

Wymagania edukacyjne na poszczególne oceny z informatyki

w I Liceum Ogólnokształcącym z Oddziałami Dwujęzycznymi im. Mikołaja Kopernika w Krośnie

1. Ogólne Zasady oceniania

Nauczyciel na początku każdego roku szkolnego (wrzesień) informuje uczniów o wymaganiach edukacyjnych wynikających z realizowanego przez siebie programu nauczania oraz przedstawia uczniom Wymagania edukacyjne z informatyki na poszczególne oceny.

Ocenianie dokonywane będzie według sześciostopniowej skali ocen. W półroczu każdy uczeń otrzyma oceny częściowe w ilości co najmniej „tygodniowa ilość godzin przedmiotu +1”. W przypadku gdy uczeń z powodu nieobecności nie zdobędzie ocen częściowych nauczyciel ma prawo zadać pracę dodatkową po ustaleniu zakresu, tematyki i formy pracy. Kryteria oceniania i forma sprawdzianu bądź pracy kontrolnej będą omówione każdorazowo przed sprawdzianem lub pracą kontrolną.

Ocena śródroczna i roczna nie jest średnią arytmetyczną ocen częściowych. Ocena roczna stanowi podsumowanie rocznej pracy ucznia. Nauczyciel bierze pod uwagę oceny z obu okresów. Ocena jest jawna, umotywowana na każdą prośbę ucznia lub jego rodziców. Ocena półroczna i roczna uwzględnia wszystkie oceny bieżące.

Sprawdziany i inne prace uczniów przechowywane są w szkole do końca roku szkolnego. Sprawdzone i ocenione sprawdziany i ćwiczenia uczeń może otrzymać do wglądu podczas zajęć lekcyjnych na własną prośbę. Jeżeli uczeń nie uczestniczył w sprawdzianie to ma obowiązek zaliczenia go w ciągu 2 tygodni, w uzgodnionym z nauczycielem terminie. Zwrot poprawionych prac następuje w terminie do 14 dni (wyjątek stanowią przypadki losowe np. choroba nauczyciela).

Uczeń nieobecny na lekcji ma obowiązek zapoznania się z tematyką i zakresem wiedzy realizowanymi na zajęciach, na których był nieobecny oraz odrobienia zadania domowego.

Uczeń nieobecny na sprawdzianie ma obowiązek usprawiedliwienia swej nieobecności i ustalenie terminu zaliczenia sprawdzianu.

2. Wymagania ogólne podlegające ocenie

1. Uczniowie otrzymują oceny za posiadane indywidualne umiejętności praktyczne z przedmiotu.
2. Z przedmiotu uczniowie oceniani są za:
 - praktyczne umiejętności obsługi komputera
 - umiejętność posługiwania się wybranym programem komputerowym
 - aktywność i pomysłowość w czasie zajęć stacjonarnych i zdalnych
 - wykorzystanie technik informatycznych w realizowanych zadaniach i projektach

3. Nauczyciel w czasie oceniania uwzględnia:
- zaangażowanie ucznia podczas realizacji danego zadania
 - ukierunkowanie klasy
 - terminowość przesłania prac
4. Sprawdziany mają głównie charakter indywidualnej pracy przy komputerze. Uczeń samodzielnie wykonuje zadania dotyczące programu poznanego w czasie zajęć lekcyjnych. W klasach z rozszerzonym programem przedmiotu Informatyka przewiduje się także sprawdziany pisemne i testy z wiadomości teoretycznych
5. Z przeprowadzanych sprawdzianów stosuje się kryteria ocen zgodne z przepisami zawartymi w WSO
6. Wymagania edukacyjne uwzględniają oceny:
- a) częściowe – uczeń otrzymuje je za:
- pracę na lekcji w czasie trwania półrocza,
 - odpowiedzi ucznia,
 - sprawdziany praktyczne zapowiadane przez nauczyciela,
 - zadania domowe,
 - aktywność w czasie zajęć,
 - inne zadania zlecane przez nauczyciela i wykonywane przez ucznia w czasie trwania PÓŁROCZA (dodatkowa praca , pomoc w realizacji zadań, osiągnięcia w konkursach itp.)
- b) półroczne (końcoworoczne) – ustalane przez nauczyciela na koniec danego semestru (roku szkolnego) na podstawie ocen częściowych jakie uzyskał uczeń w ciągu trwania danego semestru.
7. Przy wystawianiu ocen częściowych w danej klasie nauczyciel prowadzący zajęcia uwzględnia te kryteria z danego zakresu materiału, który został zrealizowany w ciągu semestru w danej klasie – nauczyciel na początku roku szkolnego zapoznaje uczniów jakie wymagania na ocenę częściową i semestralną musi uczeń spełnić w danym roku szkolnym.

3. Ogólne kryteria oceny

Ocenę **celującą** otrzymuje uczeń, który:

- Prowadzi samodzielną i twórczą działalność rozwijającą własne uzdolnienia;
- Biegle posługuje się zdobytymi wiadomościami w rozwiązywaniu problemów teoretycznych lub praktycznych, proponuje rozwiązania nietypowe.
- Posiada wiedzę wykraczającą poza podstawę programową przedmiotu
- Swoją wiedzę wykorzystuje dla potrzeb szkoły
- Startuje w olimpiadach i konkursach przedmiotowych reprezentując szkołę i osiąga w nich wymierne wyniki

Ocenę **bardzo dobrą** otrzymuje uczeń, który:

- Opanował pełen zakres wiedzy i umiejętności określonych programem nauczania informatyki;
- Sprawnie komunikuje się z komputerem za pomocą systemu operacyjnego i w pełni wykorzystuje jego możliwości;
- Swobodnie posługuje się oprogramowaniem użytkowym, umiejętnie dobiera je do wykonywanych zadań;
- Dobrze zna pojęcia informatyczne, występujące w programie nauczania i swobodnie je stosuje;
- Posiadaną wiedzę informatyczną stosuje w zadaniach praktycznych i teoretycznych;

Ocenę **dobrą** otrzymuje uczeń, który:

- Posiadał niepełny zakres wiedzy i umiejętności z przedmiotu określonego programem nauczania w danej klasie;
- Poprawnie stosuje nabyte wiadomości, rozwiązuje samodzielnie typowe zadania teoretyczne i praktyczne;
- Poprawnie posługuje się oprogramowaniem użytkowym;
- Umiejętnie korzysta z pomocy wszelakich środków masowego przekazu

Ocenę **dostateczną** otrzymuje uczeń, który:

- Opanował wiadomości i umiejętności określone programem nauczania informatyki na poziomie nie przekraczającym wymagań zawartych w podstawie programowej;
- Rozwiązuje typowe zadania teoretyczne i praktyczne o średnim stopniu trudności i przy pomocy nauczyciela;
- Stosuje zdobytą wiedzę do celów poznawczych i teoretycznych pod kierunkiem nauczyciela;
- Umie komunikować się z komputerem za pomocą systemu operacyjnego;
- Umie uruchomić omawiane oprogramowanie użytkowe;
- Popęnia liczne błędy merytoryczne;

Ocenę **dopuszczającą** otrzymuje uczeń, który:

- Posiada braki w opanowaniu podstawy programowej informatyki, ale braki te nie przekreślają możliwości uzyskania przez ucznia podstawowej wiedzy z informatyki;
- Rozumie pytania i polecenia;

- Zna pojęcia informatyczne występujące w materiale nauczania;
- Wie, czym zajmuje się informatyka i jakie programy użytkowe są omawiane;
- Poprawnie uruchamia komputer i omawiane programy użytkowe;
- Potrafi zastosować omawiane wiadomości do wykonania bardzo prostych czynności;
- Popełnia liczne błędy merytoryczne;

Ocenę **niedostateczną** otrzymuje uczeń, który:

- Nie opanował wiadomości i umiejętności określonych w podstawie programowej, a braki te uniemożliwiają mu dalsze zdobywanie wiedzy w zakresie tego przedmiotu;
- Nie zna pojęć informatycznych występujących w programie nauczania;
- Nie potrafi zastosować nabytych wiadomości do zadań praktycznych;
- Nie rozumie poleceń i pytań;
- Nie wie, czym zajmuje się informatyka i nie wie, jakie są jej metody;
- Nie potrafi uruchomić omawianego programu użytkowego;
- Nie potrafi komunikować się z systemem operacyjnym;
- W wypowiedziach popełnia liczne błędy merytoryczne;

4. Formy sprawdzania wiadomości i umiejętności

Sprawdziany pisemne

Dłuższe prace pisemne lub zadania praktyczne po każdym dziale tematycznym lub większej partii materiału, zapowiedziane z tygodniowym wyprzedzeniem, oceniane w systemie punktowym, przelicza się poprzez skalę procentową w sposób następujący:

Ocena	Przedział procentowy
Celujący	Od 96% - do 100%
Bardzo dobry	Od 86% - do 95%
Plus dobry	Od 80% - do 85%
Dobry	Od 70% - do 79%
Plus dostateczny	Od 60% - do 69%
Dostateczny	Od 50% - do 59%
Dopuszczający	Od 40% - do 49%
Niedostateczny	Od 0% - do 39%

Krótkie sprawdziany

W ciągu semestru, mogą być niezapowiedziane. Obejmują materiał z trzech ostatnich tematów, mogą dotyczyć zadania domowego.

W przypadku krótkich pisemnych form sprawdzania osiągnięć edukacyjnych (np. kartkówki), nie zawierających poleceń lub zadań spełniających kryteria oceny celującej, obowiązuje następująca skala procentowa:

Ocena	Przedział procentowy
Bardzo dobry	Od 90% - do 100%
Plus dobry	Od 80% - do 89%
Dobry	Od 70% - do 79%
Plus dostateczny	Od 60% - do 69%
Dostateczny	Od 50% - do 59%
Dopuszczający	Od 40% - do 49%
Niedostateczny	Od 0% - do 39%

Odpowiedzi ustne

Sprawdzające bieżącą wiedzę ucznia, obejmują materiał z trzech ostatnich tematów.

Zadania domowe

Odrobione samodzielnie, złożone w formie i czasie ustalonej przez nauczyciela i przedstawionej do wiadomości uczniów.

Praca w grupach

Wcześniej zapowiadana i przygotowana lub odbywająca się bez zapowiedzi.

5. Kryteria szczegółowe oceny

Ocenę celującą otrzymuje uczeń, który:

Samodzielnie wykonuje na komputerze wszystkie zadania z lekcji i zadania dodatkowe. Jego wiadomości i umiejętności wykraczają poza te, które są zawarte w programie informatyki. Jest aktywny na lekcjach i pomaga innym. Ćwiczenia na lekcji wykonuje bezbłędnie, trzeba dostarczać mu dodatkowych, trudniejszych zadań. Bierze udział w konkursach przechodząc w nich poza etap wstępny. Wykonuje dodatkowe prace tj. przygotowanie materiałów pomocniczych na komputerze, pomoc innym nauczycielom w wykorzystaniu komputera na ich lekcjach.

Ocenę bardzo dobrą otrzymuje uczeń, który:

Samodzielnie wykonuje na komputerze wszystkie zadania z lekcji. Opanował wiadomości i umiejętności zawarte w programie informatyki. Na lekcjach jest aktywny, pracuje systematycznie i potrafi pomagać innym w pracy. Zawsze kończy wykonywane na lekcji ćwiczenia i wykonuje je bezbłędnie.

Ocenę dobrą otrzymuje uczeń, który:

Samodzielnie wykonuje na komputerze nie tylko proste zadania. Opanował większość wiadomości i umiejętności zawartych w programie informatyki. Na lekcjach pracuje systematycznie i wykazuje postępy. Prawie zawsze kończy wykonywane na lekcji ćwiczenia i wykonuje je niemal bezbłędnie.

Ocenę dostateczną otrzymuje uczeń, który:

Potrafi wykonać na komputerze proste zadania. Czasem z niewielką pomocą opanował wiadomości i umiejętności na poziomie nieprzekraczającym wymagań zawartych w podstawie programowej informatyki. Na lekcjach stara się pracować systematycznie, wykazuje postępy. W większości wypadków kończy wykonywane na lekcji ćwiczenia.

Ocenę **dopuszczającą** otrzymuje uczeń, który:

Czasami potrafi wykonać na komputerze proste zadania, opanował część umiejętności zawartych w podstawie programowej. Na lekcjach pracuje niesystematycznie, jego postępy są zmienne, nie kończy niektórych wykonywanych ćwiczeń. Braki w wiadomościach i umiejętnościach nie przekreślają umiejętności uzyskania przez ucznia podstawowej wiedzy i umiejętności informatycznych w toku dalszej nauki.

Ocenę **niedostateczną** otrzymuje uczeń, który:

Może być konsekwencją daleko idących zaniedbań w pracy ucznia, gdy nie opanował on podstawowych umiejętności zawartych w podstawie programowej oraz nie zdobył wiadomości i umiejętności niezbędnych dla kontynuowania nauki na wyższym poziomie.

6. Tryb i formy ustalania rocznej oceny klasyfikacyjnej wyższej niż przewidywana.

Uczeń ma prawo ubiegania się o ocenę roczną wyższą od proponowanej przez uczącego po spełnieniu wymagań zawartych w WSO §63. i §63a.

7. Nauczanie zdalne

1. Nauczanie zdalne polega na kontaktowaniu się i pracy nauczyciela z daną klasą poprzez urządzenia takie jak: komputer, tablet, telefon; przy użyciu komunikatorów społecznych, e-dzienników lub innego rodzaju platform.
2. Przy ocenianiu uczniów w czasie nauczania zdalnego obowiązuje skala ocen stosowana dotychczas.
3. Ocenianiu podlegają następujące aktywności uczniów:
 - a. odpowiedzi ustne,
 - b. prace wykonane,
 - c. aktywność na zajęciach,
 - d. terminowość odsyłania prac,
 - e. postawa ucznia wobec przedmiotu.
4. Przesyłanie prac odbywa się w sposób zaproponowany przez nauczyciela (e-dziennik, Microsoft Teams, e-mail, komunikatory społeczne, Platformy E-learningowe).

5. W razie problemów z uczestnictwem w nauczaniu zdalnym wynikającym z różnych przyczyn (w tym sprzętowych) uczeń ma obowiązek powiadomić o tym nauczyciela w jakiegokolwiek formie. Nauczyciel mając wiedzę o takich problemach po stronie ucznia postara się pomóc, lub wypracować inną skuteczną formę komunikacji z uczniem. Brak jakiegokolwiek informacji ze strony ucznia lub rodzica/opiekuna utrudni skuteczną naukę zdalną i może być interpretowany jako negatywna postawa wobec przedmiotu.

8. Szczegółowe wymagania edukacyjne z informatyki na poziomie podstawowym i rozszerzonym

ZAWĘŻONA PODSTAWA PROGRAMOWA 2024

Teraz bajty. Informatyka dla szkoły ponadpodstawowej. Zakres podstawowy. Klasa I

Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne dla klasy I

Autor: Grażyna Koba

MIGRA 2024

Przedstawiamy wymagania na poszczególne oceny szkolne dla klasy I, uwzględniające zmiany wynikające z zawężenia podstawy programowej dla szkoły podstawowej na podstawie rozporządzenia MEN z 2024 roku: *Rozporządzenie Ministra Edukacji zmieniającego rozporządzenie w sprawie podstawy programowej kształcenia ogólnego dla liceum ogólnokształcącego, technikum oraz branżowej szkoły II stopnia.*

Moduł A. Wokół komputera

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

Systemy operacyjne i inne oprogramowanie

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

Moduł B. Wokół dokumentów komputerowych

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

	wie do czego służy podział dokumentu na sekcje; rozmieszcza tekst w kolumnach	zmian, wstawia komentarze		
--	--	--------------------------------------	--	--

Opracowywanie grafiki rastrowej

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

Tworzenie prezentacji multimedialnej

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

Moduł C. Wokół algorytmiki i programowania

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

Tworzenie programów w wybranym języku programowania

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

Moduł D. Wokół Internetu i projektów

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

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

Zadania projektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
omawia etapy tworzenia projektu grupowego; przestrzega zasad korzystania z cudzych materiałów	wyjaśnia, jak przeprowadza się debatę za i przeciw; wyjaśnia pojęcie; <i>prawo autorskie, domena publiczna</i> ; stosuje poznane metody wyszukiwania informacji	prawidłowo zapisuje, przechowuje i udostępnia dokumenty potrzebne do realizacji projektu	pełni rolę koordynatora projektu grupowego; przydziela zadania szczegółowe; scala dokumenty wykonane przez członków grupy	proponuje tematykę własnego projektu, samodzielnie wyznacza zadania szczegółowe i sposób ich realizacji; koordynuje realizację projektu

ZAWĘŻONA PODSTAWA PROGRAMOWA 2024

Teraz bajty. Informatyka dla szkoły ponadpodstawowej. Zakres podstawowy. Klasa II

Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne dla klasy II

Autor: Grażyna Koba

MIGRA 2024

Przedstawiamy wymagania na poszczególne oceny szkolne dla klasy II, uwzględniające zmiany wynikające z zawężenia podstawy programowej dla szkoły podstawowej na podstawie rozporządzenia MEN z 2024 roku: *Rozporządzenie Ministra Edukacji zmieniającego rozporządzenie w sprawie podstawy programowej kształcenia ogólnego dla liceum ogólnokształcącego, technikum oraz branżowej szkoły II stopnia.*

Moduł A. Wokół komputera i sieci komputerowych

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

Bezpieczeństwo i ochrona danych

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

Moduł B. Wokół dokumentów komputerowych

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

Filtry oraz tabele i wykresy przestawne w arkuszu kalkulacyjnym

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

Opracowywanie grafiki wektorowej

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

Moduł C. Wokół algorytmiki i programowania

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

Programowanie wybranych algorytmów

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

Moduł D. Wokół Internetu i projektów

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

Komunikacja i wymiana informacji w Internecie

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wymienia podstawowe zasady pisania listów elektronicznych; podaje przykładowe sposoby komunikacji i wymiany informacji z wykorzystaniem Internetu; zna i stosuje zasady netykiety; podaje przynajmniej dwie korzyści wynikające z rozwoju technologii informacyjno-komunikacyjnych (TIK); jest świadomy istnienia zagrożeń wynikających z rozwoju TIK	poprawnie redaguje listy elektroniczne, dbając o ich formę i treść; omawia wybrane formy komunikacji i wymiany informacji; zna i stosuje zasady netykiety dotyczące form komunikacji, z których korzysta; wyjaśnia, na czym polega komunikacja w czasie rzeczywistym; wie, czym jest chmura obliczeniowa; wymienia przynajmniej dwie możliwości korzystania z chmury; podaje zalety korzystania z komunikacji za pomocą Internetu; wymienia podstawowe zagrożenia wynikające z rozwoju technologii informacyjno-komunikacyjnej	rozdziela poszczególne formy komunikowania się przez Internet; rozdziela poszczególne sposoby komunikacji i wymiany informacji; omawia działanie poczty elektronicznej; zna i stosuje zasady netykiety dotyczące wszystkich form komunikacji; zna i omawia możliwości korzystania z chmury obliczeniowej; omawia szczegółowo zagrożenia wynikające z rozwoju technologii informacyjno-komunikacyjnej (m.in.: cyberprzemoc, anonimowość kontaktów, uzależnienie od komputera); wie, czym jest fake news; omawia korzyści i zagrożenia dotyczące korzystania z różnych form komunikacji i wymiany informacji z wykorzystaniem	potrafi dokonać analizy porównawczej różnych form komunikacji i wymiany informacji, podając opis poszczególnych form i niezbędne wymagania; porównuje metody dostępu do poczty elektronicznej; wyjaśnia, na czym polega korzystanie z oprogramowania w chmurze i na czym polega współdzielenie dokumentów umieszczonych w chmurze; wyjaśnia, jakie korzyści daje rozwój technologii informacyjno-komunikacyjnej osobom z niepełnosprawnościami; bierze aktywny udział w debacie na temat szans i zagrożeń wynikających z rozwoju TIK.	samodzielnie wyszukuje w różnych źródłach informacje na temat nowoczesnych możliwości korzystania z Internetu, np. za pomocą urządzeń mobilnych; wyszukuje w Internecie informacji na temat łączenia się różnych urządzeń znajdujących się w domach czy samochodach z chmurą; potrafi samodzielnie ocenić znaczenie technologii informacyjno-komunikacyjnej w komunikacji i wymianie informacji oraz zna najnowsze osiągnięcia w tej dziedzinie

		Internetu		
--	--	-----------	--	--

Tworzenie stron internetowych

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

Zadania projektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
omawia etapy tworzenia projektu grupowego; wykonuje proste zadania szczegółowe z projektu grupowego; przestrzega zasad korzystania z cudzych materiałów	wyjaśnia, jak przeprowadza się debatę za i przeciw; wyjaśnia pojęcie; <i>prawo autorskie, domena publiczna</i> ; stosuje poznane metody wyszukiwania informacji	prawidłowo zapisuje, przechowuje i udostępnia dokumenty potrzebne do realizacji projektu	pełni rolę koordynatora projektu grupowego; przydziela zadania szczegółowe; scala dokumenty wykonane przez członków grupy	proponuje tematykę własnego projektu; samodzielnie wyznacza zadania szczegółowe i sposób ich realizacji; koordynuje realizację projektu

ZAWĘŻONA PODSTAWA PROGRAMOWA 2024

Teraz bajty. Informatyka dla szkoły ponadpodstawowej. Zakres podstawowy. Klasa III

Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne dla klasy III

Autor: Grażyna Koba

MIGRA 2024

Przedstawiamy wymagania na poszczególne oceny szkolne dla klasy III, uwzględniające zmiany wynikające z zawężenia podstawy programowej dla szkoły podstawowej na podstawie rozporządzenia MEN z 2024 roku: *Rozporządzenie Ministra Edukacji zmieniającego rozporządzenie w sprawie podstawy programowej kształcenia ogólnego dla liceum ogólnokształcącego, technikum oraz branżowej szkoły II stopnia.*

Moduł A. Wokół komputera i sieci komputerowych

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

Wybrane urządzenia cyfrowe

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

Moduł B. Wokół dokumentów komputerowych

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

Wykonywanie podstawowych operacji na relacyjnej bazie danych

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

Projektowanie modeli dwuwymiarowych i trójwymiarowych				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
wyjaśnia różnice między grafiką 2D i 3D; wymienia nazwy programów do tworzenia grafiki 3D; planuje kroki wykonania projektu ogrodu; korzysta z podstawowych narzędzi programu SketchUp	potrafi zainstalować i skonfigurować program SketchUp; realizuje przekształcenie modelu 2D w 3D; korzysta z dodatkowych narzędzi programu SketchUp; umieszcza gotowe elementy z biblioteki	tworzy obiekty z zachowaniem odpowiedniej skali; przekształca pliki graficzne; korzysta z dodatkowych narzędzi programu SketchUp jak Offset, Pull/Push, Orbit; przekształca pliki graficzne, uwzględniając wielkość i jakość obrazów	wykonuje modele trójwymiarowe z zachowaniem skali i wytycznych; tworzy modele, wykorzystując różne możliwości programu	zapoznaje się z możliwościami wybranego programu graficznego, korzystając z Pomocy i innych źródeł; przygotowuje złożone projekty z różnych dziedzin; uczestniczy w konkursach dotyczących grafiki komputerowej

Moduł C. Wokół algorytmiki i programowania

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

Algorytmy szyfrowania w językach C++ i Python

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

Algorytmy na liczbach w językach C++ i Python

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

Moduł D. Wokół Internetu i projektów

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

Więcej na temat tworzenia stron internetowych – projekt

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

Opis założonych osiągnięć ucznia – przykłady wymagań na poszczególne oceny szkolne

opracowany na podstawie materiału edukacyjnego:

Grażyna Koba, *Teraz bajty. Informatyka dla szkół ponadpodstawowych. Zakres rozszerzony. Materiał edukacyjny. Część 1*,
dostępnego na stronie internetowej ir.migra.pl

Autor: Grażyna Koba

MIGRA 2020

A. Wokół komputera

Uwaga: Zgodnie z podstawą programową do informatyki dla szkół ponadpodstawowych: „**Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:**”

Narzędzia i sposoby dbania o komputer				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
potrafi sprawdzić zajętość dysku; usuwa zbędne pliki; odinstalowuje zbędne programy	zna możliwości wybranych narzędzi administracyjnych systemu Windows, m.in. Oczyszczanie dysku; dba, aby system operacyjny był na bieżąco aktualizowany	sprawdza ustawienia pamięci dyskowej komputera; wie, jak włączyć automatyczne zwalnianie miejsca na dysku twardym; wie, na czym polega proces defragmentacji dysku	wie, na czym polega działanie narzędzia Przywracanie systemu; wie, jak przyspieszyć pracę komputera, m.in. sprawdza aktualnie działające programy, wyłącza niepotrzebne programy; omawia, na czym powinna polegać konserwacja sprzętu komputerowego	potrafi zadbać o własny sprzęt komputerowy; samodzielnie zapoznaje się z możliwościami różnych narzędzi administracyjnych systemu Windows; odszukuje inne, niż omówione w temacie A1, możliwości dbania o sprzęt komputerowy; samodzielnie wykonuje niezbędne czynności konserwujące sprzęt komputerowy

Planowanie rozbudowy i zakup nowego komputera				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna i omawia parametry podstawowych elementów komputera; zna i przestrzega zasady bezpiecznej pracy z urządzeniami elektronicznymi, m.in. podczas demontażu obudowy komputera;	wie, co można zrobić, aby rozbudować komputer przenośny, m.in.: podłączenie modemu GSM, zewnętrznego dysku twardego, zwiększenie wielkości pamięci RAM; potrafi sprawdzić parametry komputera	wie, co można zrobić, aby rozbudować komputer stacjonarny, m.in.: wymiana kart rozszerzeń, zwiększenie wielkości pamięci RAM; omawia konfigurację oprogramowania dla komputera przeznaczonego do konkretnych zastosowań, np. do użytku domowego, do gier	potrafi zaprojektować konfigurację nowego komputera przeznaczonego do konkretnych zastosowań, np. do użytku domowego, do gier; potrafi zaprojektować konfigurację oprogramowania przed zakupem komputera przeznaczonego do konkretnych zastosowań, np. do użytku domowego, do gier; potrafi przygotować nośnik z dystrybucją Linuxa i uruchomić za jego pomocą komputer	samodzielnie projektuje rozbudowę komputera przenośnego i stacjonarnego; samodzielnie projektuje konfigurację oprogramowania przed zakupem komputera przeznaczonego do profesjonalnych zastosowań, np. dla grafika; dzieli się z innymi swoją wiedzą i doświadczeniami w/w zakresach; wyszukuje dodatkowe informacje; porównuje funkcjonalności wybranej dystrybucji systemu Linux i systemu Windows

B. Wokół dokumentów komputerowych

Uwaga: Zgodnie z podstawą programową do informatyki dla szkół ponadpodstawowych: „**Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:**”

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

Praca z warstwami obrazu i animacje

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

C. Wokół algorytmiki i programowania

Uwaga: Zgodnie z podstawą programową do informatyki dla szkół ponadpodstawowych: „*Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:*”

Prezentacja algorytmu liniowego w wybranej notacji				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
zna sposoby prezentacji algorytmów; testuje działanie algorytmu liniowego zapisanego w postaci listy kroków i przedstawionego w postaci schematu blokowego; potrafi narysować (odręcznie) schemat blokowy algorytmu liniowego; zapisuje prosty algorytm liniowy w wybranym języku programowania	zna pojęcie specyfikacji zadania i potrafi zapisać specyfikację zadania; zna i stosuje zasady tworzenia listy kroków; przedstawia algorytm liniowy w postaci listy kroków; zna podstawowe zasady graficznego prezentowania algorytmów: podstawowe rodzaje bloków, ich przeznaczenie i sposoby umieszczania w schemacie blokowym; podczas rysowania schematów blokowych potrafi wykorzystać Kształty z edytora tekstu; pisze programy komputerowe realizujące dane algorytmy na podstawie ich list kroków i/lub schematów blokowych	przedstawia dokładną specyfikację dowolnego zadania. analizuje poprawność budowy schematu blokowego; objaśnia dobrany algorytm, uzasadnia poprawność rozwiązania na wybranych przykładach danych; realizuje przykładowy algorytm liniowy w wybranym języku programowania na podstawie ich list kroków i schematów blokowych; testuje program dla różnych danych	potrafi samodzielnie zapoznać się z nowym programem edukacyjnym przeznaczonym do konstrukcji schematów blokowych; potrafi przeprowadzić szczegółową analizę poprawności konstrukcji schematu blokowego; analizuje działanie algorytmu dla przykładowych danych; potrafi wskazać i poprawić błędy w liście kroków czy w schemacie blokowym; Samodzielnie pisze program realizujący algorytm liniowy.	zapisuje specyfikację zadania i przedstawia w postaci listy kroków i schematu blokowego rozwiązanie trudniejszego zadania; w wybranym języku programowania (C++ lub Python) pisze trudniejsze programy realizujące algorytmy przedstawione w postaci list kroków i schematów blokowych; uczestniczy w konkursach/olimpiadach informatycznych

Prezentacja algorytmu z warunkami w wybranej notacji

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

Prezentacja algorytmu iteracyjnego w wybranej notacji

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

Funkcje w wybranym języku programowania

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wie, czym są podprogramy; W wybranym języku programowania definiuje prostą funkcję niezwracającą wartości bez parametrów. Wie, jak wywołać funkcję niezwracającą wartości bez parametrów w programie. Wywołuje taką funkcję w programie.	Wie, na czym polega programowanie strukturalne; Pisze programy, stosując funkcje. Definiuje funkcje niezwracające wartości bez parametrów i z parametrami. Stosuje te funkcje w programach. Wyjaśnia pojęcia zmienna lokalna i zmienna globalna.	Wymienia modele programowania; Rozumie i stosuje zasady programowania strukturalnego. Wyjaśnia, czym są parametry funkcji. Wyjaśnia, czym jest funkcja zwracająca wartość. Porównuje do funkcji niezwracającej wartości. Wskazuje różnice. Wie, na czym polega wywołanie funkcji z parametrami i wywołuje taką funkcję w programach. Definiuje funkcje zwracające wartości bez parametrów i z parametrami oraz stosuje je w programach. Wie, co to jest zasięg zmiennej i jakie ma znaczenie. Deklaruje odpowiednio zmienne lokalne i globalne w programach.	Omawia modele programowania; Rozumie zasady postępowania przy rozwiązywaniu problemu metodą zstępującą. Wyjaśnia, czym różni się programowanie zstępujące od wstępującego. Definiuje funkcje niezwracające wartości i zwracające wartość bez parametrów i z parametrami oraz stosuje je w programach. Wyjaśnia, na czym polega przesłanie zmiennych globalnych. Wyjaśnia, na czym polega przesłanie parametrów funkcji. Modyfikuje programy, stosuje zmienne globalne i lokalne, objaśnia otrzymane wyniki.	Porównuje modele programowania, wskazując różnice. Sprawnie definiuje i stosuje funkcje niezwracające wartości i zwracające wartość bez parametrów i z parametrami w programach. Rozwiązuje przykładowe zadania z matury i olimpiady informatycznej. Potrafi, na przykładzie programu utworzonego według własnego pomysłu, wyjaśnić różnice w stosowaniu zmiennych lokalnych i globalnych, omówić zasięg zmiennych i przesłanie zmiennych. Potrafi, na przykładzie programu utworzonego według własnego pomysłu, wyjaśnić przesłanie parametrów funkcji.

Tablice w wybranym języku programowania

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Analizuje i omawia gotowe przykłady programów, w których są użyte tablice lub listy. Potrafi wyjaśnić, dlaczego do rozwiązania niektórych zadań należy użyć tablic (list).	Zna pojęcia: <i>tablica</i>, <i>lista</i>, <i>zmienna indeksowana</i>. Na bazie przykładów z podręcznika, deklaruje tablicę i/lub listę, wczytuje i wyprowadza elementy tablicy i/lub listy, definiując odpowiednie funkcje w wybranym języku programowania.	Rozróżnia struktury danych: proste i złożone. Podaje przykłady. Wczytuje i wyprowadza elementy tablicy i/lub listy. Definiuje odpowiednie funkcje. Wie, jak odwołać się do elementu tablicy i/lub listy. Potrafi zastosować tablicę (listę) w zadaniach oraz modyfikować program, znaleźć błędy i je poprawić.	Rozumie, na czym polega dobór struktur danych do algorytmu i tworzy programy, dobierając odpowiednie struktury danych do programu. Deklaruje tablicę dwuwymiarową w języku C++. Definiuje odpowiednie funkcje. Definiuje różne listy (w tym dwuwymiarowe) w języku Python. Tworzy programy, dobierając odpowiednie struktury danych do programu.	Omawia podobieństwa i różnice w definiowaniu tablic w języku C++ i list w języku Python Stosuje w programach listy jednowymiarowe i dwuwymiarowe, odpowiednio dobierając określoną strukturę danych (tu: rodzaj listy) do algorytmu. Rozwiązuje przykładowe zadania z konkursów i olimpiady informatycznej, w których należy zastosować tablice i/lub listy i funkcje. Dobiera najlepszy algorytm i odpowiednie struktury danych do rozwiązania postawionego problemu.

Stosowanie instrukcji iteracyjnych

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

Iteracyjna realizacja wybranych algorytmów, w tym algorytmów wyszukiwania

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Potrafi odróżnić algorytm liniowy od algorytmu iteracyjnego. Omawia algorytm znajdowania elementu najmniejszego. Analizuje listę kroków i rysuje schemat blokowy na jej podstawie. Analizuje i omawia gotową listę kroków i schemat blokowy algorytmu Euklidesa w jednej z wersji. Testuje algorytm na podstawie listy lub schematu.	Zna przykładowe algorytmy na liczbach naturalnych; Wie, na czym polega metoda wyszukiwania liniowego i przez połowienie. Zapisuje algorytm znajdowania najmniejszego (największego) elementu w postaci programu. Zna iteracyjną postać algorytmu Euklidesa w wersji z odejmowaniem. Potrafi napisać listy kroków algorytmu i narysować schemat blokowy Euklidesa w wersji z odejmowaniem. Zna metodę „dziel i zwyciężaj”. Zapisuje algorytm Euklidesa w wersji z odejmowaniem w postaci programu w wybranym języku programowania.	Potrafi omówić algorytm naiwny i optymalny jednoczesnego znajdowania największego i najmniejszego elementu w zbiorze. Zna iteracyjną postać algorytmu Euklidesa z resztą z dzielenia. Píše listę kroków tego algorytmu. Potrafi narysować schemat blokowy algorytmu Euklidesa w wersji z resztą z dzielenia. Zapisuje algorytm Euklidesa w wersji z resztą z dzielenia w postaci programu w wybranym języku programowania. Wyjaśnia na przykładzie różnicę między wersją algorytmu Euklidesa z odejmowaniem a wersją z resztą z dzielenia.	Określa liczbę porównań w algorytmie naiwnym i optymalnym znajdowania największego i najmniejszego elementu w zbiorze. Porównuje otrzymane wyniki. Píše listę kroków algorytmu jednoczesnego znajdowania minimum i maksimum z wykorzystaniem metody dziel i zwyciężaj. Zapisuje ten algorytm w postaci programu w języku C++ i/lub Python. Omawia zastosowanie schematu Hornera do obliczania wartości wielomianu; píše listę kroków, rysuje schemat blokowy tego algorytmu i píše program realizujący algorytm obliczania wartości wielomianu według schematu Hornera. Programując w/w algorytmy, definiuje odpowiednie funkcje,	Podaje przykłady problemów, w których można zastosować wyszukiwanie liniowe lub przez połowienie. Píše trudniejsze programy komputerowe, w których wykorzystuje poznane algorytmy. Korzystając z dodatkowych źródeł, wyszukuje informacje o zastosowaniu metody „dziel i zwyciężaj” oraz píše program według własnego pomysłu pokazujący zastosowanie tej metody. Samodzielnie zapoznaje się ze schematem Hornera i zapisuje go w postaci programu, dopierając poprawne struktury danych. Korzystając z dodatkowych źródeł, omawia przykłady zastosowań algorytmu Euklidesa i poznaje trudniejsze algorytmy, np. trwałego małżeństwa,

			dobiera struktury danych. Dba o stosowanie podstawowych zasad programowania.	problem ośmiu hetmanów, algorytm znajdowania liczb bliźniaczych. Potrafi zapisać je w języku programowania.
--	--	--	---	---

Elementy analizy algorytmów

2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wymienia własności algorytmów. Potrafi przeanalizować przebieg prostego algorytmu zapisanego w postaci listy kroków lub w postaci schematu blokowego dla przykładowych danych i ocenić w ten sposób jego poprawność.	Zna i omawia własności algorytmów. Wie, kiedy algorytm jest poprawny. Potrafi przeanalizować przebieg algorytmu (np. obliczania silni) zapisanego w postaci listy kroków lub w postaci schematu blokowego dla przykładowych danych i ocenić w ten sposób jego poprawność. Analizuje program wybranego algorytmu (np. obliczania silni) i ocenia jego poprawność.	Wie, jak sprawdzić, czy algorytm jest skończony. Potrafi ocenić poprawność działania algorytmu i jego zgodność ze specyfikacją. Określa liczbę prostych działań zawartych w algorytmie. Określa liczbę prostych działań zawartych w algorytmie. Potrafi poprawić program, który jest niepoprawny.	Wie, kiedy algorytm jest skończony. Potrafi przeanalizować przebieg algorytmu zapisanego w postaci listy kroków lub w postaci schematu blokowego dla przykładowych danych i ocenić w ten sposób jego skończoność. Analizuje program realizujący wybrany algorytm i ocenia jego skończoność. Modyfikuje program, aby działał poprawnie. Sprawdza poznane własności algorytmów, rozwiązując zadania, m.in. uzasadnia skończoność algorytmu znajdowania największego wspólnego dzielnika (NWD) dwóch liczb naturalnych. Oblicz liczbę operacji porównania w algorytmie wyboru minimum z tablicy zawierającej n losowo uporządkowanych liczb.	Potrafi samodzielnie ocenić poprawność i skończoność wybranych algorytmów. Potrafi samodzielnie wymyśleć zadanie (problem), napisać specyfikację zadania, listę kroków i program realizujący to zadanie. Korzysta samodzielnie z dodatkowej literatury fachowej. Poznane dodatkowe możliwości wybranego języka programowania (np. standardowe funkcje) i stosuje w programach. Rozwiązuje przykładowe zadania z olimpiady informatycznej.

D. Wokół Internetu i projektów

Uwaga: Zgodnie z podstawą programową do informatyki dla szkół ponadpodstawowych: „**Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:**”

Praca w chmurze i zadania projektowe				
2	3	4	5	6
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna etapy pracy nad projektem i bierze udział w pracy grupowej jako członek zespołu. Uczestniczy czynnie w projekcie grupowym, wykonując proste zadania, np. wprowadza dane do bazy i je aktualizuje.	Planuje temat projektu. Bierze aktywny udział w pracy grupowej jako członek zespołu, gromadząc i selekcionując materiały do projektu. Bierze udział w testowaniu projektu. Uczestniczy w przygotowaniu dokumentacji projektu, korzystając z edytora tekstu; ewentualnie z wybranego szablonów.	Korzystając z chmury, potrafi udostępnić pliki, linki do folderu; umożliwić współdzielenie danego folderu. Realizuje projekt na zadany (lub samodzielnie wybrany) temat zgodnie z etapami projektowania; przygotowuje dokumentację projektu. Współpracuje w grupie, wykonując projekt na temat projektowania zakupu nowego zestawu komputerowego oraz oprogramowania dla ucznia szkoły ponadpodstawowej; oraz wpływu trendów w historycznym rozwoju pojęć i metod informatyki oraz technologii na możliwości rozwoju grafiki komputerowej.	Korzysta z oprogramowania dostępnego w chmurze, m.in. tworzy dokumenty w edytorze tekstu i umieszcza w chmurze. Współdzieli je z innymi współużytkownikami. Przygotowuje wybrane zadanie szczegółowe zgodnie z etapami przygotowania projektu. Gromadzi materiały i inne pomoce, opracowuje dokumentację projektu, wykorzystując m.in. możliwości pracy w chmurze. Udostępnia pliki, linki do folderu. Wyszukuje informacje na e-platformach do e-nauczania. Prezentuje wykonane zadanie projektowe wykorzystaniem projektora innych przygotowanych pomocy lub materiałów. Inicjuje dyskusje.	Sprawnie posługuje się środowiskiem przeznaczonym do współpracy i realizacji projektów zespołowych, w tym środowiskiem w chmurze. Potrafi pełnić funkcję koordynatora grupy. Koordynuje wykonywanie zadań szczegółowych na poszczególnych etapach. Zarządza folderami (współdzieleniem) i pracą nad dokumentami, w tym dokumentacją projektu. Ustala sposób prezentacji projektu i wyznacza osobę (osoby) do prezentacji. W miarę możliwości współtworzy zasoby udostępniane na platformach do e-nauczania. Przygotowuje projekt na wybrany przez siebie temat.