

Wymagania edukacyjne z informatyki w zakresie rozszerzonym dla klasy pierwszej (dla uczniów z klas z informatyką na poziomie rozszerzonym obowiązują także wymagania na poziomie podstawowym)

**Dopuszczający:** Wie, co to jest algorytm. Określa dane do zadania oraz wyniki. Zna podstawowe zasady graficznego prezentowania algorytmów: podstawowe rodzaje bloków, ich przeznaczenie i sposoby umieszczania w schemacie blokowym. Potrafi narysować (odręcznie) schemat blokowy algorytmu liniowego. Określa sytuacje warunkowe. Podaje przykłady zadań, w których występują sytuacje warunkowe. Wie, na czym polega powtarzanie tych samych operacji. Potrafi omówić, na przykładzie, algorytm znajdowania najmniejszego z trzech elementów. Potrafi zanalizować przebieg algorytmu dla przykładowych danych i ocenić w ten sposób jego poprawność. Zna klasyfikację języków programowania. Zna ogólną budowę programu i najważniejsze elementy języka – słowa kluczowe, instrukcje, wyrażenia, zasady składni. Potrafi zrealizować prosty algorytm liniowy w języku wysokiego poziomu; potrafi skompilować i uruchomić gotowy program. Wymienia przykłady prostych struktur danych. Potrafi zadeklarować zmienne typu liczbowego (całkowite, rzeczywiste) i stosować je w zadaniach. Określa następujące pojęcia: bit, bajt. Zna pojęcie systemu pozycyjnego. Wymienia części składowe zestawu komputerowego, podaje ich parametry i przeznaczenie. Rozróżnia rodzaje pamięci komputera, określa ich własności i przeznaczenie. Wie, co to jest system operacyjny, i korzysta z jego podstawowych funkcji. Wykonuje podstawowe operacje na plikach i folderach. Zna pojęcie sieci komputerowej, potrafi wymienić jej rodzaje. Zna pojęcie logowania. Potrafi wymienić kilka cech pracy w sieci, odróżniających ją od pracy na autonomicznym komputerze. Zna kilka sposobów połączenia z Internetem. Potrafi omówić historię komputerów. Umie wskazać ogólny kierunek zmian w technologiach komputerowych. Zna podstawowe zasady netykiety. Wie, czym zajmuje się informatyka i jakie są jej metody, umie uruchomić komputer i programy komputerowe, umie komunikować się z komputerem za pomocą systemu operacyjnego w elementarnym zakresie, umie posługiwać się podstawowym oprogramowaniem użytkowym w elementarnym zakresie, umie stosować posiadane wiadomości do wykonywania elementarnych czynności w bardzo prostych sytuacjach, w wypowiedziach popełnia liczne błędy merytoryczne; zna podstawowe pojęcia dotyczące sieci komputerowej, wie, jakie urządzenia realizują przepływ informacji przez sieć, potrafi samodzielnie konfigurować komputery do pracy w sieciach, zna zastosowanie narzędzi administracyjnych systemu Windows

**Dostateczny:** Wymienia przykłady czynności i działań w życiu codziennym oraz zadań szkolnych, które uważa się za algorytmy. Zna pojęcie specyfikacji zadania. Zna wybrane sposoby prezentacji algorytmów. Przedstawia algorytm w postaci listy kroków. Tworzy schemat blokowy algorytmu z warunkiem prostym i pętlą. Podczas rysowania schematów blokowych potrafi wykorzystać Autokształty z edytora tekstu. Korzysta (w stopniu podstawowym) z programu edukacyjnego do symulacji działania

algorytmu skonstruowanego w postaci schematu blokowego. Potrafi odróżnić algorytm liniowy od algorytmu z warunkami (z rozgałęzieniami). Zna pojęcie iteracji i rozumie pojęcie algorytmu iteracyjnego. Podaje ich przykłady. Wie, od czego zależy liczba powtórzeń. Potrafi omówić algorytm porządkowania elementów (metodą przez wybór) na praktycznym przykładzie, np. wybierając najwyższego ucznia z grupy. Omawia i analizuje wybrane techniki sortowania w postaci gotowych schematów blokowych, skonstruowanych w programie edukacyjnym. Potrafi ocenić poprawność działania algorytmu i jego zgodność ze specyfikacją. Określa liczbę prostych działań zawartych w algorytmie. Zapisuje program w czytelnej postaci – stosuje wcięcia, komentarze. Rozumie pojęcia: implementacja, kompilacja, uruchomienie, testowanie. Rozumie znaczenie i działanie podstawowych instrukcji wybranego języka programowania wysokiego poziomu. Rozróżnia i poprawia błędy kompilacji i błędy wykonania. Potrafi zrealizować algorytmy iteracyjne w języku wysokiego poziomu. Zna podstawowe zasady poprawnego programowania; testuje tworzone programy; wie, jak unikać problemów, takich jak np. zapętlenie się programu. Wie, czym jest zmienna w programie i co oznacza przypisanie jej konkretnej wartości. Rozróżnia struktury danych: proste i złożone. Podaje przykłady. Deklaruje typy złożone. Wie, co to jest system binarny, i potrafi dokonać zamiany liczby z systemu dziesiętnego na binarny i odwrotnie. Potrafi sklasyfikować środki (urządzenia) i narzędzia (oprogramowanie) technologii informacyjnej. Wie, jak działa komputer. Wyjaśnia rolę procesora. Rozumie organizację pamięci komputerowych. Potrafi omówić funkcje systemu operacyjnego. Zna zasady ochrony plików. Potrafi nadać podstawowe atrybuty plikom, jak też wyszukać poszczególne pliki. Wymienia korzyści płynące z korzystania z sieci. Zna podstawowe klasy i topologie sieciowe. Potrafi wymienić urządzenia i elementy sieciowe oraz omówić ich ogólne przeznaczenie. Zna cechy systemu działającego w szkolnej pracowni. Orientuje się – w zakresie podstawowym – w działaniu Internetu. Potrafi określić nowoczesne trendy w zastosowaniu urządzeń komputerowych. Jest w stanie omówić prawne i społeczne aspekty zastosowania informatyki. Spełnia wszystkie wymagania na ocenę niższą. Wymienia algorytmy klasyczne zawarte w podstawie programowej.

**Dobry:** Określa zależności między problemem, algorytmem a programem komputerowym. Potrafi odpowiedzieć na pytanie, czy istnieją działania, które nie mają cech algorytmów. Przedstawia dokładną specyfikację dowolnego zadania. Zna znaczenie i działanie instrukcji symbolicznego języka programowania (pseudojęzyka). Potrafi zapisać algorytm z warunkami zagnieżdżonymi i pętlą w wybranej postaci. Potrafi skonstruować algorytm z warunkami zagnieżdżonymi i pętlą za pomocą programu edukacyjnego. Analizuje algorytmy, w których występują powtórzenia (iteracje). Zna sposoby zakończenia iteracji. Określa kroki iteracji. Potrafi zapisać w wybranej notacji np. algorytm sumowania  $n$  liczb, algorytm obliczania silni, znajdowania minimum w ciągu  $n$  liczb, algorytm rozwiązywania równania liniowego. Rozumie, co to jest złożoność algorytmu i potrafi określić liczbę operacji wykonywanych na elementach zbioru w wybranym algorytmie sortowania. Potrafi prezentować złożone algorytmy (z podprogramami) w wybranym języku programowania. Zna rekurencyjne realizacje prostych algorytmów. Rozumie i stosuje zasady programowania strukturalnego. Wie, na czym polega różnica pomiędzy

przekazywaniem parametrów przez zmienną i przez wartość w procedurach. Wie, jakie znaczenie ma zasięg działania zmiennej. Rozumie zasady postępowania przy rozwiązywaniu problemu metodą zstępującą. Zna zasady działania wybranych algorytmów sortowania. Zna podstawowe procedury graficzne, potrafi narysować na ekranie wykres funkcji i podstawowe figury geometryczne. Potrafi zastosować łańcuchowy i tablicowy typ danych w zadaniach. Potrafi wykonać działania arytmetyczne na liczbach binarnych (dodawanie i odejmowanie). Zna system szesnastkowy i potrafi wykonać konwersję liczb binarnych na liczby w systemie szesnastkowym i odwrotnie. Analizuje model logiczny komputera. Wie, co to jest kod ASCII. Potrafi wymienić rodzaje aktualnie używanych komputerów. Zna metody wyszukiwania plików. Zna znaczenie protokołu w sieciach (w tym TCP/IP). Wie, na czym polega wymiana informacji w sieci. Zna zasady pracy w sieci, m.in. zasady udostępniania zasobów. Potrafi omówić zagrożenia płynące z sieci. Charakteryzuje różne połączenia z Internetem; potrafi omówić przesyłanie pakietów danych w Internecie. Potrafi wskazać nowości w zakresie usług internetowych oraz odszukać informacje na temat najnowszych pomysłów na komputery. Spełnia wszystkie wymagania na oceny niższe.

**Bardzo dobry:** Zapisuje dowolny algorytm w wybranej przez siebie postaci (notacji), m.in. w pseudojęzyku. Zapisuje algorytmy z pętlą zagnieżdżoną. Potrafi przeprowadzić szczegółową analizę poprawności konstrukcji schematu blokowego. Analizuje działanie algorytmu dla przykładowych danych. Stosuje swobodnie oprogramowanie edukacyjne do graficznej prezentacji i analizy algorytmów. Zna metodę „dziel i zwyciężaj”, algorytm generowania liczb Fibonacciego, schemat Hornera. Omawia ich iteracyjną realizację i potrafi przedstawić jeden z nich w wybranej notacji. Zna inne algorytmy sortowania, np. kubełkowe, przez wstawianie. Zna przynajmniej jeden algorytm numeryczny, np. obliczanie wartości pierwiastka kwadratowego. Wskazuje różnicę między rekurencją a iteracją. Zna rekurencyjną realizację wybranych algorytmów, np. silnię i algorytm Euklidesa. Potrafi zamienić algorytm zapisany iteracyjnie na postać rekurencyjną. Potrafi porównać złożoność różnych algorytmów tego samego zadania dla tych samych danych. Wie, kiedy algorytm jest uniwersalny. Wie, jaka jest różnica między językiem wysokiego poziomu a językiem wewnętrznym; potrafi określić rolę procesora i pamięci operacyjnej w działaniu programów. Potrafi realizować nawet bardzo złożone algorytmy, stosować procedury graficzne w realizacji skomplikowanych zadań – np. tworzyć własne animacje. Potrafi prezentować algorytmy rekurencyjne w postaci programu; potrafi zamienić rozwiązanie iteracyjne algorytmu na rekurencyjne. Zapisuje w postaci programu wybrane algorytmy sortowania. Opracowuje złożony program – umie podzielić zadania, ustalić sposoby przekazywania danych pomiędzy procedurami. Zabezpiecza tworzone programy przed wprowadzeniem przez użytkownika błędnych danych. Rozumie, na czym polega dobór struktur danych do algorytmu. Potrafi zastosować rekordowy typ danych. Potrafi omówić dokładnie działanie procesora. Potrafi wykonać dowolną konwersję pomiędzy systemem dziesiętnym, dwójkowym i szesnastkowym. Zna sposób zapisu liczby całkowitej i rzeczywistej (zmiennoprzecinkowej). Umie wymienić przynajmniej dwa systemy operacyjne i podać ich najważniejsze funkcje. Zna zaawansowane metody

wyszukiwania i odzyskiwania plików. Zna przynajmniej jeden algorytm szyfrowania danych. Potrafi zaszyfrować i odszyfrować prosty tekst. Zna schemat działania sieci komputerowych. Potrafi wymienić zalety i wady różnych topologii sieci. Charakteryzuje topologie gwiazdy, magistrali i pierścienia. Zna podstawowe cechy systemu Linux. Umie z pomocą nauczyciela zrealizować małą sieć komputerową – skonfigurować jej składniki, udostępnić pliki, dyski, drukarki, dodać użytkowników. Przygotowuje analizę porównawczą, pokazującą na przestrzeni wielu lat rozwój informatyki, w tym sieci komputerowych, oraz multimediiów. Spełnia wszystkie wymagania na oceny niższe. Potrafi zapisać algorytmy klasyczne zawarte w podstawie programowej w postaci schematu blokowego i języka programowania. Wie, jak tworzyć formularz z podformularzami, potrafi tworzyć makropolecenia, potrafi pisać procedury obsługi zdarzeń, potrafi budować środowisko zarządzania bazą danych, potrafi zabezpieczać bazę danych, potrafi stosować kwerendę składającą i definiującą dane, zna struktury listowe, wie, jak implementowane są drzewa binarne i potrafi je przeszukiwać, zna wybrane algorytmy dotyczące tekstów, potrafi stosować wybrane techniki rozszerzające funkcjonalność języka HTML, potrafi przetwarzać pliki tekstowe umieszczone w zasobach serwera www, potrafi budować proste aplikacje multimedialne. Celujący: Przestrzega zasad zapisu algorytmów w zadanej postaci (notacji). Potrafi trafnie dobrać do algorytmu sposób prezentacji. Stosuje poznane metody prezentacji algorytmów w opisie zadań (problemów) z innych przedmiotów szkolnych oraz różnych dziedzin życia. Potrafi samodzielnie zapoznać się z nowym programem edukacyjnym przeznaczonym do konstrukcji schematów blokowych. Potrafi zaproponować własny pseudojęzyk (postać instrukcji i zasady składni). Rozumie dokładnie technikę rekurencji (znaczenie stosu). Potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję. Zna trudniejsze algorytmy, np. algorytm trwałego małżeństwa, problem ośmiu hetmanów. Zna inne techniki sortowania, np. sortowanie przez scalanie ciągów i metodę szybką. Potrafi zapisać je w różnych notacjach (również w języku programowania wysokiego poziomu). Zna inne algorytmy numeryczne, np. wyznaczanie miejsca zerowego funkcji. Korzysta samodzielnie z dodatkowej literatury. Ocenia złożoność czasową i pamięciową algorytmu. Zna odpowiednie wzory. Ocenia efektywność działania programu. Wie, na czym polega programowanie obiektowe i zdarzeniowe. Potrafi stosować techniki programowania dynamicznego lub programowania obiektowego. Zna i rozumie podobieństwa i różnice w strukturze programu zapisanego w różnych językach programowania – w deklaracji zmiennych i procedur, w składni i zasadach działania poszczególnych procedur. Sprawnie korzysta z dodatkowej, fachowej literatury. Zna dynamiczne struktury danych. Potrafi zastosować zmienne typu wskaźnikowego w zadaniach. Zna struktury listowe, np. stos, kolejkę, listę. Rozumie i potrafi zastosować typ obiektowy. Zna operacje logiczne na liczbach binarnych i przesunięcia bitowe. Potrafi zapisać w języku programowania wysokiego poziomu algorytm konwersji liczb z dowolnego systemu pozycyjnego na inny. Wykonuje sprawnie operacje na liczbach zapisanych w różnych systemach pozycyjnych. Potrafi odzyskać utracony plik, stosując zaawansowane metody. Potrafi omówić różne systemy operacyjne, wskazać ich najważniejsze funkcje. Samodzielnie wyszukuje informacje na temat kompresji i szyfrowania danych. Zna kilka sposobów szyfrowania informacji. Potrafi

zapisać algorytm szyfrowania w postaci programu. Zna działanie algorytmu kompresji. Omawia szczegółowo model warstwowy sieci. Omawia różne systemy sieciowe. Dokonuje ich analizy porównawczej. Charakteryzuje system Linux. Potrafi samodzielnie zbudować małą sieć domową. Wskazuje tendencje w rozwoju informatyki i jej zastosowania, dostrzegając przeobrażenia w tej dziedzinie w kraju i na świecie. Spełnia wszystkie wymagania na oceny niższe. Stosuje algorytmy klasyczne zawarte w podstawie programowej do rozwiązywania zadań maturalnych.

## Wymagania edukacyjne z informatyki w zakresie podstawowym dla klasy pierwszej

### Ocena dopuszczająca:

#### Dział 1. Usługi internetowe.

Uczeń:

- umie wymienić inne narzędzia do wyszukiwania informacji w Internecie niż wyszukiwarki
- wymienia adresy ogólnoświatowych wyszukiwarek
- podaje adresy stron o charakterze encyklopedycznym lub słownikowym
- podaje budowę adresu e-mail
- określa zasady netykiety
- rozwija skrót FTP, podaje przykład klienta FTP, podaje przykłady e-learningu

#### Dział 2. Budowa komputera

Uczeń:

- wymienia podzespoły komputerowe
- wymienia urządzenia peryferyjne
- wylicza rodzaje pamięci masowych
- wymienia podzespoły niezbędne do złożenia komputera sieciowego

#### Dział 3. Grafika i multimedia

Uczeń:

- wylicza rodzaje kompresji
- wymienia formaty plików graficznych
- podaje przykłady edytorów grafiki rastrowej
- podaje przykłady edytorów grafiki wektorowej

– wymienia formaty plików wideo

- podaje przykłady programów do obróbki wideo

#### **Dział 4. Zaawansowana edycja tekstu**

Uczeń:

- określa zalety stosowania list wielopoziomowych
- wymienia właściwości stylu
- wylicza rodzaje odwołań i spisów
- definiuje pojęcie szablonu
- podaje typowe przykłady dokumentów tworzonych na podstawie szablonu
- definiuje korespondencję seryjną
- wymienia kolejne kroki tworzenia korespondencji seryjnej
- określa rodzaje dokumentów tworzonych tą metodą
- wymienia przykłady stron z edytorami on-line
- wylicza dostępne rodzaje usług (edytorów)

#### **Dział 5. Arkusz kalkulacyjny**

Uczeń:

- wymienia sposoby importowania danych
- wylicza typy reguł formatowania warunkowego dostępne w MS Excel
- wymienia typy wykresów dostępne w MS Excel
- wymienia przykłady stron z arkuszami on-line
- określa rodzaje operacji bazodanowych dostępnych w Excelu

#### **Dział 6. Relacyjne bazy danych**

Uczeń:

- wylicza obiekty baz danych
- wymienia typy danych
- wymienia rodzaje relacji
- określa rodzaje kluczy w relacjach
- wymienia rodzaje kwerend

- wylicza operacje logiczne
- określa funkcje agregujące
- wymienia typy kwerendfunkcjonalnych

## **Dział 7. Prezentowanie danych**

Uczeń:

- określa zasady tworzenie prezentacji multimedialnych
- wylicza nazwy programów wspomagających tworzenie prezentacji
- wymienia rozszerzenia wyeksportowanych plików
- wymienia sposoby łączenia (osadzania) stylów CSS
- podaje przykład selektora i jego cechy
- wymienia cechy JavaScript
- podaje przykłady zdarzeń

## **Dział 8. Algorytmy i programowanie**

Uczeń:

- wymienia elementy schematublokowego
- wylicza operatory(przypisania, porównań)
- wymienia rodzaje pętli
- podaje sposoby zatrzymywania działania pętli
- uruchamia aplikację
- tworzy najprostszy skrypt
- uruchamia skrypt
- wymienia cechy językaPascal
- uruchamia edytor Pascala
- tworzy najprostszy program
- kompiluje programy
- uruchamia programy



## **Dział 9. Projekty międzyprzedmiotowe (tematy alternatywne)**

Uczeń:

- zaangażował się w prace tylko w niewielkim zakresie
- posiada wiedzę i umiejętności wyłącznie na tematy przez siebie opracowane
- nie uczestniczył w prezentacji efektów pracy

### **Ocena dostateczna:**

#### **Dział 1. Usługi internetowe.**

Uczeń:

- zna budowę wyszukiwarki
- rozróżnia pojęcia: wyszukiwarka i przeglądarka
- wyjaśnia pojęcie słowa kluczowego
- odszukuje plany miast wraz z widokiem satelitarnym lub widokiem ulicy
- charakteryzuje zasady korzystania z internetowych baz danych
- tworzy kontakty i grupy kontaktów
- wyjaśnia pojęcie: ukryta kopia
- opisuje różnice pomiędzy „odpowiedz” a „odpowiedz wszystkim”
- wysyła i odbiera wiadomości e-mail
- wyjaśnia działanie usługi FTP
- charakteryzuje sposoby łączenia się z udziałem sieciowym (lokalnym)
- omawia wady i zalety e-learningu

#### **Dział 2. Budowa komputera**

Uczeń:

- zna pojęcia: BIOS POST
- umie omówić start komputera
- rozróżnia podzespoły komputerowe
- rozpoznaje urządzenia peryferyjne i pamięci masowe
- wyjaśnia potrzebę zastosowania podzespołów w projekcie komputera sieciowego

### **Dział 3. Grafika i multimedia**

Uczeń:

- wyjaśnia pojęcia: grafika rastrowa, bitmapa, rozdzielczość, głębia kolorów
- charakteryzuje kompresję stratną i bezstratną
- opisuje cechy formatów graficznych
- omawia wady i zalety grafiki wektorowej
- wyjaśnia sposób tworzenia grafiki wektorowej
- opisuje cechy formatów wideo
- wyjaśnia pojęcie rozdzielczości
- wyjaśnia związek pomiędzy jakością filmu a jego rozmiarem

### **Dział 4. Zaawansowana edycja tekstu**

Uczeń:

- rozpoznaje listy: numerowaną, punktowaną i wielopoziomową
- uzasadnia stosowanie stylów w dokumentach
- uzasadnia używanie odwołań i spisów
- omawia wady i zalety korzystania z szablonów
- tworzy dokument na podstawie gotowych szablonów (koperta, etykiety, lista adresatów MS Word)
- scala i drukuje dokumenty
- omawia różnice i podobieństwa pomiędzy wersją desktopową a on-line'ową pakietu biurowego

### **Dział 5. Arkusz kalkulacyjny**

Uczeń:

- omawia budowę pliku CSV
- wyjaśnia różnice pomiędzy poszczególnymi typami reguł formatowania warunkowego
- wyjaśnia różnicę pomiędzy poszczególnymi typami wykresów
- rozróżnia elementy wykresu (np. seria danych, oś, obszar kreślenia, znacznik itp.)
- omawia różnice i podobieństwa pomiędzy wersją desktopową a on-line'ową arkusza kalkulacyjnego
- wyjaśnia pojęcia: sortowanie, filtrowanie, tabela przestawna

## **Dział 6. Relacyjne bazy danych**

Uczeń:

- charakteryzuje: tabelę, rekord, pole
- porównuje typy danych
- charakteryzuje relacje: jeden-do-jednego, jeden-do-wielu, wiele-do-wielu
- wyjaśnia różnicę pomiędzy kluczem podstawowym a obcym
- charakteryzuje kwerendę szczegółową (wybierającą) i podsumowującą
- wyjaśnia różnicę pomiędzy sumą a iloczynem logicznym warunków
- charakteryzuje kwerendę tworzącą tabele, dołączającą, usuwającą, aktualizującą

## **Dział 7. Prezentowanie danych**

Uczeń:

- wyjaśnia pojęcia: slajd, konspekt, przejście slajdu, chronometraż
- rozpoznaje rozszerzenia plików i kojarzy je z odpowiednią aplikacją
- wyjaśnia różnicę pomiędzy .ppsx a .pptx
- charakteryzuje formaty zapisu plików w różnych wersjach PowerPointa
- wyjaśnia pojęcia: CSS, selektor, cecha, wartość
- charakteryzuje zalety użycia CSS
- wyjaśnia pojęcie: interpreter
- rozróżnia zdarzenia (np. onClick, onMouseOver itp.)

## **Dział 8. Algorytmy i programowanie**

Uczeń:

- rozpoznaje elementy schematu blokowego
- charakteryzuje elementy schematu blokowego
- charakteryzuje poszczególne rodzaje pętli
- wyjaśnia pojęcia: inkrementacja, dekrementacja, inicjalizacja zmiennej
- tworzy skrypty zawierające instrukcje czytaj/pisz
- inicjalizuje zmienne

- stosuje instrukcję przypisania
- używa operatorów arytmetycznych
- charakteryzuje typy danych Pascala
- tworzy programy zawierające instrukcje czytaj/pisz
- inicjalizuje zmienne
- stosuje instrukcję przypisania
- używa operatorów arytmetycznych

### **Dział 9. Projekty międzyprzedmiotowe (tematy alternatywne)**

Uczeń:

- zaangażował się w prace w stopniu średnim
- posiada pobieżną wiedzę ze wszystkich działów tematu
- uczestniczył w prezentacji efektów pracy

**Ocena **dobra**:**

### **Dział 1. Usługi internetowe.**

Uczeń:

- wyszukuje informacje tekstowe i multimedialne na podstawie podanego słowa kluczowego
- tłumaczy pojedyncze słowa w języku obcym
- planuje trasę przejazdu z wykorzystaniem map elektronicznych
- posługuje się różnymi narzędziami wyszukując informacje w Internecie
- porównuje ceny towarów na aukcjach i w sklepach
- łączy się z serwerem FTP
- pobiera i wysyła dane FTP
- pobiera i wysyła dane na wirtualne dyski
- pobiera dane z udziałów sieciowych (lokalnych)
- wyszukuje lekcje e-learningu na dany temat
- aktywnie korzysta ze-lekcji

## **Dział 2. Budowa komputera**

Uczeń:

- określa zadania poszczególnych podzespołów komputerowych
- porównuje zastosowanie urządzeń peryferyjnych i pamięci masowych
- dobiera podzespoły komputera sieciowego z uwzględnieniem ich kompatybilności
- dobiera podzespoły komputera sieciowego z uwzględnieniem potrzeb i kosztów
- umie omówić schemat blokowy działania komputera
- zna rodzaje pamięci w komputerze
- umie przeprowadzić reinstalację systemu operacyjnego

## **Dział 3. Grafika i multimedia**

Uczeń:

- wykonuje podstawowe operacje na grafice rastrowej (kadrowanie, zmiana rozmiaru, obracanie)
- przekształca obrazy grafiki rastrowej, wykorzystując efekty
- tworzy albumy internetowe
- tworzy proste obiekty (linie, figury)
- formatuje obiekty graficzne (np. rozmiar, styl i kolor obramowania, styl i kolor wypełnienia)
- ustala zależności między obiektami (np. położenie, wyrównanie)
- przeprowadza montaż filmu
- stosuje efekty i przejścia
- umieszcza w filmie napisy
- dodaje do filmu ścieżkę dźwiękową

## **Dział 4. Zaawansowana edycja tekstu**

Uczeń:

- tworzy i edytuje listę wielopoziomową (korzystając z gotowego szablonu)
- stosuje w dokumencie wbudowane style
- stosuje podpisy i spisy (treści, obiektów itp.)
- tworzy dokument na podstawie gotowego szablonu

- łączy dokument z zewnętrzną bazą danych (np. Excel, Access)
- wstawia i formatuje pola korespondencji seryjnej
- tworzy i udostępnia dokument tekstowy
- współdziela podczas edycji dokumentu on-line

## **Dział 5. Arkusz kalkulacyjny**

Uczeń:

- importuje dane z plików CSV
- tworzy formatowanie warunkowe z jedną regułą
- tworzy wykresy kolumnowe, liniowe, punktowe i kołowe
- dostosowuje wygląd (style linii, kolory wypełnienia, style czcionek) do własnych potrzeb
- tworzy on-line i udostępnia arkusz kalkulacyjny
- współdziela podczas edycji dokumentu on-line
- udostępnia arkusz w sieci lokalnej
- współdziela podczas edycji arkusza w sieci lokalnej
- sortuje dane tekstowe i liczbowe
- filtruje dane według jednego warunku
- tworzy tabelę przestawną

## **Dział 6. Relacyjne bazy danych**

Uczeń:

- tworzy tabelę w bazie danych
- wypełnia tabelę danymi
- importuje dane do tabeli
- tworzy bazę danych zawierającą powiązania jeden-do-jednego oraz jeden-do-wielu
- tworzy kwerendę wybierającą
- sortuje i filtruje dane w kwerendzie
- stosuje alternatywę i koniunkcję warunków
- projektuje kwerendę tworzącą tabele

- projektuje kwerendę dołączającą

## **Dział 7. Prezentowanie danych**

Uczeń:

- tworzy prezentację według konspektu
- umieszcza w prezentacji grafikę i tekst
- zarządza przejściami slajdów i chronometrażem
- zapisuje prezentacje jako .ppsx, .ppt, .pdf
- konwertuje prezentacje pomiędzy .odp a .pptx (i odwrotnie)
- łączy plik .html z .css
- ustala podstawowe parametry czcionek, akapitów, grafik
- wstawia skrypt do dokumentu .html
- stosuje `document.write` do wyświetlenia tekstu i kodu HTML
- stosuje operatory arytmetyczne
- korzysta z funkcji
- obsługuje okna dialogowe

## **Dział 8. Algorytmy i programowanie**

Uczeń:

- formułuje specyfikację algorytmu
- określa wynik działania algorytmu
- formułuje specyfikację algorytmu
- określa wynik działania rozgałęzionego algorytmu
- tworzy skrypty zawierające instrukcję warunkową
- używa operatorów logicznych
- tworzy programy zawierające instrukcję warunkową
- efektywnie korzysta z informacji kompilatora (poprawia błędy w kodzie)

Dział 9. Projekty międzyprzedmiotowe (tematy alternatywne)

Uczeń:

- zaangażował się w prace w stopniu wysokim
- posiada ogólną wiedzę ze wszystkich działów tematu
- aktywnie uczestniczył w prezentacji efektów pracy

Ocena **bardzo dobra**:

### **Dział 1. Usługi internetowe.**

Uczeń:

- dobiera słowa kluczowe
- tłumaczy wyrażenia i idiomy na wiele języków
- korzysta z nowych usług implementowanych w wyszukiwarce (np. kalkulator, wyszukiwanie graficzne)
- planuje trasę podróży z uwzględnieniem alternatywnych dróg, transportu publicznego oraz obiektów godnych zwiedzenia
- po wieloaspektowej analizie (parametry, cena, koszty transportu, zaufanie do sprzedawcy) dokonuje zakupów on-line
- udostępnia dane z wirtualnych dysków
- udostępnia pliki i foldery w sieci lokalnej
- konfiguruje udostępnianie plików w sieci lokalnej
- mapuje dyski i udziały sieciowe
- planuje swój rozwój i zdobywanie wiedzy z wykorzystaniem e-learningu

### **Dział 2. Budowa komputera**

Uczeń:

- porównuje podstawowe parametry użytkowe procesorów, pamięci RAM, pamięci masowych, zasilaczy, kart graficznych i sieciowych
- ocenia podstawowe parametry drukarek, skanerów, monitorów
- analizuje projekty komputera sieciowego (podaje ich mocne i słabe strony)
- rozumie pracę procesora i umie ją wyjaśnić za pomocą schematu blokowego

### **Dział 3. Grafika i multimedia**

Uczeń:



- dokonuje masowej zmiany rozdzielczości, rozmiaru, konwersji między formatami
- planuje prace projektowe z uwzględnieniem wykorzystania narzędzi zwiększających efektywność (kopiowanie, klonowanie, grupowanie itp.)
- konwertuje filmy pomiędzy różnymi formatami
- udostępnia filmy w Internecie

#### **Dział 4. Zaawansowana edycja tekstu**

Uczeń:

- konfiguruje własną listę wielopoziomową
- tworzy i edytuje własne style
- dostosowuje podpisy i spisy do swoich potrzeb
- tworzy własny szablon dokumentu (np. dyplom, list, papier firmowy)
- tworzy szablony zawierające pola
- tworzy dokument, korzystając z reguł
- tworzy i edytuje dokumenty z wykorzystaniem recenzji

#### **Dział 5. Arkusz kalkulacyjny**

Uczeń:

- importuje dane z plików TXT
- kopiuje dane z plików HTML
- tworzy zaawansowane formatowanie z wieloma regułami
- tworzy wykresy zawierające wiele serii danych
- stosuje skalę logarytmiczną oraz wykres o dwóch osiach pionowych
- umieszcza na wykresie linie trendu wraz z równaniem
- zarządza prawami dostępu
- rozwiązuje konflikty jednoczesnego dostępu
- filtruje dane, uwzględniając wiele warunków (koniunkcja i alternatywa)
- analizuje dane w zaawansowanej tabeli przestawnej (wielowymiarowej, z filtrowaniem, grupowaniem, wieloma wartościami)

## **Dział 6. Relacyjne bazy danych**

Uczeń:

- projektuje i edytuje formularze elektroniczne
- tworzy bazę danych zawierającą powiązania wiele-do-wielu
- tworzy kwerendę podsumowującą (skróconą)
- projektuje kwerendy funkcjonalne wprowadzające nieodwracalne zmiany

## **Dział 7. Prezentowanie danych**

Uczeń:

- umieszcza i dostosowuje ścieżkę dźwiękową do prezentacji
- kompresuje multimedia
- stosuje animacje obiektów
- zapisuje prezentację w postaci spakowanej na zewnętrznym nośniku
- przeprowadza emisję pokazu slajdów
- wykorzystuje w stylach CSS klasy i pseudoklasy
- stosuje funkcję warunkową
- korzysta z operatorów porównań
- tworzy formularze elektroniczne z obsługą zdarzeń

## **Dział 8. Algorytmy i programowanie**

Uczeń:

- tworzy prosty algorytm liniowy i zapisuje go w postaci schematu blokowego i listy kroków
- stosuje instrukcję warunkową w algorytmie
- tworzy rozgałęzione algorytmy zawierające pętle i zapisuje je w postaci schematu blokowego
- dobiera rodzaj pętli odpowiednio do treści zadania
- stosuje wielokrotnie zagnieżdżoną instrukcję warunkową
- tworzy skrypty zawierające pętle
- określa wynik działania skryptu
- stosuje wielokrotnie zagnieżdżoną instrukcję warunkową

- tworzy programy zawierające pętle
- określa wynik działania programu

### **Dział 9. Projekty międzyprzedmiotowe (tematy alternatywne)**

Uczeń:

- zaangażował się w prace w stopniu bardzo wysokim
- posiada ogromną wiedzę i umiejętności ze wszystkich działów tematu
- był wysoce zaangażowany w prezentację efektów pracy

### **Ocena celująca:**

#### **Dział 1. Usługi internetowe.**

Uczeń:

- aktywnie uczestniczy w życiu Wikipedii lub projektów o zbliżonej koncepcji
- opracowuje trasy wielodniowych wycieczek (również zagranicznych) z uwzględnieniem geolokalizacji GPS oraz map wielowarstwowych (np. utrudnienia w ruchu, natężenie ruchu, zdjęcia i filmy innych użytkowników, artykuły z Wikipedii)
- eksportuje / importuje kontakty z wykorzystaniem plików CSV, wizytówek vCard, telefonu komórkowego
- konfiguruje czytnik poczty (z wykorzystaniem POP3, SMTP, SSL)
- zarządza serwerem FTP (instaluje serwer, tworzy konta, nadaje uprawnienia)
- ustala szczegółowe prawa dostępu do plików z wykorzystaniem uprawnień NTFS
- administruje platformą e-learningową
- tworzy materiały szkoleniowe i umieszcza je w serwisie e-learningowym

#### **Dział 2. Budowa komputera**

Uczeń:

- opiniuje przyszłość technologii komputerowej, wizje, kierunki oraz problemy i ograniczenia rozwoju
- opracowuje projekt komputera sieciowego z uwzględnieniem jego rozbudowy w przyszłości

#### **Dział 3. Grafika i multimedia**

Uczeń:

- dokonuje zaawansowanych przekształceń zdjęć, z wykorzystaniem np. masek i warstw
- projektuje materiały (logo, plakaty, ulotki itp.) na potrzeby szkoły

- tworzy scenariusz i reżyseruje film dydaktyczny

#### **Dział 4. Zaawansowana edycja tekstu**

Uczeń:

- planuje wygląd zaawansowanego dokumentu
- projektuje wygląd i właściwości list, stylów, odwołań i spisów
- projektuje, tworzy i udostępnia szablony druków szkolnych (ta sama szata graficzna)
- projektuje uniwersalny dokument korespondencji seryjnej (np. zaproszenie) wraz z bazą danych
- w projekcie przewiduje przyszłe, wielokrotne użycie dokumentu
- redaguje i zarządza szkolną gazetką internetową
- tworzy, udostępnia i edytuje dokumenty, wykorzystując smart fon

#### **Dział 5. Arkusz kalkulacyjny**

Uczeń:

- eksportuje dane z innych aplikacji
- analizuje i poprawia pliki tekstowe z uwzględnieniem ich późniejszego importu do arkusza kalkulacyjnego
- tworzy niestandardowe typy wykresów (np. skumulowany, giełdowy, bąbelkowy)
- tworzy, udostępnia i edytuje arkusze, wykorzystując smartfon
- tworzy arkusz ze sprawdzaniem poprawności danych
- analizuje dane, wykorzystując sumy częściowe

#### **Dział 6. Relacyjne bazy danych**

Uczeń:

- optymalizuje projekt tabeli (właściwości pól, typy danych, skalowalność i uniwersalność tabeli, minimalizacja użytej pamięci itp.)
- projektuje intuicyjne formularze elektroniczne z uwzględnieniem maksymalnej funkcjonalności
- opracowuje raporty (na podstawie tabel)
- opracowuje raporty (na podstawie tabel i kwerend)
- tworzy zapytania do baz danych w SQL

#### **Dział 7. Prezentowanie danych**

Uczeń:

- przygotowuje prezentację wielomonitorową
- tworzy prezentację w postaci filmu wideo i udostępnia ją w internecie
- tworzy i edytuje skomplikowane struktury plików .css
- zmienia (tworzy własne) .css w systemach CMS
- projektuje i publikuje zaawansowane skrypty zawierające pętle, tablice, obiekty

## **Dział 8. Algorytmy i programowanie**

Uczeń:

- analizuje działania algorytmów (ocenia ich złożoność)
- porównuje dwa alternatywne rozwiązania zadania i udowadnia wyższość jednego z nich
- analizuje działania algorytmów (ocenia ich złożoność)
- porównuje dwa alternatywne rozwiązania zadania i udowadnia wyższość jednego z nich
- tworzy w pełni funkcjonalne skrypty według własnego projektu
- porównuje dwa alternatywne rozwiązania zadania i udowadnia wyższość jednego z nich
- tworzy w pełni funkcjonalne programy według własnego projektu
- porównuje dwa alternatywne rozwiązania zadania i udowadnia wyższość jednego z nich

## **Dział 9. Projekty międzyprzedmiotowe (tematy alternatywne)**

Uczeń:

- był liderem i koordynatorem projektu
- był kreatywnym inicjatorem ewentualnych zmian
- wspomagał innych
- opracował od strony merytorycznej najtrudniejsze części projektu
- posiada ogromną wiedzę i umiejętności ze wszystkich działów tematu
- był wysoce zaangażowany w prezentację efektów pracy