

Ciekawi.
Świata

Informatyka

Program nauczania dla szkół ponadgimnazjalnych

Zakres rozszerzony

Wojciech Hermanowski

MOPERON

Edukacja jest podrozg

Spis treści

1. Wstęp	3
2. Szczegółowe cele kształcenia i wychowania	4
3. Treści edukacyjne	26
4. Sposoby osiągnięcia celów kształcenia i wychowania	30
5. Opis założonych osiągnięć ucznia	37
6. Propozycje kryteriów oceny i metod sprawdzania osiągnięć ucznia	38

1. Wstęp

Komputer i sieć Internet są ważnym elementem życia młodego człowieka. Nikt nie wyobraża sobie skutecznej pracy, nauki i rozrywki bez ich użycia. Dlatego prawidłowe, bezpieczne i efektywne korzystanie z nowoczesnych technologii informatycznych jest jedną z najważniejszych umiejętności współczesnego człowieka.

Uczeń przystępujący do IV etapu kształcenia dysponuje sporą wiedzą i umiejętnościami w posługiwaniu się komputerami. W związku z tym zadaniem nauczyciela przedmiotu Informatyka jest nie tylko zrealizowanie założeń podstawy programowej, ale także uporządkowanie i wykorzystanie wiedzy i umiejętności już nabytych. Będzie to wymagało dodatkowego wysiłku i indywidualizacji pracy z uczniami o różnym stopniu zaawansowania. Nie można bowiem dopuścić do sytuacji, w której część uczniów będzie musiała wykonywać zadania z ich punktu widzenia infantylne, mogąc w tym samym czasie zdobywać nowe umiejętności. Oczywiście, podobnie sprawa wygląda z uczniami, których stan wiedzy jest niższy. Dla nich zbyt wysokie wymagania będą czynnikiem zniechęcającym do wykonywania ćwiczeń, a także powodem problemów natury psychologicznej. Bez względu na poziom zaawansowania zadań muszą one mieć określony cel, który uczeń pozna przed przystąpieniem do pracy.

Nauka Informatyki na poziomie rozszerzonym obejmuje także zaawansowane metody tworzenia grafiki i multimediiów. Uczniów czeka także nauka niektórych metod numerycznych i języków programowania. Odpowiednio dobrane przykłady zadań i otwartość na różnorodność rozwiązań wpłyną na rozwój umiejętności algorytmicznego myślenia i wykorzystania metod numerycznych do rozwiązywania problemów informatycznych. Wszystkie omawiane zagadnienia powinny łączyć się w całość i w efekcie prowadzić do opracowywania przez uczniów zaawansowanych projektów.

Najważniejszym czynnikiem osiągnięcia sukcesu będzie pełna realizacja programu nauczania. Z tym związane jest wcześniejsze przygotowanie pracowni, materiałów ćwiczeniowych, skorzystanie z odpowiedniego podręcznika i platformy zdalnego nauczania oraz efektywne wykorzystanie czasu.

2. Szczegółowe cele kształcenia i wychowania

Przedmioty informatyczne we współczesnej szkole to nie tylko nauka posługiwania się komputerem i jego programowania. Coraz więcej aspektów życia codziennego łączy się z siecią, która odzwierciedla zachowania użytkowników czy zjawiska socjologiczne przenoszone z realnego świata. W większości przypadków jedynym miejscem kształtującym postawy wobec technologii informacyjnej i jej usług jest szkoła. W czasie zajęć w pracowni informatycznej cele wychowawcze muszą być realizowane równoległe z celami dotyczącymi podstawy programowej. Część z nich jest także w niej uwzględniona.

Sposób osiągnięcia opisanych celów zależy od etapu kształcenia.

Cel ogólny	Cele szczegółowe	Uwagi
Świadome stosowanie przepisów i zasad bezpiecznego eksploataowania urządzeń elektrycznych oraz zachowania przy stanowisku pracy	Uczeń: – zna i umie zinterpretować poszczególne punkty regulaminu pracowni, – potrafi ocenić wzrokowo stan urządzenia elektrycznego (komputera) i przewodów doprowadzających zasilanie i na tej podstawie ocenia bezpieczeństwo pracy z tym urządzeniem, – zna podstawowe zasady bezpiecznego użytkowania i podłączania urządzeń peryferyjnych, – nie stwarza zagrożeń: odpowiednio eksploatuje urządzenia w pracowni, zachowuje ostrożność podczas poruszania się po pracowni, np. w czasie pracy w zespole.	Regulamin i wynikające z niego zasady powinny być omówione na pierwszych zajęciach w pracowni i stosowane w czasie wszystkich następnych lekcji. Konieczne jest wpisanie odpowiedniego tematu do dziennika i złożenie podpisu. Jest to ślad po przeprowadzeniu lekcji na temat zachowania w pracowni.
Przestrzeganie zasad netykiety i współużytkowania sieci komputerowej	Uczeń: – zna i stosuje podstawowe zasady obowiązujące podczas korzystania z usług sieciowych, w tym: komunikatorów, forów, poczty itp., – zachowuje ostrożność: ustala i chroni bezpieczne hasła, nie	Oprócz zajęć poświęconych tej tematyce należy zwrócić uwagę na zachowanie uczniów w czasie zajęć i dyskretnie kontrolować ich działania w sieci. W przypadku naruszenia zasad należy oddziaływać

	<p>udostępnia informacji osobistych, chroni swoje zasoby,</p> <ul style="list-style-type: none"> – szanuje współuczestników usług sieciowych, – dba o swój wizerunek w sieci, – unika agresji w sieci, – zna podstawowe prawa dotyczące ochrony wizerunku i prywatnych danych. 	wychowawczo – nie lekceważyć.
Wzrost świadomości na temat przestępczości w sieci	<p>Uczeń:</p> <ul style="list-style-type: none"> – wie, jak unikać zagrożeń przestępczą działalnością w sieci, – wie, które działania w sieci noszą znamiona przestępstwa lub wykroczenia, – zgłasza osobie starszej (rodzicom, nauczycielowi itp.) niepokojące go wydarzenia w sieci, np. molestowanie, próby wyłudzeń, ośmieszanie, oczernianie i ataki na siebie lub bliskich. 	<p>Na zajęciach poświęconych tej tematyce należy również zwrócić uwagę na zachowanie uczniów w czasie zajęć i dyskretnie kontrolować ich działania w sieci.</p> <p>W przypadku zauważenia niepokojących objawów należy interweniować. Nie wolno lekceważyć takich sytuacji.</p>
Umiejętność pracy w zespole	<p>Uczeń:</p> <ul style="list-style-type: none"> – umie współpracować z innymi członkami grupy, szanuje hierarchię zespołu, – prowadzi dyskusje, broni swoich racji, umie zawierać kompromisy, – nie wyręcza się pracą innych. 	<p>Cel powinien być realizowany podczas pracy nad wspólnymi projektami.</p> <p>Należy zwrócić uwagę na dobór członków zespołu.</p> <p>Nie wolno ingerować w pracę zespołu, ale czuwać nad odpowiednimi relacjami wśród jego członków.</p>
Poszanowanie dla własności intelektualnej	<p>Uczeń:</p> <ul style="list-style-type: none"> – wie, czym są prawa autorskie i majątkowe, a zwłaszcza prawa do dzieł rozpowszechnianych na nośnikach cyfrowych, – szanuje prawa autorskie twórców i kolegów oraz dba o przestrzeganie własnych, do zdjęć, tekstów, zadań domowych itp. 	<p>Uczniowie często lekceważą te prawa, uważając, że co w sieci, to za darmo. Jest to zjawisko powszechne w grupie młodzieży młodszej i gimnazjalnej. Im szybciej uczniowie będą świadomi praw twórców, tym bardziej będą szanować ich pracę, w tym także kolegów, nauczycieli, autorów podręczników i publikacji itd.</p>

Szanowanie stanowiska pracy	<p>Uczeń:</p> <ul style="list-style-type: none"> – nie niszczy sprzętów w pracowni i dba o ich właściwy stan, – nie zmienia bez pozwolenia konfiguracji systemów i programów. 	<p>Kolejna klasa przychodząca do pracowni powinna zastać podstawowy stan systemów i urządzeń.</p> <p>Jeśli uczniowie nie będą samodzielnie, bez polecenia, zmieniać ich ustawień, nauczyciel nie będzie musiał tracić czasu na regulację tych zasobów i sprzętów, a dzięki temu zajęcia potoczą się sprawnie.</p>
-----------------------------	---	---

Poniżej przedstawiono cele kształcenia z podziałem na rozdziały w podręczniku.

Systemy operacyjne. Sieci komputerowe.

Wiedza. Uczeń:	Umiejętności. Uczeń:
Pamięta podstawowe informacje dot. systemu operacyjnego i budowy komputera z 2, 3, 4 (zakres podstawowy) okresu nauki	Umie nazwać i scharakteryzować poszczególne elementy komputera i sprawnie posługuje się systemem operacyjnym.
Wie, do czego służą programy narzędziowe i diagnostyczne	Umie posługiwać się programami diagnostycznymi, np. SIW i interpretować wyniki ich działania. Określa na podstawie wyników działania tych programów moc komputera i kwalifikuje go do odpowiednich zastosowań.
Wie, jak powinna wyglądać struktura zespołów pracujących nad opracowaniem danego projektu	Umie zorganizować pracę zespołu pracującego nad dowolnym projektem realizowanym za pomocą komputerów.
Wie, jakie usługi w chmurach informatycznych mogą usprawnić pracę zespołu	Zna podstawową strukturę chmury informatycznej i rodzaje programów, jakie można spotkać w popularnych i darmowych chmurach.
Zna różne komunikatory sieciowe i wie, jakie mogą mieć zastosowanie w pracy zespołu i e-pracy	Dokonuje wyboru komunikatora zgodnego z zapotrzebowaniem indywidualnym lub grupy.
Wie, na czym polega współpraca komputerów i urządzeń sieciowych w chmurze informatycznej	Umie wyznaczyć wymagania chmury dla komputerów członków zespołu.
Zna podstawowe cechy chmury informatycznej, w tym nieograniczony zasięg, małe wymagania	Znając cechy chmury informatycznej, umie zorganizować strukturę zespołu pracującego nad

sprzętowe, nieograniczona przestrzeń dyskowa dostępna z każdego komputera w sieci, współdzielenie dokumentów, integracja usług i programów, doskonałe środowisko do e-pracy	projektemnp. Szkolnego Portalu Informacyjnego, wybrać odpowiednią do tego zadania chmurę, przydzielić uprawnienia.
Wie, jakie są różnice w pracy zespołu w chmurze i poza nią	Umie wykorzystać usługi sieciowe, takie jak e-mail, komunikator, FTP do pracy zespołu poza chmurą.
Zna kilka adresów darmowych chmur informatycznych, w tym Google i Windows Live	Zakłada konto w chmurze informatycznej, umie uruchamiać jej programy i narzędzia. Umie wykorzystać chmurę informatyczną do usprawnienia organizacji pracy zespołu np. za pomocą kalendarza chmury.
Zna zasady współdzielenia plików w chmurze informatycznej, w tym Google i Windows Live	Wykorzystuje opcję współdzielenia plików w chmurze informatycznej do wspólnego z kolegami opracowania dokumentów projektu, np. artykułów Szkolnego Portalu Informacyjnego.
Zna możliwości chmur w zakresie publikacji stron internetowych, np. w Google	Tworzy prostą witrynę bezpośrednio w chmurze informatycznej, np. Google, zawierającą informacje dla członków zespołu.
Wie, na czym polega e-praca	Umie wymienić zalety i wady e-pracy. Wskazuje zastosowanie e-pracy w kilku zawodach, w tym w takim, który chce wykonywać.
Zna charakterystyczne podstawowe cechy różnych systemów operacyjnych, w tym Windows, Linux, MacOS, Android, na podstawie których możliwa jest ich identyfikacja	Na podstawie wyglądu pulpitów i funkcji systemu określa jego nazwę. Przyporządkowuje klasy i typy komputerów przystosowanych do pracy z danym systemem operacyjnym.
Wie, jak znaleźć najnowsze wersje darmowych systemów operacyjnych, w tym Linux Ubuntu	Umie pobrać najnowszą wersję systemu Linux np. Ubuntu, zapisać ją na płycie butowalnej i uruchomić w komputerze. Umie instalować systemy operacyjne, w tym Windows i Linux.
Wie, na czym polega praca wirtualnej maszyny	Instaluje wirtualną maszynę, np. VirtualBox, w systemie Windows i uruchamia w nim inny system, np. Linux Ubuntu.
Zna nazwy alternatywnych systemów operacyjnych np. uruchamianych w przeglądarce – eyeOS	Uruchamia w przeglądarce system operacyjny chmury informatycznej, np. eyeOS.
Zna warstwowy model komputera z systemem operacyjnym Windows	Posługuje się programem Dane o systemie w celu odczytania danych systemu. Umie tworzyć konta użytkowników.
Zna podstawowe cechy i zadania systemu Windows, w tym współpraca z BIOS, obsługa pamięci masowych, zarządzanie strukturą danych, obsługa urządzeń peryferyjnych,	Umie śledzić aktywność systemu i programów komputerowych za pomocą programów diagnostycznych, np. systemowym Monitorem zasobów. Modyfikuje interfejs użytkownika i

udostępnianie interfejsu graficznego, umożliwienie instalacji i uruchamiania programów komputerowych, połączenie z sieciami, zapewnienie bezpieczeństwa danych	dostosowuje go do potrzeb różnych użytkowników.
Wie, jak pobrać dodatki do systemu Windows	Pobiera, selekcjonuje i instaluje niektóre dodatki systemu Windows, w tym Poczta, usługi Windows Live, Messenger, Windows Live Movie Maker, Microsoft Security Essentials.
Zna strukturę menu systemu Linux, w tym Ubuntu	Znajduje analogie pomiędzy interfejsami systemów Windows i Linux. Posługuje się niektórymi programami systemu Linux, w tym diagnostycznymi narzędziami systemowymi np. Monitor Systemu, i umie ocenić informacje przez nie przekazywane. Porównuje te programy do podobnych z systemu Windows, np. Monitor systemu z Monitorem zasobów.
Wie, do czego służy terminal systemu Linux. Wie, jak znaleźć listę poleceń systemu Linux uruchamianych w Terminalu systemu.	Korzysta z terminala systemu Linux, wydając systemowi podstawowe polecenia, w tym ls i mkdir.
Wie, jakie programy komputerowe oferuje dana dystrybucja systemu Linux, w tym pakiet biurowy OpenOffice lub LibreOffice, przeglądarki stron internetowych, gFTP, Gnote, programy do zapisu na płytach optycznych, komunikatory, np. Kadu	Umie posłużyć się Centrum oprogramowania np. systemu Linux Ubuntu do pobrania i zainstalowania darmowych programów, w tym edytorów grafiki, odtwarzaczy multimedialnych, edytora skryptów, edytorów schematów elektrycznych itp.
Wie, jak korzystać z chmury informatycznej systemu Linux, np. Ubuntu	Korzysta z podstawowych funkcji chmury Linux Ubuntu
Wie, na czym polega przywracanie systemu i na czym polega tworzenie punktów przywracania	Umie utworzyć punkt przywracania systemu i gdy zajdzie taka potrzeba, przywraca system.
Wie, jakie znaczenie ma regularna aktualizacja systemu operacyjnego	Umie przeprowadzić selektywną aktualizację systemu, odrzucając mniej znaczące elementy oferowane przez producenta systemu.
Wie, jakie programy mogą pomóc w utrzymaniu w sprawności systemu operacyjnego, np. CCleaner	Sprawnie posługuje się programami narzędziowymi, w tym CCleaner, do utrzymania odpowiedniego stanu systemu operacyjnego.
Wie, na czym polega defragmentacja dysku i czemu ona służy. Wie, jak ustawić opcję automatycznej defragmentacji.	Umie przeprowadzić defragmentację dysku komputera za pomocą programu systemowego Defragmentator dysku.
Zna warstwowy model komunikacji sieciowej OSI i TCP/IP. Wie, na czym polega różnica pomiędzy tymi modelami sieci.	Opisuje warstwy modelu OSI, drogę informacji w komunikacji sieciowej i warstwowy model TCP/IP. Porównuje oba modele.
Wie, jak transportowane są dane w modelu TCP/IP. Zna pojęcie ramki i nagłówka.	Umie opisać funkcje ramki i nagłówków i urządzenia sieciowe w modelu TCP/IP.
Wie, jak identyfikowane są komputery w	Używa polecenia tracert i programu

sieciach. Zna pojęcie protokół. Wie, jak śledzić trasę połączeń sieciowych, w tym z użyciem polecenia tracert i programu VisualRoute Lite Edition.	diagnostycznego, np. VisualRoute Lite Edition, do śledzenia drogi połączenia sieciowego z dowolną stroną internetową.
Zna klasy adresów sieciowych. Wie, jak zbudowany jest adres IP. Wie, czym jest maskowanie i maska oraz brama i DNS.	Umie określić adres sieci na podstawie maski. Umie skonfigurować połączenie z Internetem na podstawie znajomości maski, adresu bramy i DNS. Ustawia automatyczne łączenie się komputera z wykrytą siecią i Internetem.
Wie, jaka jest różnica pomiędzy adresem fizycznym a symbolicznym	Posługuje się poleceniem ping do sprawdzenia połączenia sieciowego z komputerem w sieci i określenia adresu fizycznego serwera dowolnej strony www. Posługuje się poleceniem ipconfig w celu odczytania pełnej konfiguracji karty sieciowej danego komputera.
Wie, jak zoptymalizować działanie sieci, w tym bezprzewodowej	Wykorzystuje program diagnostyczny, np. inSSIDer, do analizy ruchu w sieci bezprzewodowej i ustalenia optymalnego kanału.
Wie, jaka jest rola poszczególnych elementów bezprzewodowej sieci komputerowej	Konfiguruje podstawowe urządzenia sieci bezprzewodowej Access Point i karty sieciowe.
Wie, jakie znaczenie ma odpowiednie zabezpieczenie sieci bezprzewodowej i jakie są różnice pomiędzy stosowanymi zabezpieczeniami	Zabezpiecza sieć bezprzewodową w dostępnych standardach, w tym WAP, WPA i WPA2, z zastosowaniem PSK.
Wie, jakie dane są potrzebne do skonfigurowania połączenia z Internetem	Konfiguruje router i łączy za jego pośrednictwem sieć lokalną z Internetem.
Wie, jak zabezpieczyć połączenie klient-serwer	Posługuje się konsolą MMC systemu Windows, tworząc szablony zabezpieczeń i odpowiednio blokując konta. Chroni konta przed wielokrotnymi próbami wpisywania haseł. Precyzuje sposób tworzenia haseł przez użytkowników.
Wie, na czym polega udostępnianie zasobów w sieci i z jakimi zagrożeniami się to wiąże	Udostępnia pliki i foldery w sieci lokalnej.
Wie, czym jest protokół transmisji http i zna podstawowe żądania wymiany danych z serwerem – GET, HEAD, PUT, DELETE	Umie uruchomić w systemie Windows serwer IIS i udostępniać strony internetowe.
Zna współczesne tendencje rozwoju środków przesyłania informacji, w tym zabezpieczenia plików, np. e-booków PDF, EPUB lub ZNO	Znajduje informacje na temat rozwoju środków i technik przesyłania informacji w sieciach.
Zna pojęcie kanału RSS i wie, jakie jest jego zastosowanie	Posługuje się programami do zarządzania kanałami RSS, np. Paseczek.
Wie, czym jest transmisja strumieniowa i czym	Odbiera transmisje strumieniowe z sieci. Przesyła

różni się od standardowej transmisji danych	multimedia, w tym filmy, za pośrednictwem programów do transmisji strumieniowej, np. VLC.
Wie, jak znaleźć w sieci stacje radiowe i telewizyjne	Odbiera programy internetowych stacji radiowych i telewizyjnych.

Grafika komputerowa i wizualizacja danych

Wiedza. Uczeń:	Umiejętności. Uczeń:
Wie, jak powstaje cyfrowy obraz wyświetlany na ekranie monitora	Zna uproszczoną budowę karty graficznej, umie ocenić jej wydajność, nazywa złącza kart graficznych i opisuje ich przeznaczenie.
Wie, jak powstają barwy na ekranie monitora	Umie opisać addytywny model kolorów używany do wyświetlania obrazów na ekranie monitora.
Wie, jaki wpływ na liczbę wyświetlanych kolorów ma liczba bitów opisujących każdą ze składowych RGB	Prezentuje barwy w zależności od wartości składowych RGB, np. w opcji ustalania koloru tła w edytorze tekstu.
Zna subtraktywne modele kolorów CMY i CMYK i wie, kiedy się je stosuje. Wie, gdzie znajduje zastosowanie model percepcyjny kolorów.	Umie posłużyć się opcjami ustawień kolorów modelu percepcyjnego w edytorze grafiki, np. GIMP.
Zna pojęcie mapy bitowej i jej podstawowe cechy wpływające na jakość grafiki i sposób jej edycji. Zna pojęcie rozdzielczości podawanej w DPI.	Umie dobrać rozdzielczość mapy bitowej do rodzaju i wielkości publikacji.
Wie, jakie są konsekwencje zastosowania nieodpowiedniej rozdzielczości dla danego typu grafiki, np. rysunków technicznych, map, fotografii	Skanuje dokumenty z odpowiednio dobraną rozdzielczością i głębią kolorów.
Wie, czym różni się edycja i zapis grafiki rastrowej od wektorowej	Umie praktycznie wykazać różnice pomiędzy grafiką rastrową a wektorową, korzystając z podstawowych narzędzi odpowiednich edytorów.
Zna podstawowe formaty plików graficznych i różnice pomiędzy nimi, w tym także typ RAW i pliki z kompresją bezstratną. Wie, jak może zmienić się rozmiar pliku w związku z konwersją do innego formatu, zmianą stopnia kompresji lub rozdzielczości.	Rozpoznaje na podstawie rozszerzeń nazwy plików formaty zapisu grafiki przez różne edytory, w tym: Adobe Photoshop, GIMP, Corel Draw, Corel Photo-Paint oraz InkScape.
Zna sposoby zmiany formatu pliku graficznego na inny i konsekwencje, jakie to za sobą pociąga. Zna przeznaczenie poszczególnych formatów graficznych, w tym przeznaczonych do publikacji na stronach internetowych, do druku i do prezentacji multimedialnych.	Konwertuje różne formaty plików graficznych, zmieniając typ kompresji i rozmiary grafiki przy pomocy edytorów graficznych i specjalizowanych programów, np. Easy Graphic Converter.

Zna możliwości kilku popularnych edytorów grafiki rastrowej, w tym GIMP, Adobe Photoshop, pixlr.com, Corel Photo-Paint, umie je porównać i określić obszary zastosowania tych programów	Porównuje możliwości edytora GIMP i pixlr.com z chmury informatycznej. Posługuje się narzędziami edycyjnymi, świadomie zmieniając ich parametry w celu osiągnięcia optymalnego lub zaplanowanego celu.
Wie, czym są warstwy i jak je wykorzystać	W edytorach grafiki, np. GIMP lub pixlr.com, posługuje się warstwami do kolorowania zdjęć, retuszu, montażu i korekty fotografii.
Zna działanie filtrów różnych edytorów i sposoby ich wykorzystania	Stosuje filtry graficzne w edytorach GIMP lub pixlr.com do modyfikacji grafiki lub zdjęć. Sprawdza doświadczalnie ich działanie.
Wie, do czego służy histogram zdjęcia i na czym polega jego analiza	Posługuje się histogramem do określenia poprawności naświetlenia zdjęcia i analizy obrazu, np. w edytorze GIMP.
Zna narzędzia i sposoby korekty jasności i kolorów w edytorach grafiki, w tym za pomocą krzywych kolorów. Zna różne techniki wykonywania i elektronicznego wywoływania zdjęć, np. HDR i panorama.	Koryguje jasność i kolory na zdjęciach za pomocą krzywych koloru i innych narzędzi dostępnych w edytorze.
Wie, jak wykorzystać warstwy i narzędzia zaznaczania do retuszu fotografii	Retuszuje fotografię, usuwając niepożądane elementy za pomocą narzędzi wycinania i wklejania odpowiednio skorygowanymi parametrami oraz warstw i skalowania. Używa narzędzi alternatywnych do edytorów, programów np. albumów fotograficznych, np. Picasa, do podstawowej korekty zdjęć.
Wie, jakich narzędzi użyć do korekcji perspektywy i korekty portretów	Używa narzędzi do korekty perspektywy i proporcji zdjęcia. Stosuje narzędzia do korekcji drobnych detali, np. pieczętki, rozmywania itp. Dla tych narzędzi używa zmiennych parametrów. Używa warstw i narzędzi malarskich do zmiany kolorów detalu z zastosowaniem poziomów przezroczystości. Stosuje filtry do uzyskania odpowiedniego efektu dla całego zdjęcia, np. sepia, wybiórczy kolor.
Wie, jakie narzędzia edytorów grafiki rastrowej mogą służyć do tworzenia grafiki komputerowej, w tym narzędzia programu GIMP	Tworzy za pomocą edytorów grafiki rastrowej ozdobne napisy, banery i elementy graficzne stron www i prezentacji. Używa narzędzia edytora, np. filtru Mapa obrazu z GIMP, do wygenerowania fragmentu kodu HTML aktywującego baner lub element graficzny.
Zna techniki łączenia elementów graficznych w jednej kompozycji. Wie, jak dopasować	Używa różnych narzędzi wycinania fragmentów zdjęcia. Precyzyjnie wycina fragmenty zdjęcia.

wklejany fragment zdjęcia, np. z zachowaniem proporcji i kąta padania światła.	Przenosi je do innego projektu z zastosowaniem schowka, importu, a także opcji skalowania warstw. Wykonuje połączenie fragmentów zdjęć z grafiką komputerową, np. z zastosowaniem gradientów. Wykonuje nagłówki dokumentów lub www (stron internetowych), łącząc grafikę z fragmentami zdjęć.
Wie, które z edytorów służą do edycji grafiki wektorowej, w tym Corel Draw, InkScape, SketchUp, CadStd Lite, OpenOffice.org Draw lub LibreOffice Draw	Rozpoznaje różne edytory grafiki wektorowej, w tym systemy CAD, 3D i uniwersalne. Porównuje ich narzędzia, znajduje analogie i różnice.
Wie, jakiego typu edytora użyć do tworzenia ulotek, plakatów itp. Wie, jak przygotować projekt do druku na drukarce i w zakładzie poligraficznym.	Planuje rozmieszczenie elementów projektu, np. plakatu promującego szkołę, i wykonuje poszczególne jego elementy z zastosowaniem napisów, narzędzi rysowania figur geometrycznych, gradientów, krzywych, wypełnień i filtrów. Wykorzystuje program InkScape. Przygotowuje gotowy projekt do wydruku na drukarce lub w zakładzie poligraficznym. Eksportuje gotowy projekt do postaci mapy bitowej. Umieszcza metadane w pliku projektu.
Wie, czym różni się edytor grafiki 2D od 3D i które narzędzia decydują o przydatności edytora do tworzenia grafiki 3D	Rysuje obiekty trójwymiarowe za pomocą narzędzi edytora grafiki 3D, np. SketchUp. Tworzy trójwymiarowe elementy stron www i prezentacji multimedialnych, np. strzałki, odnośniki itp. Tworzy trójwymiarowe modele budynków.
Zna podstawowe formaty plików wideo. Wie, jakie kodeki mogą zostać użyte do ich utworzenia.	Konwertuje pliki wideo do postaci zgodnej z miejscem i sposobem publikacji filmu. Dobiera odpowiednie kodeki i korzysta z narzędzi do konwersji, np. Any Video Converter.
Wie, jak można badać kształt wykresu funkcji za pomocą arkusza kalkulacyjnego	Sporządza arkusz do tworzenia wykresu funkcji np. kwadratowej. Prawidłowo wybiera rodzaj wykresu i jego parametry. Prawidłowo dobiera parametry i argumenty funkcji. Bada zmiany kształtu wykresu w zależności od zmian parametrów. Korzysta z automatycznych narzędzi wypełniania tabel arkusza, np. przeciąganie i zaznaczanie.
Wie, jak znaleźć dane statystyczne i wykorzystać je do wizualizacji zachodzących procesów	Tworzy wykresy na podstawie pobranych i zaimportowanych do arkusza danych statystycznych. Odpowiednio do ich rodzaju dobiera typ wykresu. Prawidłowo opisuje osie.

Wie, jakie funkcje arkusza kalkulacyjnego można wykorzystać do opracowania, wizualizacji i symulacji wyników doświadczeń	Używa arkusza kalkulacyjnego do symulacji a następnie opracowania i wizualizacji wyników doświadczeń np. z fizyki. Tworzy odpowiednie tabele do zapisywania wyników doświadczeń. Tworzy odpowiednie wykresy i tabele z wynikami do analizy. Wykorzystuje arkusz do obliczenia błędów wyników doświadczeń.
Zna kilka programów do tworzenia arkuszy	Korzysta z MS Excel, OpenOffice.org Calc lub LibreOffice Calc i arkuszy z chmur informatycznych, w tym Google i Windows Live.

Kody binarne. Przyszłość informatyki.

Wiedza. Uczeń:	Umiejętności. Uczeń:
Wie, jakich standardów używa się do szyfrowania poczty elektronicznej – PGP i S/MIME. Wie, czym są certyfikaty i klucze, w tym publiczny. Wie, na jakiej zasadzie funkcjonuje klucz publiczny symetryczny i asymetryczny.	Zabezpiecza plik w pakiecie MS Office i LibreOffice Writer (OpenOffice.org Writer). Umie szyfrować pocztę elektroniczną za pomocą darmowych programów szyfrujących.
Zna podstawy kodów binarnych. Wie, jaka jest ich budowa i jak zbudowane są kody wagowe. Zna metody zmiany postaci liczny z binarnej na dziesiętną i odwrotnie. Zna wielokrotności bajtu jako jednostki informacji. Zna charakterystyczne liczby związane z bajtem i jego wielokrotnością, np. liczbę możliwych do zapisania liczb za pomocą 8 bitów.	Zapisuje liczby dziesiętne w postaci binarnej w Naturalnym Kodzie Binarnym NKB. Konwertuje liczniz kodu NKB do dziesiętnego i odwrotnie. Używa kalkulatora dla informatyków, np. z systemu Windows, do konwersji liczb.
Wie, jak zapisywać liczby binarne mniejsze od 0	Zapisuje dowolne liczby całkowite mniejsze od zera w kodzie U2. Zamienia liczby dziesiętne na postać U2.
Zna podstawowe prawa arytmetyki binarnej	Umie dodawać, odejmować, mnożyć liczby w systemie dwójkowym NKB i U2.
Zna podstawowe prawa algebry Boole`a	Wykorzystuje prawa algebry Boole`a do wykonywania podstawowych działań, takich jak iloczyn i suma logiczna.
Wie, jakie znaczenie i zastosowanie w informatyce ma kod szesnastkowy, np. w zapisie adresów IP, MAC adresu, zawartości komórek pamięci itp. Zna zależności pomiędzy liczbami w kodzie szesnastkowym i binarnym.	Zapisuje liczby w kodzie szesnastkowym i konwertuje je do postaci dziesiętnej i binarnej. Odczytuje za pomocą odpowiedniego programu lub polecenia systemowego, np. ipconfig, adresy sieciowe i przedstawia je w postaci dziesiętnej.
Wie, w jaki sposób zapisać ułamkowe liczby binarne. Rozróżnia liczby stało- i zmiennoprzecinkowe.	Zapisuje ułamkowe liczby binarne i uzasadnia niedokładności takiego zapisu w porównaniu z liczbami dziesiętnymi.

Zna sposób cyfrowego zapisu znaków alfanumerycznych za pomocą kodu ASCII w odmianie ISO 8859-2 i CP-1250	Umie znaleźć kod ASCII danego znaku w tabeli oraz z wykorzystaniem edytora tekstu.
Zna historię maszyn liczących i komputerów i wie, jaki wpływ na współczesną technikę wywarły pierwsze konstrukcje mechaniczne i elektroniczne. Zna kierunki rozwoju współczesnej informatyki, w tym transmisje strumieniowe, telefonię internetową, zwiększanie przepustowości łączy i dynamiczny rozwój chmur internetowych.	Korzysta z nowych usług chmur informatycznych, posługuje się różnymi rodzajami komputerów, np. smartfonem, tabletem, netbookiem, czytnikiem e-booków itp.
Wie, czym jest e-learning, i wie, jakie znaczenie ma w życiu współczesnego człowieka. Zna strukturę lekcji i kursu elektronicznego.	Umie skonfigurować platformę zdalnego nauczania, np. Moodle. Tworzy strukturę kursów i lekcji w Moodle. Umieszcza w lekcjach materiały edukacyjne, tworzy testy, quizy itp. Umie zorganizować krótkie szkolenie w ramach e-learningu.
Zna prawa człowieka dotyczące prywatności i wykorzystania wizerunku. Wie, w jakich dokumentach należy szukać przepisów prawnych dotyczących bezpieczeństwa w sieci i ochrony dóbr osobistych.	Umie ocenić, czy dana informacja, zdjęcie itp. narusza czyjeś prawo do prywatności lub ochrony wizerunku.
Zna zasady, dzięki którym nie narusza się praw osobistych innych osób	Stosuje zasady bezpiecznego publikowania informacji i wizerunku.
Zna nazwy różnych rodzajów przestępczości elektronicznej	Umie opisać poszczególne rodzaje przestępczości elektronicznej i wie, na co jest narażona osoba, wobec której przestępca je zastosował.
Wie, jak zachować się w obliczu przestępstwa elektronicznego. Rozumie, na czym polegają zagrożenia związane z korzystaniem z sieci Internet, w tym uzależnienie od gier sieciowych, informacji, kontaktów towarzyskich w portalach społecznościowych, zakupów elektronicznych itp.	Stara się zabezpieczać przed atakami przestępców elektronicznych, wykorzystując wiedzę o sposobach ich działania.
Wie, że nielegalne udostępnianie plików z zawartością chronioną prawem autorskim bez zezwolenia jest przestępstwem. Zdaje sobie sprawę z tego, że nielegalne pobieranie plików chronionych prawem majątkowym i autorskim jest kradzieżą.	Umie legalnie kupować utwory muzyczne i inne pliki chronione prawem autorskim. Umie korzystać z kina internetowego. Posługuje się systemem płatności elektronicznej.
Wie, w jakim charakterze może pracować wykształcony informatyk. Wie, jakie wymagania stawiają przed informatykiem różne stanowiska pracy.	Umie opisać wymagania stawiane przed informatykiem pracującym na różnych stanowiskach, w tym programisty, administratora sieci komputerowej i serwera, konserwatora

	sprzętu informatycznego itp.
Wie, jakimi cechami powinien charakteryzować się dobry informatyk	Umie ułożyć plan drogi do zdobycia zawodu informatyka.
Zna podstawowe rodzaje języków programowania i obszary ich zastosowania	Posługuje się edytorem środowiska programistycznego Free Pascal.
Wie, czym jest kompilator, interpreter oraz czym różni się edytor środowiska programistycznego od edytora tekstu	Korzysta z opcji edytora środowiska Free Pascal.
Zna nazwy kilku najpopularniejszych kompilatorów języków programowania, w tym Turbo Pascal, C++, Borland Pascal, Free Pascal	Stosuje prawidłową strukturę programu i umie odpowiednio i czytelnie ją zapisywać.
Zna przeznaczenie i działanie poszczególnych opcji edytora Free Pascal	
Wie, jak kompilować program	

Podstawy programowania

Wiedza. Uczeń:	Umiejętności. Uczeń:
Zna strukturę programu ułożonego w języku Pascal	Stosuje identyfikatory
Wie, czym są słowa kluczowe, procedury i funkcje, zmienne, stałe	Umie ułożyć, zapisać i skompilować prosty program, np. wykonując proste działania arytmetyczne z wprowadzaniem danych i wyprowadzaniem wyniku na ekran komputera.
Zna typy identyfikatorów i ich właściwości	Umie odczytać komunikaty kompilatora dotyczące wyników kompilacji, w tym rodzaje błędów, oraz poprawić błędy.
Wie, jak deklarować stałe i zmienne	Używa w programie prawidłowo zadeklarowanych stałych i zmiennych.
Wie, czym jest wyrażenie i zna jego typy	Układa proste programy z zastosowaniem różnych typów operacji: na liczbach całkowitych, rzeczywistych, typie znakowym i logicznym.
Zna podstawowe operacje arytmetyczne wykonywane przez program	Umie ułożyć program obliczający wartość według złożonego wzoru matematycznego z zachowaniem kolejności działań.
Zna rodzaje instrukcji, w tym: proste, warunkowe, wyboru, powtarzania	Układa proste programy z zastosowaniem różnych rodzajów instrukcji, w tym warunkowych, prostych, wyboru i powtarzania. Zagnieżdża w sobie instrukcje warunkowe.
Wie, jak zbudowana jest tablica, jakie typy danych można w niej zapisać. Zna typy tablicowe, w tym złożone.	Układa proste programy z zastosowaniem tablic.
Rozumie znaczenie i strukturę łańcuchów. Zna działanie funkcji dotyczących łańcuchów.	Stosuje w programach łańcuchy, np. do wyświetlania komunikatów.
Wie, czym są procedury i potrafi określić ich znaczenie w strukturze programu. Wie, czym różnią się zmienne lokalne od globalnych i na czym polega przekazywanie przez zmienną i przez wartość. Zna pojęcie podprogramu.	Definiuje procedury i podprogramy oraz umie je ułożyć oraz wywołać w programie.
Zna definicję i przeznaczenie funkcji i wie, czym różni się ona od procedury. Wie, czym są funkcje predefiniowane i jak znaleźć opis ich działania.	Prawidłowo używa funkcji w tworzeniu podprogramów. Używa funkcji predefiniowanych.
Wie, czym są złożone struktury danych, w tym rekordy	Używa rekordów do grupowania danych różnego typu.
Wie, czym są operacje wyjścia-wejścia i kiedy się je stosuje. Rozumie pojęcie fizycznych zbiorów danych i definiuje właściwości i rodzaje plików. Zna różnice pomiędzy dostępem sekwencyjnym a swobodnym do elementów	Układa programy wykorzystujące zapis i odczyt danych w pliku, w tym tekstowym. Deklaruje zmienne plikowe. Odczytuje poszczególne elementy z pliku, np. sekwencje znaków w pliku tekstowym. Stosuje procedury predefiniowane do

pliku. Zna typy plików, w tym: tekstowe, zdefiniowane.	operacji na plikach, w tym: Reset, Close, Seek, Eof.
Wie, czym jest debugowanie (śledzenie) i symulacja działania programu	Analizuje treść programu i znajduje błędy powodujące jego wadliwe działanie.
Definiuje pojęcie rekurencji w programie komputerowym	Układa program zawierający rekurencję, np. do obliczenia silni.
Wie, na czym polega modularyzacja programu	Stosuje moduły w programach złożonych.

Podstawy algorytmiki

Wiedza. Uczeń:	Umiejętności. Uczeń:
Definiuje pojęcie algorytmu oraz poszczególnych jego elementów	Podaje przykłady algorytmów różnych znanych czynności.
Wie, czym dla algorytmu są dane wejściowe i wyjściowe, oraz definiuje związek pomiędzy nimi. Wie, czym jest specyfikacja algorytmu.	Zapisuje specyfikację algorytmu: formułuje problem, sporządza jego opis, definiuje dane wejściowe i wyjściowe.
Zna sposoby zapisu algorytmu, w tym: lista kroków, pseudocode, schemat blokowy	Opisuje kolejność czynności za pomocą różnych sposobów zapisu algorytmu.
Zna różne edytory schematów blokowych	Układa algorytmy za pomocą edytora np. Dia.
Wie, jakie są różnice pomiędzy algorytmem liniowym a rozgałęzionym	Edytuje i układa proste algorytmy rozgałęzione, używając edytora.
Zna pojęcie złożoności algorytmu, w tym złożoności pamięciowej i czasowej, pesymistycznej, oczekiwanej i optymistycznej	Analizuje algorytmy pod kątem złożoności.

Algorytmy działań na liczbach całkowitych

Wiedza. Uczeń:	Umiejętności. Uczeń:
Wie (powtórka z matematyki), czym są liczby pierwsze	Tworzy specyfikację problemu dotyczącą sprawdzania, czy dana liczba jest liczbą pierwszą. Układa algorytm programu testującego liczby i wykrywającego liczby pierwsze. Układa program realizujący ten algorytm.
Wie (powtórka z matematyki), czym są liczby doskonałe	Tworzy specyfikację problemu dotyczącą sprawdzania, czy dana liczba jest liczbą doskonałą. Układa algorytm programu testującego liczby i wykrywającego liczby doskonałe. Układa program realizujący ten algorytm.
Wie (powtórka z matematyki), czym jest największy wspólny dzielnik	Tworzy specyfikację problemu dotyczącą znajdowania największego wspólnego dzielnika metodą Euklidesa.

	Układa algorytm programu na podstawie metody Euklidesa. Układa program realizujący ten algorytm.
Definiuje ciąg Fibonacciego	Tworzy specyfikację problemu dotyczącą wyznaczenia n-tego elementu ciągu Fibonacciego. Układa algorytm programu do obliczania n-tego elementu ciągu Fibonacciego. Układa program realizujący ten algorytm.
Wie, na czym polega metoda zachłanna wydawania reszty	Tworzy specyfikację problemu dotyczącą wydawania reszty w możliwie najmniejszej ilości nominałów (metodą zachłanną). Układa algorytm programu do wydawania reszty. Układa program realizujący ten algorytm.
Wie, na czym polega rozkładanie liczby na czynniki pierwsze	Tworzy specyfikację problemu dotyczącą rozkładu liczby na czynniki pierwsze. Układa algorytm programu znajdującego czynniki pierwsze danej liczby. Układa program realizujący ten algorytm.

Algorytmy wyszukiwania i sortowania

Wiedza. Uczeń:	Umiejętności. Uczeń:
Wie, czym różni się wyszukiwanie liniowe od wyszukiwania liniowego z wartownikiem	Tworzy specyfikację problemu dotyczącą wyszukiwania liniowego i liniowego z wartownikiem. Układa algorytm programu według specyfikacji wyszukiwania liniowego i liniowego z wartownikiem. Układa program realizujący ten algorytm.
Wie, czym jest tymczasowe minimum i maksimum i jakie mają one znaczenie dla algorytmu sortowania. Wie, czym jest proces rekurencji i na czym polega metoda dziel i zwyciężaj.	Tworzy specyfikację problemu dotyczącą największego lub najmniejszego elementu w zbiorze liczb. Układa algorytm programu według specyfikacji największego lub najmniejszego elementu w zbiorze liczb. Układa algorytm optymalny znajdujący największą i najmniejszą liczbę w zbiorze metodą rekurencyjną dziel i zwyciężaj. Układa program realizujący ten algorytm.
Wie, na czym polega sortowanie bąbelkowe i jaka jest złożoność obliczeniowa tej metody	Tworzy specyfikację problemu dotyczącą metody bąbelkowej sortowania liczb. Układa algorytm programu według specyfikacji metody bąbelkowej. Układa program realizujący ten algorytm.

Zna metodę sortowania przez wybieranie	<p>Tworzy specyfikację problemu dotyczącą metody sortowania liczb przez wybieranie.</p> <p>Układa algorytm programu według specyfikacji metody sortowania przez wybieranie.</p> <p>Układa program realizujący ten algorytm.</p>
Zna metodę sortowania przez wstawianie	<p>Tworzy specyfikację problemu dotyczącą metody sortowania liczb przez wstawianie.</p> <p>Układa algorytm programu według specyfikacji metody sortowania przez wstawianie.</p> <p>Układa program realizujący ten algorytm.</p>
Zna metodę sortowania QuickSort	<p>Tworzy specyfikację problemu dotyczącą metody sortowania szybkiego QuickSort.</p> <p>Układa algorytm programu według specyfikacji metody sortowania szybkiego.</p> <p>Układa program realizujący ten algorytm.</p>
Zna metodę sortowania przez scalanie metodą dziel i zwyciężaj	<p>Tworzy specyfikację problemu dotyczącą metody sortowania przez scalanie.</p> <p>Układa algorytm programu według specyfikacji metody sortowania przez scalanie.</p> <p>Układa program realizujący ten algorytm.</p>
Zna metodę sortowania przez zliczanie. Wie, że w tej metodzie nie występuje porównywanie liczb i że jest to metoda stabilna.	<p>Tworzy specyfikację problemu dotyczącą metody sortowania przez zliczanie.</p> <p>Układa algorytm programu według specyfikacji metody sortowania przez zliczanie.</p> <p>Układa program realizujący ten algorytm.</p>
Zna metodę sortowania kubełkowego	<p>Tworzy specyfikację problemu dotyczącą metody sortowania kubełkowego.</p> <p>Układa algorytm programu według specyfikacji metody sortowania kubełkowego.</p> <p>Układa program realizujący ten algorytm.</p>
Wie, czym jest zbiór uporządkowany i jaki ma to wpływ na wyszukiwanie. Wie, że można w tym przypadku stosować metodę rekurencyjną (dziel i rządź), jak i iteracyjną.	<p>Tworzy specyfikację problemu dotyczącą metody wyszukiwania binarnego w tablicy uporządkowanej.</p> <p>Układa algorytm programu według specyfikacji metody wyszukiwania binarnego dla tablic uporządkowanych.</p> <p>Układa program realizujący ten algorytm.</p>

Algorytmy numeryczne

Wiedza. Uczeń:	Umiejętności. Uczeń:
Zna metodę Newtona-Raphsona do obliczania wartości pierwiastka kwadratowego z zastosowaniem elementarnych działań matematycznych	Tworzy specyfikację problemu dotyczącą metody obliczania wartości pierwiastka kwadratowego metodą Newtona-Raphsona. Układa algorytm programu według specyfikacji metody obliczania wartości pierwiastka kwadratowego metodą Newtona-Raphsona. Układa program realizujący ten algorytm.
Zna schemat Hornera do obliczania wartości wielomianu. Zna jego niewielką złożoność obliczeniową.	Tworzy specyfikację problemu dotyczącą schematu Hornera. Układa algorytm programu według specyfikacji schematu Hornera. Układa program realizujący ten algorytm.
Zna metody numeryczne przedstawiania liczb za pomocą różnych kodów liczbowych, w tym dziesiętnego, NKB i Hex	Tworzy specyfikacje problemu dotyczące zamiany postaci liczb. Układa algorytmy programów według specyfikacji zamiany postaci liczb dla różnych kodów liczbowych, w tym dziesiętnego, NKB i Hex. Układa program realizujący ten algorytm.
Zna schemat Hornera do szybkiego podnoszenia liczby do potęgi (od lewej do prawej)	Tworzy specyfikację problemu dotyczącego szybkiego podnoszenia liczby do potęgi za pomocą schematu Hornera. Układa algorytm programu według specyfikacji szybkiego podnoszenia liczby do potęgi za pomocą schematu Hornera. Układa program realizujący ten algorytm.
Wie, czym są miejsca zerowe funkcji, i wie, na czym polega metoda numeryczna ich wyznaczania, w tym metoda biselekcji – połowienia (oparta na twierdzeniu Bolzano-Cauchy`ego)	Tworzy specyfikację problemu dotyczącego znajdowania miejsca zerowego funkcji metodą biselekcji. Układa algorytm programu według specyfikacji znajdowania miejsca zerowego funkcji metodą biselekcji. Układa program realizujący ten algorytm.

Algorytmy operacji na tekstach

Wiedza. Uczeń:	Umiejętności. Uczeń:
Zna pojęcie palindromu i wie, jak sprawdzać, czy dany ciąg znaków go tworzy	Tworzy specyfikację problemu dotyczącego badania, czy ciąg znaków tworzy palindrom. Układa algorytm programu według specyfikacji badania, czy ciąg znaków tworzy palindrom.

	Układa program realizujący ten algorytm
Zna pojęcie anagramu i wie, jak sprawdzić, czy ciąg znaków go tworzy	Tworzy specyfikację problemu dotyczącego badania, czy ciąg znaków tworzy anagram. Układa algorytm programu według specyfikacji badania, czy ciąg znaków tworzy anagram. Układa program realizujący ten algorytm.
Wie, na czym polega porządkowanie leksykograficzne (alfabetyczne)	Tworzy specyfikację problemu dotyczącego porządkowania leksykograficznego metodą kubelkową. Układa algorytm programu według specyfikacji porządkowania leksykograficznego metodą kubelkową. Układa program realizujący ten algorytm.
Wie, na czym polegają metody naiwna i Boyera-Moore'a stosowane do wyszukiwania wzorca w tekście	Tworzy specyfikację problemu dotyczącego wyszukiwania wzorca w tekście metodami naiwna i Boyera-Moore'a. Układa algorytm programu według specyfikacji wyszukiwania wzorca w tekście metodami naiwna i Boyera-Moore'a. Układa program realizujący ten algorytm.
Wie, czym jest wyrażenie ONP i jakie ma zastosowanie przy obliczaniu wartości wyrażenia	Tworzy specyfikację problemu dotyczącego obliczania wartości wyrażenia NOP. Układa algorytm programu według specyfikacji obliczania wartości wyrażenia NOP. Układa program realizujący ten algorytm.

Algorytmy kompresji i szyfrowania

Wiedza. Uczeń:	Umiejętności. Uczeń:
Zna zasadę stosowania szyfru przestawieniowego, np. Cezara i wieloalfabetowego	Tworzy specyfikację problemu dotyczącego szyfrowania metodami przestawieniowymi, np. Cezara. Układa algorytm programu według specyfikacji szyfrowania metodami przestawieniowymi, np. Cezara. Układa program realizujący ten algorytm.
Wie, czym są kody znaków o zmiennej długości na podstawie kodu Morse'a	Tworzy specyfikację problemu dotyczącego szyfrowania informacji kodem Morse'a jako przykładem kodu dla zmiennych o różnych długościach. Układa algorytm programu według specyfikacji kodowania kodem Morse'a. Układa program realizujący ten algorytm.
Wie, czym są kody znaków o zmiennej długości	Tworzy specyfikację problemu dotyczącego

na podstawie kodu Huffmana i zna jego właściwości kompresowania	szyfrowania informacji kodem Huffmana. Układa algorytm programu według specyfikacji kodowania kodem Huffmana. Układa program realizujący ten algorytm.
Wie, czym jest klucz publiczny i prywatny (RSA) oraz gdzie znajdują one zastosowanie. Zna wzory obliczania kluczy.	Umie zbudować klucz publiczny na podstawie wzorów.

Algorytmy badające własności geometryczne

Wiedza. Uczeń:	Umiejętności. Uczeń:
Zna metody badania warunków trójkąta	Tworzy specyfikację problemu dotyczącą badania warunków trójkąta. Układa algorytm programu badającego warunki trójkąta. Układa program realizujący ten algorytm.
Wie, na czym polega numeryczne badanie położenia punktu względem prostej i numeryczna prezentacja odcinka na płaszczyźnie	Tworzy specyfikację problemu dotyczącą badania położenia punktu względem prostej na płaszczyźnie i przynależności punktu do odcinka. Układa algorytm programu do badania położenia punktu względem prostej na płaszczyźnie i przynależności punktu do odcinka. Układa program realizujący ten algorytm.
Zna podstawy matematyczne dotyczące odcinków i prostych na płaszczyźnie	Tworzy specyfikację problemu dotyczącą badania odcinków na płaszczyźnie, a w szczególności ich przecinania się. Układa algorytm programu na podstawie specyfikacji. Układa program realizujący ten algorytm.
Wie, kiedy punkt należy do wielokąta na płaszczyźnie i jakie właściwości wielokąta pozwalają na opracowanie odpowiedniego algorytmu	Tworzy specyfikację problemu dotyczącą badania przynależności punktu do wielokąta. Układa algorytm programu do badania przynależności punktu do wielokąta. Układa program realizujący ten algorytm.
Wie, czym jest dywan Sierpińskiego, i wie, jak można go wykreślić na płaszczyźnie za pomocą metody rekurencyjnej	Tworzy specyfikację problemu dotyczącą wykreślenia dywanu Sierpińskiego za pomocą metody rekurencyjnej. Układa algorytm programu do wykreślenia dywanu Sierpińskiego za pomocą metody rekurencyjnej. Układa program realizujący ten algorytm.
Wie, czym jest płatek Kocha i jak można go wykreślić	Tworzy specyfikację problemu dotyczącą wykreślenia płatka Kocha na płaszczyźnie. Układa algorytm programu wykreślenia płatka

	Kocha na płaszczyźnie. Układa program realizujący ten algorytm.
Rozumie pojęcie drzewa binarnego i wie, jak ono powstaje	Tworzy specyfikację problemu dotyczącą wykreślanie drzewa binarnego na płaszczyźnie. Układa algorytm programu realizującego wykreślanie drzewa binarnego na płaszczyźnie. Układa program realizujący ten algorytm.

Indywidualna i zespołowa realizacja projektu informatycznego

Wiedza. Uczeń:	Umiejętności. Uczeń:
Wie, jak wygląda cykl tworzenia oprogramowania komputerowego i zna funkcje członków przykładowego zespołu	Planuje pracę zespołu informatyków pod kątem tematu i wielkości projektu.
Wie, jak tworzy się specyfikację wymagań na podstawie problemu przedstawionego przez klienta	Opracowuje specyfikacje prostych projektów informatycznych.
Wie, czym jest dokument projektu, jakie ma on znaczenie dla sprawnej pracy zespołu i jakie zasady obowiązują przy jego tworzeniu	Opracowuje dokument prostego projektu informatycznego z uwzględnieniem poznanych zasad.
Wie, jakie ustalenia należy poczynić, by praca zespołu przebiegała sprawnie, w tym: ujednoczenie nazw, formatu komentarzy, testowania i sposobu zapisu programu	Opracowuje tabele opisujące jednolite warunki sposobu tworzenia projektu
Wie, jak wielkie znaczenie ma testowanie gotowego programu i jak powinno być ono zorganizowane	Testuje swoje programy zgodnie z poznanymi zasadami. Układa scenariusz testowania programu.
Wie, co powinna zawierać dokumentacja techniczna programu komputerowego	Opracowuje dokumentację prostego programu komputerowego i zapisuje ją w dokumencie tekstowym.
Zna zasady tworzenia dokumentacji użytkownika, w tym: instrukcji obsługi programu. Zna procedury przekazania oprogramowania klientowi.	Sporządza instrukcję obsługi programu z wykorzystaniem elementów graficznych, zrzutów ekranowych itp. Przeprowadza instalację programu i wstępny instruktaż obsługi.

Bazy danych

Wiedza. Uczeń:	Umiejętności. Uczeń:
Rozumie pojęcia: tabele, wiersze i klucze i wie, jakie role pełnią w bazie danych	Opisuje i przedstawia na przykładach znaczenie kluczy dla bazy danych.
Wie, czym różnią się pierwsza postać normalna tabeli od drugiej i dlaczego po przekształceniu powstaje więcej tabel. Wie, na czym polega redundancja danych w tabeli i jak ją eliminować.	Przekształca tabele z pierwszej postaci do drugiej, eliminuje redundancję w tabelach (przeprowadza proces normalizacji).
Wie, jak wygląda 3 i 4 postać normalna tabeli w bazie danych. Wie, jak eliminować dane nienależące do klucza.	Przekształca proste tabele do postaci 3 i 4 normalnej, eliminując zależności wielowarstwowe, niedotyczące klucza głównego.
Wie, na czym polegają i jakie znaczenie dla budowy bazy mają relacje między tabelami. Rozumie relacje 1..1 i 1..wielu. Zna pojęcie klucza obcego.	Określa relacje pomiędzy tabelami w zależności o wymagań stawianych bazie.
Zna kilka programów, w których występują systemy baz danych pozwalające na ich tworzenie za pomocą kreatorów, np. OpenOffice.org Base, LibreOffice Base, MS Access	Posługuje się menu i kreatorem systemu do tworzenia relacyjnych baz danych, np. OpenOffice.org Base.
Zna pojęcia: formularz, kwerenda, raport	Tworzy podstawową strukturę bazy danych za pomocą kreatora, korzystając z własnego projektu prostej bazy.
Wie, jak utworzyć bazę danych z pominięciem kreatora	Wykorzystuje system np. OpenOffice.org Base do tworzenia relacyjnych baz danych z pominięciem kreatora. Modyfikuje tabele i inne elementy bazy danych.
Wie, czym są indeks i relacje i jakie mają znaczenie dla funkcjonowania bazy danych	Indeksuje tabele bazy danych i tworzy relacje pomiędzy nimi.
Wie, jak wprowadzać dane do bazy danych poprzez arkusz danych lub formularz	Opracowuje formularze do wprowadzania danych do bazy. Wprowadza dane do bazy danych.
Wie, jak dobierać kryteria podczas wyszukiwania danych w bazie	Korzysta z opcji wyszukiwania danych w systemie bazy danych, np. OpenOffice.org Base.
Wie, jak dobrać parametry sortowania i filtrowania danych w bazie	Sortuje i filtruje dane w tabelach bazy danych.
Wie, jak wykorzystać kreatora do zbudowania kwerendy dla bazy danych. Zna opcje pozwalające na budowanie kwerend bez użycia kreatora.	Buduje kwerendy dla bazy danych w systemie bazodanowym, np. OpenOffice Base, przy pomocy kreatora i z jego pominięciem. Odczytuje przy ich pomocy dane z bazy.
Wie, jak drukować dane z bazy danych	Kreuje formularze wydruku dla systemu bazy danych, w tym także z zastosowanie styli.

Wie, czym jest SQL i gdzie znajduje zastosowanie

Wykorzystuje opcje systemu bazy danych, np. OpenOffice.org Base, do opracowania kwerendy w języku SQL.

3. Treści edukacyjne

Systemy operacyjne. Sieci komputerowe.

1. Charakterystyka różnych systemów operacyjnych, w tym Windows, Linux, Mac OS i Android (1.2, 7.1)
2. Obsługa programów diagnostycznych i narzędziowych służących do sprawdzania i zmiany zasobów systemowych, budowy komputera i diagnostyki (1.2, 1.4, 5.25)
3. Ochrona antywirusowa i upgrade systemu operacyjnego (1.2, 5.25)
4. Wirtualna maszyna (1.4)
5. Narzędzia dotyczące higieny dysku komputera (5.25)
6. Warstwowy model sieci komputerowej, w tym TCP/IP(1.3)
7. Określenie i modyfikacja ustawień sieciowych związanych z jego pracą w lokalnej sieci komputerowej i Internecie (1.3)
8. Identyfikacja komputera w sieci (1.3)
9. Wykorzystanie programów diagnostycznych do monitorowania pracy sieci i jej ustawień (1.3, 1.4, 5.25)
10. Protokoły sieciowe, w tym do transmisji bezprzewodowej, w sieci lokalnej i programy do ich monitorowania (1.3, 1.4)
11. Administrowanie siecią klient-serwer na przykładzie Windows Serwer(udostępnianie zasobów, protokół http) (1.3, 1.4)
12. Rozwój nowoczesnych środków przesyłania informacji za pomocą sieci (RSS, transmisje strumieniowe) (1.3, 3.1)
13. Wykorzystanie chmury informatycznej do pracy w zespole, przechowywania i udostępniania zasobów oraz komunikacji (3.1, 3.2)
14. Wyszukiwanie informacji z różnych dziedzin, w tym przedmiotów szkolnych (2.4, 6.2)
15. Posługiwanie się terminologią sieciową (1.3, 3.1, 6.1, 7.3)

Grafika komputerowa i wizualizacja danych

1. Podstawowe modele barw, przeznaczenie i ich zastosowanie (4.1)
2. Edytory grafiki rastrowej i wektorowej, w tym darmowe i on-line w chmurze informatycznej (4.2, 4.3, 1.1)
3. Właściwości grafiki rastrowej i wektorowej, podstawowe różnice i obszary zastosowań (4.2,
4. Tworzenie i edycja grafiki wektorowej i rastrowej (w tym fotografii) za pomocą edytorów (1.1, 4.2, 4.3, 5.1, 5.2)
5. Warstwy, filtry i inne przekształcenia obrazów komputerowych w edytorach grafiki (1.1, 4.2, 4.3)
6. Zmiana właściwości obrazów komputerowych, w tym rozdzielczości, modeli barw i rozmiaru (1.1, 4.2, 4.3)
7. Zasady dobrej fotografii i jej edycja (1.1, 4.3)
8. Pliki graficzne – formaty i przeznaczenie oraz praktyczne wykorzystanie ich właściwości (4.2, 4.3)
9. Prosta edycja wideo i jej wykorzystanie w prezentacjach itp. (1.1, 4.3)
10. Obrazowanie informacji i danych za pomocą wykresów arkusza kalkulacyjnego (4.4)

Kody binarne. Przyszłość informatyki.

1. Istota stosowania kodów liczbowych w praktyce informatyka (1.1)
2. Kody binarne jako podstawa funkcjonowania komputera i sieci informatycznych (1.1, 5.11a)
3. Kod ASCII i jego znaczenie w informatyce (1.1, 5.11a)
4. System szesnastkowy w praktyce informatyka (1.1, 5.11a)
5. Historia komputerów i technik komunikacyjnych (7.1)
6. Zastosowanie chmur informatycznych i ich wpływ na rozwój i zmiany różnych dziedzin życia (1.4, 7.1, 7.2)
7. E-learning jako czynnik wpływający na rozwój cywilizacyjny (1.4, 7.1, 7.2, 7.5)
8. Wpływ rozwoju elektronicznych środków komunikacji, przechowywania i upowszechniania informacji na rozwój cywilizacyjny (7.1, 7.2, 7.5)

Bezpieczeństwo informacji i przestępczość elektroniczna

1. Szyfrowanie danych z zastosowaniem programów szyfrujących, w tym szyfrujących e-mail i dokumenty tekstowe (1.3, 2.5, 5.25)
2. Szyfrowanie z zastosowaniem kluczy (2.5, 5.25)
3. Zastosowanie certyfikatów i elektronicznego podpisu (2.5, 5.11e)
4. Programy antywirusowe, w tym zapora ogniowa (2.5, 5.25)
5. Zagadnienia przestępczości elektronicznej, w tym kradzież własności intelektualnej i prawo do ochrony wizerunku (7.3, 7.4)

Gromadzenie informacji i danych

1. Tabele, relacje, indeksy, formularze, kwerendy i inne zagadnienia związane z komputerowymi bazami danych (2)
2. Projektowanie i realizacja relacyjnej bazy danych, zapewnienie integralności danych i prostych metod zapisu danych w bazie (2.1, 5.1, 5.2)
3. Metody wyszukiwania i przetwarzania informacji w relacyjnej bazie danych (2.2)
4. Wykorzystanie systemów relacyjnych baz danych, w tym niekomercyjnych
5. Proste aplikacje bazodanowe z zastosowaniem języka AQL (2.3, 5.1, 5.2)

Rozwiązywanie problemów i podejmowanie decyzji za pomocą komputera – programowanie w języku Pascal

1. Zasady programowania modularnego (strukturalnego) w programowaniu (5.15)
2. Podstawowe konstrukcje programistyczne w języku Pascal: instrukcje iteracyjne i warunkowe, rekurencje, funkcje i procedury, instrukcje wejścia-wyjścia (5.23)
3. Struktura kodu źródłowego programu komputerowego (5.23)
4. Układanie programu prowadzącego do rozwiązania problemu (5.23, 5.28, 6.1)

5. Działania na plikach o różnej zawartości (5.23, 5.28, 6.1)
6. Ocena poprawności działania ułożonego programu komputerowego i jego testowanie (5.25, 5.26)
7. Pojęcie błędu względnego i bezwzględnego oraz wskazanie źródła błędów (5.27)
8. Praca zespołowa nad projektem informatycznym (5.28, 6.1)
9. Tworzenie dokumentacji programu komputerowego (5.28)

Algorytmiczne rozwiązywanie problemów

1. Analiza i modelowanie problemów informatycznych i sytuacji problemowych z różnych dziedzin (5.1)
2. Stosowanie podejścia algorytmicznego do rozwiązywania problemów z różnych dziedzin (5.2)
3. Określanie i formułowanie sytuacji problemowych, których rozwiązanie wymaga użycia komputera (5.2, 5.3)
4. Podstawowe techniki algorytmiczne (5.1, 5.2, 5.5)
5. Dobór i układanie efektywnych algorytmów do rozwiązania problemów (5.1, 5.2, 5.4, 5.7)
6. Zapis algorytmu w różnych notacjach (5.1, 5.2, 5.4)
7. Ocena zgodności algorytmu z jego specyfikacją, efektywność oraz oszacowanie jego mocy obliczeniowej (5.1, 5.6, 5.7)
8. Metoda „dziel i zwyciężaj” w zastosowaniu algorytmicznym (algorytm i realizacja programowa) (5.8)
9. Rekurencja w prostych sytuacjach problemowych (algorytm i realizacja programowa) (5.9)
10. Metoda zachłanna w rozwiązywaniu problemów informatycznych (algorytm i realizacja programowa – wydawanie reszty) (5.10)
11. Konwersje liczb do różnych postaci (algorytm i realizacja programowa) (5.11a, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
12. Badanie i wykrywanie liczb – pierwszych, doskonałych (algorytm i realizacja programowa) (5.11a, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
13. Rozkład liczby na czynniki pierwsze (algorytm i realizacja programowa) (5.11a, 5.12, 5.13, 5.14, 5.15, 5.21, 5.22, 5.23, 5.24)
14. Iteracyjna i rekurencyjna realizacja algorytmu Euklidesa (algorytm i realizacja programowa) (5.11a, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
15. Obliczanie wartości elementu ciągu Fibonacciego metodami iteracyjną i rekurencyjną (algorytm i realizacja programowa) (5.11a, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
16. Sortowanie liczb z jednoczesnym znajdowaniem elementu najmniejszego i największego z zastosowaniem algorytmu naiwnego i optymalnego (algorytm i realizacja programowa) (5.11b, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
17. Sortowanie metodami: bąbelkową, przez wybór, przez wstawianie, przez scalanie, szybką i kubełkową (algorytm i realizacja programowa) (5.11b, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
18. Obliczanie wartości pierwiastka kwadratowego metodą Newtona-Raphsona (algorytm i realizacja programowa) (5.11c, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)
19. Obliczanie wartości wielomianu wg schematu Hornera (algorytm i realizacja programowa) (5.11c, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5.24)

20. Podnoszenie liczb do potęgi metodą Hornera (algorytm i realizacja programowa) (5.11c, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
21. Wyznaczanie miejsc zerowych funkcji metodą połowienia (algorytm i realizacja programowa) (5.11c, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
22. Obliczanie pola powierzchni obszarów zamkniętych (algorytm i realizacja programowa) (5.11c, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
23. Wyznaczanie miejsc zerowych funkcji (algorytm i realizacja programowa) (5.11, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
24. Wykrywanie palindromów i anagramów w ciągu znaków (algorytm i realizacja programowa) (5.11d, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
25. Porządkowanie alfabetyczne ciągu znaków (algorytm i realizacja programowa) (5.11d, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
26. Wyszukiwanie wzorca w tekście metodą naiwną i Boyera-Moore'a (algorytm i realizacja programowa) (5.11d, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
27. Obliczanie wartości wyrażenia w postaci ONP (algorytm i realizacja programowa) (5.11d, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
28. Kody znaków o zmiennej długości – Huffmana i Morse'a (algorytm i realizacja programowa) (5.11e, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
29. Algorytmy szyfrowania – Cezara, przestawieniowy (algorytm i realizacja programowa) (5.11e, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
30. Algorytm szyfrowania z kluczem jawnym (5.11e, 5.12, 5.13, 5.14, 5.15, 5.21, 5.22, 5.23, 5,24)
31. Sprawdzanie warunku trójkąta (algorytm i realizacja programowa) (5.11f, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
32. Badanie położenia punktu względem prostej (algorytm i realizacja programowa) (5.11f, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
33. Badanie przynależności punktu do odcinka (algorytm i realizacja programowa) (5.11f, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
34. Wykrywanie zjawiska przecinania się odcinków na płaszczyźnie (algorytm i realizacja programowa) (5.11f, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
35. Wykrywanie przynależności punktu do obszaru na płaszczyźnie (algorytm i realizacja programowa) (5.11f, 5.12, 5.13, 5.14, 5.15,5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
36. Realizacja algorytmiczna i programowa konstrukcji rekurencyjnych – drzewo binarne, płatek Kocha, dywan Sierpińskiego (algorytm i realizacja programowa) (5.9, 5.11f, 5.12, 5.13, 5.14, 5.15, 5.16, 5.17, 5.21, 5.22, 5.23, 5,24)
37. Obliczanie mocy obliczeniowej algorytmu (5.18, 5.19, 5.20)

4. Sposoby osiągnięcia celów kształcenia i wychowania

Założenia ogólne

Nauczanie informatyki w IV etapie edukacyjnym jest ukierunkowane na praktyczne wykorzystanie wiedzy i umiejętności zdobytych we wcześniejszych etapach i na poziomie podstawowym oraz znaczne poszerzenie zagadnień. Dużą rolę odgrywa dobór podręcznika, w którym obok informacji znajduje się bogaty opis ćwiczeń. Część z nich powinna służyć doświadczalnemu sprawdzeniu podanych informacji oraz prowadzić do wyciągnięcia własnych wniosków i samodzielnego zdobywania wiedzy i umiejętności. Ćwiczenia, w ramach których uczeń wykonuje projekty, powinny dotyczyć zagadnień, z którymi spotka się w przyszłej pracy lub w czasie studiów. Zadania wykonywane w czasie zajęć i w ramach prac domowych powinny zakładać pracę w zespołach, z wykorzystaniem technologii komunikacyjnych, np. komunikatorów oraz chmur informatycznych.

W procesie nauczania należy także wykorzystać umiejętności i wiedzę uczniów zdobyte samodzielnie. Są one niejednokrotnie znacznie wyższe od nabytych w szkole. W związku z tym może wystąpić znaczna różnica w poziomach wiedzy i umiejętności pomiędzy uczniami. Dlatego bardzo ważnym elementem jest indywidualizacja zadań przewidzianych dla poszczególnych uczniów. Może ona polegać np. na poszerzeniu zakresu korzystania z danego programu lub usługi, skrócenia czasu wykonania ćwiczenia itp. Indywidualizacja nie powinna dotyczyć jedynie uczniów bardzo dobrych. Zadania i wymagania należy również dostosować do poziomu uczniów najsłabszych. Warunkiem jest jednak takie ich formułowanie, by umożliwić szybkie zniwelowanie różnic w stosunku do uczniów dobrych i zapewnić realizację podstawy programowej.

Warunki realizacji programu

Wyposażenie sali i sprzęt dla uczniów ma niebagatelne znaczenie dla przebiegu nauczania informatyki. Jest to również ważne w kontekście wykorzystania tych środków przez nauczyciela. Trudno bowiem uczyć np. o zastosowaniach i konstruowaniu prezentacji, wykorzystując wyłącznie tablicę. Elektroniczne materiały dydaktyczne, sprzęt oraz sposób, w jaki korzysta z niego, powinny być przykładem dla uczniów.

W pracowni informatycznej nieodzowne jest połączenie z Internetem zapewniające komfortową pracę przy wszystkich stanowiskach. Jest to szczególnie wskazane w czasie wykonywania ćwiczeń z wykorzystaniem chmur informatycznych.

Ćwiczenia można wykonać niezależnie od systemu operacyjnego. Należy jednak pamiętać, by odpowiednio sformułować zadania lub zmienić istniejące w podręczniku oraz dobrać dostępne dla danego systemu programy zgodne funkcjonalnie z proponowanymi w programie. W przypadku pracy z chmurami ważne jest wyposażenie systemu w odpowiednią przeglądarkę.

Zabezpieczenie antywirusowe w pracowni może polegać nie tylko na stosowaniu odpowiedniego oprogramowania. Dobrym rozwiązaniem jest wyposażenie komputerów w RecoveryCard. Karta tego typu powoduje, że wszelkie zmiany w systemie, w tym także wywołane przez wirusy, znikają po kolejnym uruchomieniu komputera. System zawsze powraca do stanu, w którym została uruchomiona karta.

Rzutnik może zostać zastąpiony dużym ekranem plazmowym lub LCD np. o przekątnej 50 cali. Jest to rozwiązanie wygodniejsze i ekonomicznie uzasadnione. Koszt jest podobny. Ekran ma jednak wiele zalet. Nie wymaga konserwacji i materiałów eksploatacyjnych, takich jak lampa. Może współpracować z dowolnymi urządzeniami wideo i nie wymaga zaciemnienia sali. Pracuje z najwyższą rozdzielczością, obraz posiada duży kontrast i jasność. Możliwe jest też zastosowanie urządzenia zamieniającego ekran w tablicę interaktywną, np. eBeam.

Oprogramowanie

Program może być zrealizowany przy użyciu programów komputerowych pozwalających na wykonie ćwiczeń w odpowiednim zakresie. Powinny one dawać możliwość indywidualizacji zadań, zarówno dla uczniów bardzo, jak i mniej zdolnych. Jeśli tylko to możliwe należy korzystać ze sprawdzonych lub proponowanych przez autorów podręczników darmowych programów komputerowych, by umożliwić uczniom należyte przygotowanie się do zajęć i wykonywania zadań domowych.

Proponowane programy komputerowe

Do realizacji większości tematów i dla zapewnienia komfortu dostępu do informacji, korzystania z platform, poczty itp. należy zapewnić sprawnie działające przeglądarki internetowe IE, Firefox, Chrome itp.

Poniższa tabela podaje przykłady programów pozwalających na pełną realizację programu nauczania.

Nazwa programu	Funkcje
SIW – System Information for Windows (www.gtopala.com)	Program diagnostyczny dla Windows
IMG BURN	Program do zapisu płyt z wykorzystaniem obrazów ISO
VirtualBox	Wirtualna maszyna
Systemowy monitor zasobów	Program systemu Windows do sprawdzania i wyświetlania informacji o zasobach systemu
Eyeos.org	System operacyjny chmury internetowej dostępny wyłącznie on-line
Linux Ubuntu	System operacyjny, który należy pobrać w najnowszej wersji
Przywracanie systemu Windows	Systemowy program do przywracania systemu i tworzenia punktów przywracania

CCleaner	Program do konserwacji systemu Windows
VisualRoute Lite Edition	Program do śledzenia połączeń w sieci
inSSIDer	Aplikacja do analizowania ruchu w sieci bezprzewodowej
Konsola MMC	Narzędzie administracyjne do ustalania parametrów zabezpieczeń hasłami sieci klient-serwer
Paseczek	Program do zarządzania kanałami RSS
Easy Graphic Converter	Program do konwersji i zmiany parametrów grafiki komputerowej
Picasa	Przeglądarka do zdjęć i grafiki z opcjami edycji
InkScape	Edytor grafiki wektorowej
GIMP	Edytor grafiki rastrowej
Pixlr.com	Edytor grafiki rastrowej i fotografii z chmury informatycznej
SketchUp	Edytor grafiki 3D
OpenOffice.org Draw lub LibreOffice Draw	Edytory grafiki pakietów biurowych
CadStd Lite	Edytor wspomagający projektowanie typu CAD
Any Video Converter	Program do zmiany parametrów materiałów i plików wideo
Free Pascal	Edytor środowiska programistycznego

Organizacja pracy w zespołach

Do większości lekcji można użyć prezentacji przygotowanych w różnych formatach, np. pptx, video. Niektóre z nich mogą wykonać uczniowie w ramach zadania domowego. Dzięki takiej formie możliwa jest indywidualizacja zadań. Może to dotyczyć nie tylko najzdolniejszych, ale także słabszych uczniów. Mobilizuje to do szukania ciekawych rozwiązań napotkanych problemów. W celu wspomagania ich pracy nauczyciel może utworzyć specjalne konto w komunikatorze i umówić się na konsultacje elektroniczne. Dobrym sposobem jest też wykorzystanie chmury informatycznej i zlecenie takiego zadania zespołom. Dzięki współdzieleniu dokumentów np. w chmurze skydrive.com lub

google uczniowie będą mogli w praktyce sprawdzić ideę e-pracy i współdziałanie w ramach zespołu. Niektóre ćwiczenia wykonywane na lekcji także mogą być realizowane w zespołach. Przykładem może być wspólne tworzenie strony internetowej i redagowanie informacji.

Wykorzystanie platformy e-learningowej w celu zoptymalizowania czasu przeznaczanego na ćwiczenia

Jeśli w szkole nie działa platforma zdalnego nauczania, np. Moodle, nauczyciel informatyki może ją zainstalować lub skorzystać z darmowej instalacji na moodle4free.com. Można także korzystać z komercyjnych platform, np. Fronter. Dzięki temu możliwe będzie znaczne przyspieszenie różnych czynności, np. udostępnienia plików do ćwiczeń. Przy niewielkim czasie przeznaczonym na zajęcia z informatyki optymalizacja czasu jest niezmiernie ważna.

Przykład.

Lekcja dotyczy retuszu fotografii prowadzonej w pracowni bez serwera plików i kont uczniowskich i 16-osobowej grupy uczniów korzystającej z indywidualnych stanowisk.

Bez użycia platformy	Czas	Z użyciem platformy	Czas
Zdjęcia do ćwiczeń należy umieścić na komputerach uczniów, rozsyłać w sieci lokalnej lub przekazywać na nośnikach. Dla każdej grupy lub klasy osobno.	Od 1 do 3 minut	Uczniowie pobierają pliki do ćwiczeń z platformy. Każda klasa te same z raz przygotowanego zestawu.	Ok. 1 minuty z zalogowaniem
Dokładny opis ćwiczenia należy podyktować lub pokazać w postaci slajdu na rzutniku	Od 1 do 3 minut	Dokładny opis ćwiczenia znajduje się na platformie. Uczeń czyta z ekranu w swoim tempie. Opis może być bardzo dokładny.	Dostęp do opisu to kilka sekund
Ocena pracy musi być indywidualna i uzasadniona	Od 8 do 15 minut	Wszyscy uczniowie przesyłają prace na swoje konto tuż przed ukończeniem pracy lub po jej dokończeniu w domu. Nauczyciel ocenia w dowolnie wybranym czasie. Oceny pojawiają się na koncie uczniów. Łatwo je przepisać do dziennika. Pozostają w archiwum.	Czas kilkanaście sekund na przesłanie. Ocena w dowolnym czasie poza lekcją lub w czasie trwania następnego ćwiczenia.
Omówienie i podyktowania zadania domowego	Od 1 do 3 minut. Sprawdzenie wrywkowe ok. 3 minut.	Zadanie gotowe na platformie i wspólne dla wielu grup, łącznie z miejscem na jego przesłanie. Uczeń czyta je w domu. Uczniowie przesyłają zadania bezpośrednio na platformę.	Kilka sekund na podanie informacji o miejscu umieszczenia opisu zadania

		Można je sprawdzić w dowolnym momencie.	
Suma	Minimum 11 minut. Przy 45-minutowej lekcji i czynnościach typu sprawdzenie obecności i wpisanie tematu daje co najwyżej ok. 30 minut na ćwiczenie.	Suma	Ok. 3 minut, w tym logowanie i pobieranie.

Zysk w postaci zaoszczędzonego czasu można poświęcić na bezpośrednią pracę przy komputerze.

Dzięki zastosowaniu platformy e-learningowej możliwe jest skuteczne indywidualizowanie pracy z uczniami o różnych potrzebach edukacyjnych. Można dla nich przygotować zadania odpowiadające ich możliwościom lub aspiracjom bez konieczności zabierania dodatkowego czasu na lekcji. Podobnie rzecz ma się z ocenianiem prac. Można poświęcić więcej czasu na omówienie metod i sposobów rozwiązania zaproponowanych przez ucznia. Platforma daje bowiem możliwość recenzowania przesłanych prac. Na platformie można także prowadzić konsultacje. Wszystkie prace pozostają na kontach uczniów do końca roku, dzięki czemu można analizować postępy w nauce poszczególnych uczniów, a w szczególności, tych ze specjalnymi wymaganiami edukacyjnymi.

Dzięki platformie możliwe jest także lepsze przygotowanie się uczniów do zajęć. Nauczyciel może umieścić na niej materiały i zadania potrzebne do przeprowadzenia następnej lekcji. Uczniowie ambitni i zainteresowani przedmiotem mogą więc wykonać część pracy w domu, np. przygotować potrzebne elementy.

Stosowanie platformy e-learningowej nie jest obowiązkowe i niezbędne do zrealizowania programu nauczania. Jest jednak wielkim ułatwieniem dla nauczyciela dysponującego niewielką ilością godzin przeznaczonych na nauczanie przedmiotu.

Indywidualizacja pracy z uczniem ze specjalnymi potrzebami edukacyjnymi

O wielu aspektach pracy z uczniami wymagającymi dostosowania metod nauczania do ich stanu opisanego w opinii poradni pisano już w niniejszym programie. Warto jednak podsumować to ważne zagadnienie w kontekście realizacji programu.

Bez względu na to, jakie metody pracy z uczniem nauczyciel stosuje, według rozporządzenia należy dostosować metody nauczania i sposób realizacji programu dla uczniów z następującymi specjalnymi potrzebami edukacyjnymi:

- Dysleksja
- Dysgrafia i dysortografia
- Dyskalkulia
- Afazja
- Zaburzenia przestrzenne
- Zespół Aspergera
- Zaburzenia zachowania

Podstawą do stosowania odrębnych metod jest przedstawiona przez ucznia lub jego opiekunów opinia specjalisty. Zespół uczący w danej klasie musi opracować kartę indywidualnych potrzeb ucznia dla każdego z podopiecznych z SPE. W większości przypadków dużą rolę będą odgrywały zajęcia z informatyki. Jest to zrozumiałe, ponieważ są to lekcje, na których uczniowie pracują indywidualnie lub w małych zespołach. A to sprzyja indywidualizacji zadań i zmianie niektórych wymagań bez wpływu na pracę pozostałych uczniów. Trudno podać jednoznacznie określone sposoby pracy z danym uczniem. Muszą one wynikać ze wspólnych ustaleń zespołu uczącego. Podstawowym celem jest zrealizowanie podstawy programowej. Zmienione mogą być jedynie sposoby i terminarz osiągnięcia tego celu. Oznacza to, że plan pracy z danym uczniem dotyczy osiągnięcia głównego celu w ciągu całego roku szkolnego.

Praktyczne porady w odniesieniu do realizacji programu z uczniami ze specjalnymi potrzebami edukacyjnymi

Uczniowie z SPE mogą mieć problemy z zapisywaniem lub zapamiętywaniem haseł, nazw kont itp. Nauczyciel powinien je znać, by można było korzystać np. z platformy e-learningowej, chmury informatycznej itp. Oczywiście jest, że dotyczy to jedynie kont wykorzystywanych na lekcjach, a nie kont osobistych ucznia.

Edytory tekstu i środowiska programistycznego na stanowiskach danych uczniów powinny mieć wyłączoną funkcję autouzupełniania lub automatycznej korekty błędów. Konieczne jednak musi być aktywna opcja automatycznego zaznaczania błędów w edytorze tekstu. Uczeń będzie widział swoje błędy i powinien je poprawić.

Zadania dotyczące edycji tekstu, grafiki, prezentacji, wyszukiwania informacji powinny być uzgodnione z nauczycielami innych przedmiotów i mogą być istotnym elementem wspomagającym ich nauczanie przy jednoczesnym zapewnieniu realizacji podstawy programowej informatyki. Należy je uwzględnić w indywidualnej karcie pracy.

Uczniowie ze specjalnymi potrzebami edukacyjnymi często posiadają duże zdolności w rozmaitych dziedzinach, np. artystyczne. Szybkie ich odkrycie pozwoli odpowiednio opracować zadania dotyczące grafiki, wideo itp. Będzie to miało duży wpływ na poczucie własnej wartości i zdobycie uznania w grupie rówieśniczej.

Problemy uczniów ze zdiagnozowaną dyskalkulią mogą spowodować trudności z używaniem arkusza kalkulacyjnego i opanowaniem metod numerycznych.

Niektórzy uczniowie, np. z zespołem Aspergera, wykazują ponadprzeciętne zdolności do pracy z komputerem. Znamienne jest bowiem, że zazwyczaj są bardzo precyzyjni i działają zgodnie z logiką. Niektóre poradnie zalecają im naukę informatyki jako drogę do zdobycia odpowiedniego dla nich zawodu. Dzięki tej cesze i dobrej pamięci są w stanie wykonać ćwiczenia znacznie szybciej od pozostałych uczniów. Podanie im dodatkowych zadań odpowiednio premiowanych oceną będzie dobrym sposobem na utrzymanie ich skupienia na pracy. Należy także inaczej traktować ich zachowanie, np. jednoczesną pracę nad zadaniem i czytanie artykułów na portalach czy czytanie książki w czasie, gdy nauczyciel omawia zadanie. Nie są to działania skierowane przeciw dyscyplinie na lekcji lub nauczycielowi. Najczęściej uczniowie ci są w stanie jednocześnie czytać i słuchać tego, co przekazuje prowadzący zajęcia.

Nie tylko wymagania, czas realizacji materiału, ale także zachowanie na zajęciach powinno być troską nauczyciela informatyki. W klasycznej sali lekcyjnej łatwiej jest zapanować na uczniami. W pracowni obowiązuje regulamin i konieczna jest dbałość o sprzęt. Uczniowie z zaburzeniami zachowania, np. ADHD, mogą mieć problem z utrzymaniem dyscypliny, dlatego dobrze jest, gdy

siedzą blisko nauczyciela, a przed każdymi zajęciami należy powtarzać ustalone wcześniej zasady zachowania.

Duży wpływ na pracę z uczniami z SPE może mieć uczestnictwo w odpowiednio dobranym zespole. Konieczność współpracy i podjęcia odpowiedzialności za wynik zespołu i udział w jego sukcesie może zmobilizować do większej dbałości o jakość wykonanej pracy.

W żadnym przypadku nauczyciel nie może przyczynić się do powstania negatywnych opinii o uczniu wśród rówieśników. Nie należy wydzielać oddzielnych stanowisk, tworzyć barier, komentować ich zachowań. Pomocna będzie platforma e-learningowa, z której każdy z uczniów pobiera zadania i w której otrzymuje materiały i miejsce na przesłanie rozwiązania. Uniknie się wtedy porównywania ocen i zadań przez innych uczniów.

Wspólna praca całego zespołu uczącego może spowodować, że uczniowie, którzy nie mogli odnaleźć się w tradycyjnym, polskim systemie oświaty, zyskają szanse zdobycia wykształcenia i zdania egzaminu maturalnego. Często bowiem to nie ograniczenia intelektualne ucznia wpływają na jego kłopoty szkolne, a niemożność przystosowania się do sposobów realizacji zajęć i sprawdzania wiedzy.

5. Opis założonych osiągnięć ucznia

Głównym założeniem jest realizacja podstawy programowej przygotowanie ucznia do zdania egzaminu maturalnego.

Po ukończeniu nauki przedmiotu informatyka na poziomie rozszerzonym uczeń:

- umie samodzielnie rozwiązywać zadania maturalne teoretyczne i praktyczne
- osiąga wynik potwierdzający predyspozycje do dalszego kształcenia w zawodach informatycznych.
- sprawnie posługuje się zaawansowanymi opcjami programów komputerowych z różnych dziedzin oraz konfiguracji i eksploatacji sieci
- radzi sobie w czasie awarii komputera lub systemu, w zakresie doboru odpowiedniego sprzętu i programów do konkretnych zastosowań
- korzysta z niekomercyjnych odpowiedników programów i systemów operacyjnych i zna ich ograniczenia
- wykorzystuje programy komputerowe, sieć i chmury informatyczne do rozwiązywania problemów z różnych dziedzin, np. innych przedmiotów szkolnych, obszarów zainteresowań i hobby, przygotowując się tym samym do przyszłej pracy zawodowej i studiów
- pracuje w zespole nad rozwiązywaniem problemów informatycznych, wykorzystując w tym celu różne sposoby komunikacji sieciowej, przesyłania danych i chmury informatyczne z ich usługami i oprogramowaniem
- sprawnie posługuje się arkuszem kalkulacyjnym do symulacji procesów, wizualizacji wyników doświadczeń itp.
- sprawnie tworzy grafikę komputerową z zastosowaniem edytorów grafiki rastrowej i wektorowej
- wykonuje korektę barwną, geometryczną, retusz, montaż i inne czynności dla fotografii cyfrowej
- przygotowuje grafikę, animacje i krótkie wideo do użycia w prezentacjach, tutorialach i umieszczenia na stronach internetowych
- dokonuje diagnostyki zestawu komputerowego i na podstawie jej wyników ocenia możliwości i pola zastosowań dla danej maszyny
- sprawnie posługuje się niektórymi programami narzędziowymi i diagnostycznymi podczas konfiguracji systemu operacyjnego i połączenia sieciowego
- konfiguruje bezprzewodowe połączenie z siecią i Internetem
- rozumie i używa pojęć informatycznych
- rozumie, wskazuje i unika zagrożeń związanych z działalnością przestępczą w sieci
- stosuje wszelkie zasady poszanowania praw autorskich i prawa do ochrony wizerunku, szczególnie w sieci Internet

Sprzyjać temu powinno odpowiednie przygotowanie materiałów i zadań, a także praca w chmurach informatycznych oraz stosowanie komunikatorów i innych sposobów elektronicznej komunikacji i przesyłania plików.

6. Propozycje kryteriów oceny i metod sprawdzania osiągnięć ucznia

Najważniejszym dokumentem opisującym sposoby oceniania postępów i osiągnięć uczniów jest wewnętrzny system oceniania. Wszystkie przedmiotowe regulaminy powinny być z nim całkowicie zgodne. Nie można więc ułożyć uniwersalnego Przedmiotowego Regulaminu Oceniania. Możliwe jest natomiast sformułowanie jego głównych założeń i niektórych kryteriów.

Uniwersalne kryteria pomocne w ułożeniu Przedmiotowego Systemu Oceniania

- a) Przed przystąpieniem do ćwiczeń uczniowie powinni znać kryteria, według których będą oceniani.
- b) Ocenie podlega ostateczny wynik ćwiczenia, np. dokument tekstowy, przetworzone zdjęcie itp.
- c) Na ocenę ma wpływ termin oddania pracy.
- d) Ocenie mogą podlegać także: aktywność na lekcji, zgodność z założeniami zadania, szybkość wykonania ćwiczenia, odpowiedzi ustne, sprawdziany on-line i off-line w postaci testów lub praktycznych ćwiczeń sprawdzających, dokładność wykonania ćwiczenia, estetyka, funkcjonalność projektu, umiejętność kierowania zespołem, prace domowe, przygotowanie do zajęć i inne wynikające z doświadczeń nauczyciela.

Kryteria na poszczególne oceny:

- ocenę **dopuszczającą** otrzymuje uczeń, który wykonuje ćwiczenie z pomocą nauczyciela lub kolegi z zespołu, niedbale korzysta z narzędzi i nie stosuje wszystkich zaleceń wynikających z zadania, nie dba o estetykę dokumentów, nie potrafi współpracować z zespołem
- ocenę **dostateczną** otrzymuje uczeń, który poprawnie użył określonych w zadaniu narzędzi, lecz nie wykorzystał ich wszystkich możliwości, w końcowym dokumencie lub projekcie występują niedoskonałości wynikające z niedbałości lub nieumiejętnego korzystania z narzędzi edytora, ma problemy ze współpracą z członkami zespołu
- ocenę **dobrą** otrzymuje uczeń, który samodzielnie wykonuje zadanie, dobrze współpracuje z członkami zespołu, oddaje prace w określonym terminie, projekty mają drobne niedoskonałości, występują nieuzasadnione odstępstwa od założeń zadania
- ocenę **bardzo dobrą** powinien uzyskać uczeń: samodzielnie wykonujący ćwiczenie lub swoje zadania w zespole, oddający prace w wyznaczonym terminie, używający wszystkich możliwych opcji narzędzi do poprawy jakości wykonanej pracy, oddający prace estetyczne bez wad, zawierające niewpływające na jakość pracy błędy lub niedociągnięcia

-ocenę **celującą** powinien uzyskać uczeń: perfekcyjnie wykonujący założenia danego projektu lub zadania, oddający prace w wyznaczonym czasie, pracujący samodzielnie lub spełniający wzorowo swoje zadania w zespole, np. lidera grupy; w przypadku zadań graficznych powinno być spełnione kryterium estetyki i zgodności z zasadami tworzenia danego rodzaju grafiki oraz prawidłowe użycie narzędzi edytorów

Na ocenę powinna mieć też wpływ ogólna postawa ucznia i kultura informatyczna. Jest to subiektywny czynnik, dlatego nauczyciel powinien stosować go z rozwagą.

Innym elementem wpływającym na ocenę pracy ucznia jest jego zaangażowanie w projekty informatyczne wykonywane na rzecz szkoły, organizacji młodzieżowej, zespołu itp. w czasie wolnym. W takim przypadku można wystawić dodatkowe oceny za aktywność.

Podsumowanie

Kryteria na poszczególne oceny powinien samodzielnie sformułować nauczyciel, korzystając z własnych doświadczeń, przykładowego opisu ocen, opierając się o Wewnątrzszkolny System Oceniania. Gotowy system powinien być ogłoszony uczniom na pierwszych zajęciach. Kryteria oceniania będą wtedy zrozumiałe dla wszystkich. Dla niektórych ćwiczeń można ustalić bardziej precyzyjne kryteria.

Uczniowie o specjalnych potrzebach edukacyjnych, dla których stosuje się kartę pracy i indywidualizację, podlegają tym samym kryteriom oceniania, z wyjątkiem ćwiczeń, w których nauczyciel przewidział odrębne kryteria.

Bardzo dobrym narzędziem do oceniania, archiwizowania i opisywania prac oraz przeprowadzania sprawdzianów wiedzy i umiejętności jest platforma e-learningowa. Jej stosowanie znacznie poprawia komfort pracy nauczyciela i pozwala uporządkować dokumentację zajęć. Sprawdziany – testy przeprowadzane za pośrednictwem platformy sprawdzane są automatycznie i oceniane według kryteriów ustalonych przez prowadzącego zajęcia. Nauczyciel może kontrolować terminowość oddawania prac i uzasadniać wystawioną ocenę. Możliwe jest też ocenianie prac po upływie pewnego czasu, co dodatkowo zwiększa komfort pracy. Oceny z platformy łatwo przepisać do dziennika. Można także analizować wyniki całej klasy lub grupy za pomocą wykresów i zestawień. Dzięki temu nauczyciel może ocenić poziom trudności postawionego przed uczniami zadania, a także poziom opanowania materiału w danej klasie. Zestawienia są generowane automatycznie i są dobrym narzędziem ewaluacyjnym.

7. Uwagi

Bezpieczeństwo

Skutki zagrożeń wynikających ze stosowania technologii informatycznych mogą być zminimalizowane dzięki właściwemu korzystaniu z programów, usług sieciowych i urządzeń telekomunikacyjnych. Dlatego ważnym celem jest uporządkowanie wiedzy uczniów na temat tych niebezpieczeństw i wskazanie odpowiednich, skutecznych metod ich unikania. Uczeń powinien także umieć łagodzić skutki już zaistniałych, negatywnych zdarzeń związanych z korzystaniem z komputerów i sieci.

Zagrożeniom podlegają nie tylko użytkownicy komputerów i sieci, ale także gromadzone dane i tworzone dokumenty. Uczeń powinien poznać i umieć stosować metody zabezpieczania danych i szyfrowania dokumentów, w tym także poczty elektronicznej.

Wyszukiwanie i przetwarzanie informacji

Uczniowie na tym etapie kształcenia potrafią wyszukiwać informacje w sieci i innych źródłach elektronicznych. Ich poczynania są często nieskuteczne, dlatego celem kształcenia powinno być uporządkowanie wiedzy, weryfikacja umiejętności i wskazanie właściwych metod zdobywania informacji, w tym skutecznego formułowania zapytań do wyszukiwarki. Uczeń powinien także umieć ustalać inne warunki wyszukiwania i oceniać wiarygodność źródeł.

Zdobyte informacje muszą być gromadzone i wykorzystane w odpowiedni sposób. Uczeń powinien umieć, na podstawie formatu pliku z informacją, odpowiednio ją odczytać, przetworzyć i zachować. Będzie więc musiał poszerzyć lub zdobyć nowe umiejętności korzystania z bazy danych, edytorów grafiki, dźwięku, wideo oraz systemów upowszechniania informacji w sieci, np. kreatorów www, blogów lub CMS. Do przetwarzania danych liczbowych konieczne będzie wykorzystanie arkusza kalkulacyjnego w szerszym niż do tej pory zakresie.

Ważną umiejętnością powinna być także weryfikacja wiarygodności informacji, szczególnie dotyczącej programowania. Należy zadbać o to, by uczniowie w czasie nauki klasycznych metod i algorytmów korzystali jedynie z przykładów i rozwiązań zgodnych ze standardami.

Rozwiązywanie problemów i podejmowanie decyzji

Programy komputerowe i zasoby Internetu mogą znacznie przyspieszyć proces nauki, opracowywania dokumentów, zdobywania wiedzy itp. Uczeń, dzięki ćwiczeniom, powinien nauczyć się wykorzystywać tę właściwość. Zdobyte informacje, wyniki symulacji, przykłady rozwiązań problemów informatycznych itp. powinny ułatwić mu podejmowanie decyzji dotyczących wyboru sposobów realizacji zadań, np. z innych przedmiotów. Do tego celu służyć także mają przykłady rozwiązań problemów znalezionych w różnych źródłach, np. sieci, ich selekcja, analiza i adaptacja do konkretnych zastosowań. Do rozwiązywania i analizy problemów uczeń powinien stosować metody planowania poszczególnych czynności – algorytmiczne. Odpowiednie zastosowanie dobrze dobranych programów komputerowych to umiejętność, której doskonalenie jest także celem ćwiczeń realizowanych w ramach zajęć przy komputerze. Dzięki zdobytym umiejętnościom wspomagania komputerowego oraz korzystania z nowoczesnych narzędzi uczniowie powinni szybciej i efektywniej rozwiązywać problemy i podejmować właściwe decyzje.

Poszerzanie wiedzy i umiejętności

Naturalne zachowania uczniów dotyczące korzystania z technologii elektronicznych, w tym rozrywki, powinny zostać wykorzystane w procesie rozwijania ich zainteresowań, pasji, a także uczenia się. Powinno to skutkować bardziej efektywnym wykorzystaniem komputerów do wspomagania procesów twórczych, w tym układania programów komputerowych oraz działań rozwijających umiejętności i wiedzę. Uczeń powinien umieć samodzielnie wykorzystać możliwości komputera osobistego do nauki, np. poprzez oglądanie materiałów filmowych, animacji ułatwiających zrozumienie niektórych tematów realizowanych na różnych przedmiotach. Powinien także umieć wykorzystać te źródła do poszerzania tej wiedzy, np. w związku z udziałem w zawodach i olimpiadach przedmiotowych.

Ocena i rozumienie społecznych skutków zastosowań informatyki

Realizacja programu nauczania powinna także prowadzić do świadomego korzystania z programów i zasobów sieciowych oraz umieszczania w sieci informacji, publikacji oraz własnych programów. Uświadomienie uczniom konsekwencji ich działań, rzeczywistego braku anonimowości w sieci, przepisów prawa dotyczącego ograniczeń i odpowiedzialności za upubliczniane informacje jest ważnym celem nie tylko przedmiotowym, ale i wychowawczym. Należy wyraźnie określić, jakie działania są niezgodne z prawem i w jaki sposób dochodzić swoich praw, gdy zostaną one naruszone. Uczeń powinien także wiedzieć, jak korzystać z fragmentów kodu programu pobranego z sieci, by nie naruszać praw, i jak udostępniać własne programy. Niezbędna jest także umiejętność przewidywania dalszego rozwoju elektroniki i informatyki oparta o lekturę czasopism, portali informatycznych i śledzenie trendów.

Diagnozowanie poziomu wiedzy i umiejętności uczniów przystępujących do nauki przedmiotu informatyka na IV etapie edukacyjnym

Dzięki dobrej diagnozie możliwa jest indywidualizacja pracy we wstępnej fazie roku szkolnego z zachowaniem wymagań programu nauczania. Ma to na celu wyrównanie poziomów umiejętności i wiedzy, które uczeń powinien nabyć w III etapie edukacyjnym oraz w czasie nauki w IV etapie na poziomie podstawowym.

Sieci i systemy komputerowe

Dział, w którym uczeń zapozna się z zaawansowanymi zagadnieniami dotyczącymi funkcjonowania sieci i różnych systemów operacyjnych. Rozpocznie także pracę z wykorzystaniem chmur informatycznych, nauczy się ich wykorzystania w pracy zespołu. Jest to przygotowanie do e-learningu i e-pracy.