Zespół Szkół Technicznych i Ogólnokształcących
im. Stefana Żeromskiego w Częstochowie

 **INFORMATYKA** (zakres rozszerzony)

**Wymagania edukacyjne na poszczególne oceny**

**Klasa 1**

**Ocenę dopuszczającą** otrzymuje uczeń, który:

* 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,
* 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 cyfrowej tożsamości,
* wymienia sposoby uwierzytelniania użytkowników e-usług,
* wskazuje miejsca występowania e-zasobów,
* rozróżnia wyszukiwarki od przeglądarek internetowych,
* korzysta w podstawowym zakresie z formatowania tekstów w edytorze tekstowym,
* wymienia etapy pracy nad dobrym wystąpieniem publicznym,
* wymienia programy komputerowe do tworzenia prezentacji,
* wyjaśnia pojęcia: wykluczenie i włączenie cyfrowe,
* podaje przykłady negatywnych zachowań w sieci internet,
* wyjaśnia znaczenie kryptografii dla bezpieczeństwa danych,
* 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,
* zapisuje wynik swojej pracy w różnych formatach graficznych,
* wyjaśnia, jak uruchomić środowisko do grafiki 3D online,
* wprowadza dane różnego typu do arkusza kalkulacyjnego,
* omawia zastosowania korespondencji seryjnej,
* wyjaśnia relacje w bazach danych.

**Ocenę dostateczną** otrzymuje uczeń, który spełnia kryteria oceny dopuszczającej, a ponadto:

* wymienia urządzenia wchodzące w skład sieci komputerowej,
* identyfikuje wersję systemu operacyjnego swojego smartfona (komputera),
* wyjaśnia różnicę pomiędzy 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 na zewnętrznym nośniku,
* uruchamia komputer w trybie awaryjnym,
* sprawdza obciążenie procesora,
* wyjaśnia pojęcia fragmentacji i defragmentacji dysku,
* wyjaśnia różnicę pomiędzy systemami plików FAT32 oraz NTFS,
* definiuje pojęcie systemu operacyjnego,
* wyjaśnia różnicę pomiędzy wirtualną a rozszerzoną rzeczywistością,
* wyjaśnia pojęcia: prawo autorskie, licencja,
* rozróżnia i definiuje pojęcia wolnego i otwartego oprogramowania,
* 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,
* wyjaśnia pojęcie licencji Creative Commons,
* wymienia wiarygodne źródła informacji w sieci internet,
* wyjaśnia, jak sprawdzić właściciela serwisu internetowego,
* omawia pojęcia związane z kryptografią,
* wyjaśnia zasadę Kerckhoffsa,
* korzysta z szablonów w edytorze tekstów,
* poprawnie stosuje style nagłówkowe,
* generuje losowe bloki tekstowe,
* ustawia marginesy w dokumencie,
* 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 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,
* 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,
* skaluje i kadruje obraz, dostosowując go do zadanego rozmiaru,
* wymienia podstawowe narzędzia programu Inkscape,
* tworzy dwuwymiarowe animacje,
* 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.

**Ocenę dobrą** otrzymuje uczeń, który spełnia kryteria oceny dostatecznej, a ponadto:

* opisuje, czym jest model warstwowy systemu komputerowego,
* wymienia i wyjaśnia zadania systemu operacyjnego,
* określa różnicę pomiędzy trybem jądra a trybem użytkownika,
* tworzy modele 3D z prostych brył 3D i ich przekształceń,
* 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,
* podaje cechy różnych rodzajów licencji oprogramowania,
* stosuje symbole i wyrażenia w wyszukiwarkach internetowych,
* dobiera kryteria wyboru elementów zestawu komputerowego w zależności od jego przeznaczenia,
* wymienia i opisuje urządzenia sieciowe,
* 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,
* tworzy i modyfikuje własne szablony oraz style tekstowe,
* dzieli tekst na kolumny,
* pracuje z wielostronicowym dokumentem w widoku konspektu,
* wymienia cechy dobrej prezentacji,
* tworzy ciekawe przejścia między slajdami,
* wymienia zasady ochrony danych osobowych,
* opisuje zastosowania technologii komputerowej w różnych dziedzinach życia,
* opisuje rodzaje ataków sieciowych,
* omawia działanie protokołu SSL,
* 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,
* pracuje na warstwach w programie do grafiki wektorowej,
* 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.

**Ocenę bardzo dobrą** otrzymuje uczeń, który spełnia kryteria oceny dobrej, a ponadto:

* 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 urządzeniami tej samej oraz różnych sieci,
* omawia procesy enkapsulacji i dekapsulacji 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,
* wyjaśnia zasady stosowania prawa autorskiego,
* wykorzystuje narzędzia współpracy zdalnej,
* korzysta z automatycznej numeracji tytułów oraz tworzy spis treści,
* tworzy spisy ilustracji i tabel,
* pracuje z dokumentem wspólnie z innymi osobami, korzystając z narzędzi pracy grupowej,
* 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,
* wyjaśnia, jak zwiększyć swoje bezpieczeństwo w sieci poprzez stosowanie różnych technik,
* omawia kryptoanalizę na wybranym przez siebie przykładzie,
* 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,
* tworzy bryły obrotowe 3D na podstawie ich przekroju,
* tworzy trójwymiarowe animacje,
* wykorzystuje zaawansowane formuły, opracowując dane w arkuszu kalkulacyjnym,
* stosuje funkcje zaokrąglające liczby,
* korzysta z możliwości obliczeń walutowych,
* rozwiązuje problemy, wykorzystując programowanie strukturalne i obiektowe.

**Ocenę celującą** otrzymuje uczeń, który spełnia kryteria oceny bardzo dobrej, a ponadto:

* 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,
* korzysta z różnych narzędzi (w tym mobilnych) podczas prezentacji,
* bierze udział w projektach zespołowych jako odpowiedzialny lider projektu,
* wypełnia wszystkie zadania wynikające z roli powierzonej mu w projekcie,
* 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.

**Klasa 2**

**Ocenę dopuszczającą** otrzymuje uczeń, który:

* definiuje podstawowe pojęcia z algorytmiki i programowania: algorytm, program, warunek, iteracja, rekurencja,
* wymienia sposoby reprezentacji algorytmów,
* 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,
* w pisanych programach korzysta ze strukturalnych typów danych: napisów, struktur, tablic,
* definiuje pojęcie systemów liczbowych,
* wyjaśnia, czym jest tablica kodów ASCII,
* wymienia systemy liczbowe używane w informatyce,
* konwertuje liczby między systemami binarnym i decymalnym,
* dodaje pisemnie liczby binarne,
* wyjaśnia, czym są palindrom i anagram, podaje przykłady,
* podaje definicje liczby pierwszej i liczby złożonej,
* implementuje w języku C++ algorytm zliczający dzielniki danej liczby,
* omawia geometryczną interpretację algorytmu Euklidesa,
* definiuje pojęcia: kryptologia, kryptografia, kryptoanaliza, tekst jawny, klucz, szyfrogram,
* 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,
* 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,
* 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,
* definiuje pojęcia iteracji i rekurencji,
* omawia zasadę złotego podziału,
* opisuje rozszerzony algorytm Euklidesa,
* omawia metody zachłanne na przykładzie problemu kasjera, harmonogramu sali, pakowania plecaka i wyszukiwania drogi,
* porównuje metody zachłanną i dynamiczną,
* uczestniczy w realizacji projektu zespołowego, wykonując powierzone mu zadania o niewielkim stopniu trudności.

**Ocenę dostateczną** otrzymuje uczeń, który spełnia kryteria oceny dopuszczającej, a ponadto:

* przedstawia krótkie algorytmy w postaci listy kroków, opisu słownego, pseudokodu, schematu blokowego,
* dodaje liczby binarne,
* 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++: short int, int, long int, long long int,
* 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,
* tworzy algorytmy sprawdzające, czy napis jest palindromem,
* przedstawia w postaci algorytmu problem wyszukiwania anagramów,
* przy pisaniu programów stosuje własne funkcje różnych typów, w tym funkcję typu void,
* 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,
* pisze program szyfrujący napis szyfrem Cezara,
* omawia algorytm zliczania znaków w tekście oraz wyszukujący maksimum z wykorzystaniem tablic,
* implementuje w języku C++ algorytmy wyszukiwania liniowego i liniowego z wartownikiem, porównuje ich efektywność,
* przedstawia w postaci listy kroków lub schematu blokowego algorytmy sortowania prostego (bąbelkowe, przez wybieranie) oraz szybkiego i przez scalanie, określa operacje dominujące,
* omawia algorytm sita Eratostenesa,
* przedstawia algorytmy znajdowania spójnych podciągów, wyznaczania najdłuższego z nich oraz podciągu o największej sumie elementów,
* omawia algorytm znajdowania idola i lidera w zbiorze,
* implementuje w języku C++ algorytmy 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, wykonując powierzone mu zadania.

 **Ocenę dobrą** otrzymuje uczeń, który spełnia kryteria oceny dostatecznej, a ponadto:

* określa specyfikację algorytmu (dane, wynik),
* pisze programy o różnym stopniu trudności, szacuje ich efektywność,
* przedstawia omawiane algorytmy w postaci opisu słownego, listy kroków, schematu blokowego, pseudokodu,
* 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,
* stosuje różne sposoby przekazywania parametrów do funkcji: przez wartość, referencję lub wskaźnik,
* implementuje w języku C++ algorytmy sprawdzające, czy napis jest palindromem,
* pisze programy sprawdzające, czy dwa napisy są anagramami, wykorzystując funkcję sort z biblioteki STL,
* 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,
* implementuje w języku C++ algorytm zliczania znaków w tekście oraz wyszukujący maksimum z wykorzystaniem tablic,
* stosuje algorytm wyszukiwania binarnego i oszacowuje jego złożoność czasową,
* pisze programy sortujące metodami prostymi z zastosowaniem funkcji typu void,
* stosuje algorytmy sortowania szybkiego i przez scalanie,
* pisze program realizujący algorytm sita Eratostenesa,
* implementuje w języku C++ algorytmy wyszukujące spójne podciągi o różnych cechach,
* stosuje w programach algorytmy wyszukiwania lidera i idola w zbiorze,
* porównuje algorytmy iteracyjne i rekurencyjne (liczbę wykonywanych operacji), szacuje ich złożoność czasową,
* zapisuje w postaci programu rozszerzony algorytm Euklidesa, wyjaśnia jego działanie i zastosowanie,
* 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.

**Ocenę bardzo dobrą** otrzymuje uczeń, który spełnia kryteria oceny dobrej, a ponadto:

* charakteryzuje sytuacje algorytmiczne, proponuje sposoby ich rozwiązania,
* pisze programy o podwyższonym stopniu trudności: oznaczone trzema gwiazdkami w podręczniku,
* optymalizuje rozwiązania,
* stosuje zaawansowane funkcje środowiska i języka programowania (np. z biblioteki STL),
* dobiera struktury danych i metody do rodzaju problemu,
* 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 liczby dziesiętnej 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,
* sprawdza, czy napisy są anagramami, stosując sortowanie lub zliczanie znaków,
* przy testowaniu liczby na pierwszość stosuje funkcję typu logicznego,
* wyszukuje liczby bliźniacze,
* wykorzystuje algorytm Euklidesa do działań na ułamkach, stosując struktury lub pary (typ pair),
* szyfruje dane wczytane z pliku z uwzględnieniem polskich znaków diakrytycznych,
* pisze program odczytujący informację ukrytą za pomocą szyfru Cezara z wykorzystaniem analizy częstości znaków w tekście,
* pisze program wyszukujący 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,
* optymalizuje program realizujący algorytm sita Eratostenesa i szacuje jego złożoność czasową,
* wyszukuje spójne podciągi w plikach tekstowych, stosując optymalne algorytmy (w tym programowanie dynamiczne), wyjaśnia ich działanie,
* pisze programy wyszukujące lidera i idola w zbiorze, optymalizuje je, szacuje złożoność czasową,
* 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 kasjera, harmonogramu wykorzystania sali, pakowania plecaka i wyszukiwania drogi, wskazuje wady i zalety obu metod, szacuje złożoność czasową,
* aktywnie uczestniczy w realizacji projektu zespołowego na wszystkich jego etapach, prezentuje efekty wspólnej pracy.

**Ocenę celującą** otrzymuje uczeń, który spełnia kryteria oceny bardzo dobrej, a ponadto:

* charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z zastosowaniem złożonych struktur danych i biblioteki STL języka C++,
* 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 szyfrów (np. permutacyjny lub Vigenere’a) i różnych kluczy,
* 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 i struktury danych do wyszukiwania spójnych podciągów,
* 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,
* stosuje w programach algorytmy sortowania inne niż omawiane na lekcjach (np. heapsort),
* bierze udział w olimpiadach i konkursach, zajmując punktowane miejsca,
* w projektach zespołowych przyjmuje rolę lidera.

**Klasa 3

Ocenę dopuszczającą** otrzymuje uczeń, który:

* 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,
* wyjaśnia, na czym polega sortowanie leksykograficzne,
* definiuje graf, wymienia elementy i rodzaje grafów, wymienia sposoby reprezentacji grafu (macierz sąsiedztwa, lista sąsiedztwa),
* omawia różnice między stałoprzecinkową a zmiennoprzecinkową reprezentacją liczb rzeczywistych w komputerze,
* wymienia rodzaje błędów w obliczeniach komputerowych, rozróżnia błąd względny i bezwzględny,
* 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,
* 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,
* 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,
* wyróżnia etapy pracy nad aplikacją internetową, rozróżnia technologie back-end i front-end,
* 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.

**Ocenę dostateczną** otrzymuje uczeń, który spełnia kryteria oceny dopuszczającej, a ponadto:

* 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 labiryntu z wykorzystaniem iteracji i rekurencji,
* 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 wszerz (BFS),
* wyjaśnia, do czego służy algorytm Dijkstry,
* wyjaśnia różnicę między przekazywaniem parametrów do funkcji przez wartość i przez referencję,
* wykorzystuje pliki tekstowe do wczytywania danych i zapisywania wyników,
* omawia algorytm znajdujący rozwinięcie binarne nieskracalnego ułamka właściwego,
* zapisuje liczby w postaci znormalizowanej,
* 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,
* 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 dostępu,
* 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,
* wyjaśnia, na czym polega praca nad aplikacją internetową, instaluje i konfiguruje niezbędne oprogramowanie, przygotowuje bazę danych na potrzeby projektu,
* tworzy proste programy sterujące robotem, korzysta przy tym ze środowiska dostępnego dla fizycznego modelu robota lub z symulatora,
* opracowuje treści internetowe, korzystając z narzędzi graficznych i multimedialnych, dbając o identyfikację wizualną,
* projektuje proste poprawne infografiki zawierające uporządkowane informacje,
* uczestniczy w realizacji projektu zespołowego, wykonując powierzone mu zadania.

**Ocenę dobrą** otrzymuje uczeń, który spełnia kryteria ocen niższych, a ponadto:

* 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 wszerz (BFS),
* omawia algorytm Dijkstry,
* znajduje reprezentację liczby zapisanej w systemie dziesiętnym jako liczby pojedynczej i liczby podwójnej precyzji,
* świadomie używa typów **float** i **double** w zadaniach,
* stosuje schemat Hornera do zamiany liczby w systemie pozycyjnym o wybranej podstawie na liczbę dziesiętną,
* stosuje metodę Monte Carlo w obliczeniach przybliżonych,
* w algorytmach badających własności geometryczne wykorzystuje macierz oraz regułę Sarrusa do obliczania wyznacznika macierzy,
* 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,
* projektuje i tworzy proste bazy danych,
* przy tworzeniu aplikacji internetowej projektuje i tworzy interfejs użytkownika, zapewnia komunikację aplikacji z bazą danych,
* programuje roboty według określonych wytycznych, np. zatrzymanie przed przeszkodą,
* tworzy podcasty i publikacje wideo, stosując elementy przyciągające uwagę użytkowników, montuje materiały, wykorzystując specjalistyczne oprogramowanie,
* 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.

**Ocenę bardzo dobrą** otrzymuje uczeń, który spełnia kryteria ocen niższych, a ponadto:

* 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,
* 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,
* stosuje schemat Hornera do szybkiego podnoszenia do potęgi,
* implementuje algorytmy numeryczne: znajdowania miejsc zerowych funkcji oraz obliczania pierwiastka kwadratowego metodą bisekcji, obliczania pierwiastka kwadratowego metodą Newtona−Raphsona, obliczania pola obszaru zamkniętego metodą prostokątów i metodą trapezów, znajdowania przybliżenia liczby pi oraz symulacja ruchów Browna metodą Monte Carlo,
* 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 wyszukujących wzorzec w tekście,
* 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,
* 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,
* tworzy aplikacje internetowe z przejrzystym interfejsem użytkownika korzystające z sieciowej bazy danych, testuje je i wprowadza poprawki,
* testuje i optymalizuje programy sterujące robotem,
* tworzy interesujące podcasty i publikacje wideo, dba o właściwy format plików, stosuje kompresję, stosuje zasady pracy z kamerą i mikrofonem,
* 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.

**Ocenę celującą** otrzymuje uczeń, który spełnia kryteria ocen niższych, a ponadto:

* charakteryzuje skomplikowane sytuacje algorytmiczne, proponuje optymalne rozwiązanie sytuacji problemowej z zastosowaniem złożonych struktur danych,
* bierze udział w konkursach i olimpiadach informatycznych i zajmuje punktowane miejsca,
* pisze programy o wysokim stopniu trudności: z olimpiad przedmiotowych, konkursów informatycznych,
* optymalizuje programy, szacuje ich efektywność,
* wykorzystuje poznane algorytmy do rozwiązywania problemów nieomawianych na lekcjach, np. sprawdzanie spójności grafu,
* projektuje rozbudowane relacyjne bazy danych, zarządza nimi, wykorzystując zaawansowane narzędzia oraz klauzule języka SQL,
* tworzy rozwinięte responsywne aplikacje internetowe wymagające dużego nakładu pracy i znajomości nowoczesnych technologii,
* tworzy infografiki, korzystając z zaawansowanych narzędzi graficznych,
* programuje roboty wg własnych projektów, wykazując się przy tym kreatywnością, oraz wykorzystuje aplikacje mobilne do sterowania nimi,
* 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,
* w dyskusjach panelowych przyjmuje funkcję eksperta.