zastosowaniem złożonych struktur danych, • pisze programy o wysokim stopniu trudności: z olimpiad przedmiotowych, konkursów informatycznych, <p>optymalizuje programy, szacuje ich efektywność</p> <ul style="list-style-type: none"> • wykorzystuje poznane algorytmy do rozwiązywania problemów nieomawianych na lekcjach, np. sprawdzanie spójności grafu,

<ul style="list-style-type: none">definiuje graf, wymienia elementy i rodzaje grafów, wymienia sposoby reprezentacji grafu (macierz sąsiedztwa, lista sąsiedztwa),	<p>labiryntu z wykorzystaniem iteracji i rekurencji,</p> <ul style="list-style-type: none">symuluje problem Flawiusza,sortuje dane leksykograficznie,stosuje typ vector do reprezentacji grafu w postaci list sąsiedztwa,omawia algorytm przeszukiwania grafu w głąb (DFS),omawia algorytm przeszukiwania grafu wszcz (BFS),wyjaśnia, do czego służy algorytm Dijkstry,wyjaśnia różnicę między przekazywaniem parametrów do funkcji przez wartość i przez referencję,		<ul style="list-style-type: none">implementuje algorytmy grafowe – BFS, DFS, algorytm Dijkstry,w reprezentacji liczb rzeczywistych w komputerze stosuje reprezentację stało- lub zmiennoprzecinkową zgodnie ze specyfikacją algorytmu, minimalizując błędy w obliczeniach,	
--	---	--	---	--

KLASA 4

2	3	4	5	6
Relacyjne bazy danych				
<ul style="list-style-type: none"> zna podstawowe pojęcia dotyczące relacyjnych baz danych, wie, co to jest język SQL, zna podstawowe klauzule tego języka, zna zasady tworzenia zapytań do bazy z wykorzystaniem języka SQL, 	<ul style="list-style-type: none"> wyszukuje informacje w bazach danych, tworzy formularze, kwerendy i raporty, wykorzystuje język SQL do tworzenia i usuwania baz danych, dodawania tabel do baz danych, usuwania tabel z baz, dodawania rekordów do tabel, importowania danych do tabel, edycji rekordów, tworzy konta użytkowników i przydziela im uprawnienia do wybranej bazy, używając języka SQL, formułuje zapytania zwracające określone dane, sortuje wyniki zapytań, 	<ul style="list-style-type: none"> projektuje i tworzy proste bazy danych, przy tworzeniu aplikacji internetowej projektuje i tworzy interfejs użytkownika, zapewnia komunikację aplikacji z bazą danych, 	<ul style="list-style-type: none"> projektuje zaawansowane relacyjne bazy danych, zarządza nimi, tworzy tabele pomostowe, formularze, kwerendy i raporty, formułuje zapytania w języku SQL, stosując selekcję, sortowanie, projekcję oraz agregowanie danych, 	<ul style="list-style-type: none"> projektuje rozbudowane relacyjne bazy danych, zarządza nimi, wykorzystując zaawansowane narzędzia oraz klauzule języka SQL,
Algorytmy numeryczne				
<ul style="list-style-type: none"> omawia różnice między stałoprzecinkową a zmiennoprzecinkową reprezentacją liczb rzeczywistych w komputerze, wymienia rodzaje błędów w obliczeniach komputerowych, 	<ul style="list-style-type: none"> wykorzystuje pliki tekstowe do wczytywania danych i zapisywania wyników, omawia algorytm znajdujący rozwinięcie binarne nieskracalnego ułamka właściwego, zapisuje liczby w postaci znormalizowanej, 	<ul style="list-style-type: none"> znajduje reprezentację liczby zapisanej w systemie dziesiętnym jako liczby pojedynczej i liczby podwójnej precyzji, świadomie używa typów <code>float</code> i <code>double</code> w zadaniach, stosuje schemat Hornera do zamiany liczby w systemie 	<ul style="list-style-type: none"> stosuje schemat Hornera do szybkiego podnoszenia do potęgi, implementuje algorytmy numeryczne: znajdowania miejsc zerowych funkcji oraz obliczania pierwiastka kwadratowego metodą bisekcji, obliczania pierwiastka kwadratowego 	<ul style="list-style-type: none"> charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z zastosowaniem złożonych struktur danych, pisze programy o wysokim stopniu trudności: z olimpiad przedmiotowych,

<p>rozróżnia błąd względny i bezwzględny,</p> <ul style="list-style-type: none"> znajduje wartość wielomianu algorytmem naiwnym, wie, na czym polegają podstawowe metody obliczeń przybliżonych, zna proste algorytmy badające własności geometryczne (np. położenie punktu względem prostej), wyjaśnia, co to jest fraktal, wskazuje przykłady struktur fraktalnych występujących w przyrodzie, 	<ul style="list-style-type: none"> definiuje liczby pojedynczej precyzji i liczby podwójnej precyzji, wykonuje działania na liczbach zmiennoprzecinkowych, wskazuje różnice między algorytmem stabilnym a algorytmem niestabilnym, znajduje pierwiastki równania kwadratowego algorytmem stabilnym i algorytmem niestabilnym, implementuje algorytm obliczający wartość wielomianu z zastosowaniem schematu Hornera, stosuje w algorytmach numerycznych metody: bisekcji, Newtona–Raphsona, trapezów, prostokątów, omawia algorytmy badające własności geometryczne – położenie punktu względem prostej, przecinania się odcinków, przynależności punktu do figury, podaje przykłady fraktali (zbiór Cantora, drzewo binarne, dywan Sierpińskiego, płatek Kocha), wyjaśnia sposób tworzenia tych fraktali, 	<p>pozycyjnym o wybranej podstawie na liczbę dziesiętną,</p> <ul style="list-style-type: none"> stosuje metodę Monte Carlo w obliczeniach przybliżonych, w algorytmach badających własności geometryczne wykorzystuje macierz oraz regułę Sarrusa do obliczania wyznacznika macierzy, 	<p>metodą Newtona–Raphsona, obliczania pola obszaru zamkniętego metodą prostokątów i metodą trapezów, znajdowania przybliżenia liczby pi oraz symulacja ruchów Browna metodą Monte Carlo,</p> <ul style="list-style-type: none"> implementuje algorytmy badające własności geometryczne, implementuje w języku JavaScript algorytmy generujące fraktale danego stopnia, stosuje metodę IFS do tworzenia fraktali w arkuszu kalkulacyjnym, stosuje funkcję haszującą oraz algorytm Karpa–Rabina w programach wyszukiwujących wzorzec w tekście, 	<p>konkursów informatycznych,</p> <ul style="list-style-type: none"> optymalizuje programy, szacuje ich efektywność
--	--	---	--	--

Zaawansowane algorytmy i techniki programistyczne

<ul style="list-style-type: none"> • wyjaśnia, na czym polega sortowanie leksykograficzne, • wyszukuje wzorzec w tekście algorytmem naiwnym, • rozumie działanie funkcji haszującej, • wskazuje różnice między kryptografią symetryczną i kryptografią asymetryczną, definiuje pojęcia klucz publiczny i klucz prywatny, • wyjaśnia, do czego służy algorytm RSA, i wyróżnia główne etapy tego algorytmu (generowanie kluczy, szyfrowanie z kluczem publicznym oraz deszyfrowanie z kluczem prywatnym), • definiuje programowanie strukturalne, • definiuje programowanie obiektowe i podstawowe pojęcia z nim związane, 	<ul style="list-style-type: none"> • implementuje algorytm naiwny wyszukiwania wzorca w tekście, • wyjaśnia metodę haszowania, • wyjaśnia, jak generuje się klucze publiczny i prywatny oraz szyfruje i deszyfruje informacje w algorytmie RSA, • wyjaśnia, na czym polegają metoda zstępująca i metoda wstępująca, • w programowaniu obiektowym definiuje własne klasy, korzystając ze specyfikatorów dostę 	<ul style="list-style-type: none"> • omawia algorytm Karpa–Rabina do wyszukiwania wzorca w tekście z zastosowaniem funkcji haszującej, • pisze program generujący klucz prywatny i klucz publiczny w algorytmie RSA, • w programowaniu obiektowym stosuje hierarchię klas, wyjaśnia, na czym polega hermetyzacja danych i jakie jest zastosowanie operatora zasięgu, 	<ul style="list-style-type: none"> • pisze programy szyfrujące i deszyfrujące informacje w algorytmie RSA, • stosuje programowanie obiektowe, definiując własne klasy, obiekty, atrybuty i metody, deklaruje konstruktory w klasach, wyjaśnia, na czym polega polimorfizm i czym są metody wirtualne, 	<ul style="list-style-type: none"> • charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z zastosowaniem złożonych struktur danych, • pisze programy o wysokim stopniu trudności: z olimpiad przedmiotowych, konkursów informatycznych, • optymalizuje programy, szacuje ich efektywność
---	---	---	---	---

Rozwiązywanie różnych problemów z wykorzystaniem komputera

<ul style="list-style-type: none"> • wyróżnia etapy pracy nad aplikacją internetową, rozróżnia technologie back-end i front-end, 	<ul style="list-style-type: none"> • wyjaśnia, na czym polega praca nad aplikacją internetową, instaluje i konfiguruje niezbędne 	<ul style="list-style-type: none"> • tworzy podcasty i publikacje wideo, stosując elementy przyciągające uwagę użytkowników, montuje 	<ul style="list-style-type: none"> • tworzy interesujące podcasty i publikacje wideo, dba o właściwy format plików, stosuje 	<ul style="list-style-type: none"> • tworzy rozwinięte responsywne aplikacje internetowe wymagające dużego nakładu pracy
---	---	---	--	---

<ul style="list-style-type: none"> • definiuje pojęcie robota, omawia funkcje wybranych robotów i ich budowę, • rozróżnia pojęcia webcast, webinarium, screencast i podcast, • definiuje pojęcie grafiki informacyjnej, wymienia przykłady grafiki narracyjnej i wizualizacji danych, • uczestniczy w realizacji projektu zespołowego, wykonując powierzone mu zadania o niewielkim stopniu trudności. 	<p>oprogramowanie, przygotowuje bazę danych na potrzeby projektu,</p> <ul style="list-style-type: none"> • opracowuje treści internetowe, korzystając z narzędzi graficznych i multimedialnych, dbając o identyfikację wizualną, • uczestniczy w realizacji projektu zespołowego, wykonując powierzone mu zadania. • projektuje proste poprawne infografiki zawierające uporządkowane informacje, 	<p>materiały, wykorzystując specjalistyczne oprogramowanie,</p> <ul style="list-style-type: none"> • projektuje infografiki, umiejętnie stosując tekst i obraz, wykazuje się przy tym znajomością doboru barw i funkcji koloru, zwraca uwagę na dostosowanie treści do odbiorców, • uczestniczy w realizacji projektu zespołowego na wszystkich jego etapach, bierze czynny udział w tworzeniu dokumentacji projektowej oraz dyskusji panelowej. 	<p>kompresję, stosuje zasady pracy z kamerą i mikrofonem,</p> <ul style="list-style-type: none"> • tworzy infografiki dostosowane do odbiorców, wykazując się dużymi umiejętnościami korzystania z narzędzi graficznych, • aktywnie uczestniczy w realizacji projektu zespołowego na wszystkich jego etapach, opracowuje prezentacje multimedialne, filmy przedstawiające efekty wspólnej pracy, • w dyskusji panelowej przyjmuje rolę moderatora. 	<p>i znajomości nowoczesnych technologii,</p> <ul style="list-style-type: none"> • tworzy infografiki, korzystając z zaawansowanych narzędzi graficznych, • tworzy podcasty i publikacje wideo wymagające znajomości zaawansowanych narzędzi i dużego nakładu pracy, • w projektach zespołowych przyjmuje rolę lidera, opracowuje złożone problemy, posługując się aplikacjami w stopniu zaawansowanym,
--	--	--	---	--