

Wymagania Informatyka klasa 3 poziom podstawowy

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

<p>definiuje pojęcie baza danych;</p> <p>wyjaśnia, na czym polega przetwarzanie danych;</p> <p>na przykładzie gotowego pliku bazy danych potrafi omówić jej strukturę – określić, jakie informacje są w niej pamiętane</p>	<p>podaje obszary zastosowań baz danych – na przykładach z najbliższego otoczenia – szkoły, instytucji naukowych, społecznych i gospodarczych;</p> <p>wyjaśnia pojęcia: <i>baza danych</i>, <i>rekord</i> i <i>pole</i>;</p> <p>rozumie organizację danych w relacyjnych bazach danych;</p> <p>potrafi przygotować schemat prostej relacyjnej bazy danych;</p> <p>tworzy prostą bazę danych, składającą się z dwóch tabel;</p> <p>planuje zawartość tabel;</p> <p>stosuje zasady tworzenia tabel</p>	<p>omawia etapy przygotowania bazy danych;</p> <p>określa odpowiednio typy danych;</p> <p>rozumie pojęcia relacji i klucza podstawowego;</p> <p>przygotowuje projekt formularza i raportu;</p> <p>tworzy tabele i korzysta z Widoku projektu</p>	<p>rozumie, co oznacza przetwarzanie danych w bazach danych;</p> <p>definiuje relacje między tabelami;</p> <p>potrafi uzasadnić, dlaczego warto umieszczać dane w kilku tabelach połączonych relacją;</p> <p>podczas rozwiązywania nowego problemu korzysta z doświadczeń zdobytych przy rozwiązaniu innego, podobnego problemu</p>	<p>analizuje problemy występujące w utworzonej bazie danych i znajduje ich rozwiązanie;</p> <p>samodzielnie przygotowuje projekt bazy danych (składającej z trzech tabel) i potrafi ją wykonać w programie do tworzenia baz danych</p>
<p>Tworzenie i stosowanie formularzy w relacyjnej bazie danych</p>				
<p>wymienia poznane obiekty bazy danych: tabele, formularze, zapytania, raporty; wskazuje je, korzystając z gotowej bazy danych;</p> <p>korzysta z gotowych formularzy, wprowadzając przykładowe dane</p>	<p>potrafi dokonać modyfikacji wyglądu formularza;</p> <p>planuje i tworzy nowe formularze zgodnie z treścią ćwiczenia;</p> <p>korzysta z kreatora formularzy;</p> <p>wie na czym polega sortowanie danych;</p> <p>na podstawie przygotowanych formularzy ćwiczy wprowadzanie i aktualizację danych</p>	<p>potrafi wykonać sortowanie rekordów według wybranego pola;</p> <p>wie, czym różni się formularz standardowy od formularza z podformularzem;</p> <p>wie, jak utworzyć formularz z podformularzem;</p> <p>umie zaimportować dane z arkusza kalkulacyjnego do bazy danych</p>	<p>planuje i tworzy formularz z podformularzem) zgodnie z treścią ćwiczenia</p>	<p>samodzielnie projektuje wygląd formularzy, tworzy je i modyfikuje, korzystając z zaawansowanych możliwości modyfikacji formularzy, np. zmieniając różne własności</p>

Wykonywanie podstawowych operacji na relacyjnej bazie danych				
<p>zna zastosowanie filtrów do wyszukiwania danych;</p> <p>potrafi wyświetlić wynik gotowego zapytania i omówić, czego zapytanie dotyczy;</p> <p>wie do czego służą raporty;</p> <p>wyświetla gotowy raport i omawia, na podstawie jakich pól został utworzony;</p> <p>omawia zastosowanie korespondencji seryjnej</p>	<p>definiuje pojęcie kwerendy;</p> <p>tworzy kwerendę wybierającą w Widoku projektu;</p> <p>przygotowuje raporty do wydruku;</p> <p>zna sposób przygotowania korespondencji seryjnej z wykorzystaniem danych z bazy danych</p>	<p>modyfikuje gotowe zapytania;</p> <p>tworzy kwerendę parametryczną;</p> <p>potrafi utworzyć raport na podstawie kwerendy;</p> <p>umieszcza w korespondencji seryjnej pola z tabeli bazy danych;</p> <p>korzysta z gotowych szablonów listów seryjnych</p>	<p>samodzielnie modyfikuje i tworzy kwerendy oraz raporty;</p> <p>wie, jaka jest korelacja między edytorem tekstu a bazą danych podczas tworzenia korespondencji seryjnej;</p> <p>w edytorze tekstu przygotowuje listy seryjne i etykiety adresowe, korzystając z danych zapisanych w bazie danych</p>	<p>potrafi zaprojektować samodzielnie relacyjną bazę danych (składającą się z trzech tabel), ustala typy pól, projektuje wygląd formularzy;</p> <p>potrafi budować złożone kwerendy z dwóch lub więcej tabel połączonych;</p> <p>planuje i projektuje raporty;</p> <p>opracowuje własny szablon listu seryjnego</p>
Algorytmy szyfrowania tekstu i porządkowania liczb				
<p>wie, czym jest szyfrowanie danych i w jakim celu się je stosuje;</p> <p>potrafi, korzystając z przykładu z podręcznika, przeanalizować prosty przykład szyfrowania;</p> <p>wie, jak tworzy się anagramy i podaje przykłady anagramów;</p> <p>objaśnia z wykorzystaniem podręcznika i pomocy dydaktycznych jeden z algorytmów sortowania (bąbelkowy lub przez wstawianie)</p>	<p>zna pojęcia: <i>szyfr</i>, <i>szyfrowanie</i>, <i>deszyfrowanie</i>;</p> <p>omawia schemat procesu szyfrowania i deszyfrowania;</p> <p>zna przynajmniej jeden algorytm szyfrowania danych, np. szyfr Cezara;</p> <p>korzystając z podręcznika szyfruje wiadomość, korzystając z szyfru Cezara;</p> <p>omawia algorytmy sortowania bąbelkowego i przez wstawianie, korzystając z pomocy dydaktycznych</p>	<p>stosuje szyfrowanie tekstu metodą Cezara do szyfrowania różnych tekstów;</p> <p>odszyfrowuje słowa zaszyfrowane metodą Cezara;</p> <p>objaśnia na przykładzie szyfrowanie tekstu metodą przestawieniową;</p> <p>tworzy anagramy</p>	<p>wie, czym zajmuje się kryptologia i kryptoanaliza;</p> <p>formułuje algorytm deszyfrowania podstawieniowego</p> <p>porządkuje liczby metodą bąbelkową;</p> <p>objaśnia szyfrowanie symetryczne i asymetryczne;</p> <p>przedstawia sposób utworzenia anagramu</p>	<p>omawia dziedziny gospodarki, w których wykorzystywane jest szyfrowanie danych;</p> <p>wyszukuje dodatkowe informacje na temat szyfrowania danych;</p> <p>omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat;</p> <p>wyszukuje dodatkowe informacje na temat Enigmy i algorytmu RSA</p>
Metoda połowienia, podejście zachłanne, rekurencja i działania na ułamkach				

<p>objaśnia na przykładzie algorytm wyszukiwania przez połowienie; wie, na czym polega metoda zachłanna – podaje przykłady jej stosowania, wykonując proste ćwiczenia z podręcznika (np. problem umieszczenia książek w samochodzie); definiuje pojęcie iteracji i rekurencji; wyjaśnia, na czym polega wyznaczanie NWD i NWW</p>	<p>analizuje algorytm wyszukiwania przez połowienie, korzystając z przykładu (z książkami) z podręcznika; potrafi dokonać analizy algorytmu wydawania reszty metodą zachłanną; zna rekurencyjną definicję funkcji obliczającej silnię liczby naturalnej; definiuje pojęcie algorytmu zachłannego; podaje przykłady zjawisk rekurencyjnych</p>	<p>Dokonuje przykładowej analizy rozwiązania algorytmu wyszukiwania przez połowienie na podanym przykładzie dziesięciu liczb; potrafi dokonać analizy wydawania reszty metodą zachłanną, korzystając z pomocy dydaktycznych; zna algorytm rekurencyjny i iteracyjny obliczania wartości elementów ciągu Fibonacciego; wie, czym jest stos w informatyce; zna wzory na obliczanie sumy ułamków zwykłych oraz mnożenie i dzielenie ułamków zwykłych</p>	<p>sprawdza działanie algorytmu wyszukiwania liczby w zbiorze uporządkowanym metodą przez połowienie; rozumie różnicę między rekurencją a iteracją; potrafi wykonać zaawansowane zadania związane z algorytmem zachłannym; oblicza wybrane liczby Fibonacciego; objaśnia samodzielnie sposób „układania” stosu wywołań rekurencyjnych i „zdejmwania” ze stosu wartości silni lub liczb Fibonacciego</p>	<p>rozumie dokładnie technikę rekurencji (znaczenie stosu); potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję; samodzielnie formułuje problem, do którego rozwiązania można zastosować rekurencję; korzysta samodzielnie z dodatkowej literatury fachowej</p>
<p>Programowanie algorytmów szyfrowania i porządkowania</p>				

<p>korzystając z podręcznika, omawia sposób zaprogramowania wybranego algorytmu szyfrowania;</p> <p>analizuje i potrafi omówić fragment programu z podręcznika realizujący wybrany algorytm porządkowania</p>	<p>wie, jak zamienić literę na jej pozycję w alfabecie i odwrotnie;</p> <p>analizuje i rozumie działanie funkcji szyfrujących (podanych w podręczniku) realizujących wybrany algorytm szyfrowania;</p> <p>pisze program w wybranym języku programowania tworzący prosty anagram polegający na przestawieniu liter w odwrotnej kolejności;</p> <p>analizuje i potrafi omówić funkcje porządkujące liczby metodą bąbelkową i przez wstawianie</p>	<p>pisze program zamieniający literę na jej pozycję w alfabecie i odwrotnie oraz używa w programie wbudowanej funkcji <code>toupper()</code> języka C++ / <code>upper()</code> języka Python;</p> <p>tworzy program realizujący wybrany algorytm szyfrowania, np. szyfr Cezara;</p> <p>testuje programy dla różnych danych;</p> <p>zna zasady programowania algorytmu szyfrowania przedstawieniowego;</p> <p>korzystając z gotowej funkcji z podręcznika pisze program porządkujący liczby metodą bąbelkową</p>	<p>tworzy programy realizujące algorytm szyfrowania metodą Cezara i przestawieniową;</p> <p>pisze funkcję deszyfrowania podstawieniowego i funkcję deszyfrowania przedstawieniowego oraz tworzy programy z wykorzystaniem tych funkcji</p> <p>pisze program w wybranym języku programowania tworzący anagramy dla podanego słowa;</p> <p>korzystając z gotowej funkcji z podręcznika pisze program porządkujący liczby metodą przez wstawiania</p>	<p>opracowuje rozbudowane programy szyfrujące i deszyfrujące;</p> <p>opracowuje rozbudowane programy szyfrowania i deszyfrowania przestawieniowego;</p> <p>opracowuje rozbudowane anagramy i potrafi je zaprogramować;</p> <p>samodzielnie tworzy programy porządkowania metodą bąbelkową i przez wstawianie</p>
<p>Programowanie algorytmów zachłannych, rekurencyjnych oraz działań na ułamkach</p>				

<p>analizuje opis programów z podręcznika; zapisuje przykładowy program realizujący algorytm zachłanny w języku C++/Python; zna iteracyjną i rekurencyjną definicję funkcji obliczającej silnię liczby naturalnej; zna podstawowe działania arytmetyczne na ułamkach</p>	<p>zapisuje algorytm Euklidesa w wersji z odejmowaniem oraz z dzieleniem w języku C++/Python; korzystając pisze program realizujący algorytm obliczania silni (w wersji iteracyjnej i rekurencyjnej) pisze program realizujący algorytm obliczania silni (w wersji iteracyjnej); opracowuje funkcję w języku programowania wyznaczającą NWD; opracowuje funkcję w języku programowania wyznaczającą NWW</p>	<p>korzystając z opisów w podręczniku, definiuje funkcję wyszukiwania liczby w zbiorze uporządkowanym metodą połowienia; korzystając z podręcznika, definiuje funkcję realizującą algorytm wydawania reszty metodą zachłanną w wybranym języku programowania; testuje programy dla różnych danych; definiuje funkcje realizujące algorytm Euklidesa w dwóch wersjach; analizuje funkcje skracającą ułamki zwykłe i dodająca ułamki zwykłe oraz stosuje je do napisania programów: skracającego ułamki zwykłe oraz dodającego ułamki zwykłe</p>	<p>pisze program realizujący algorytm wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia; pisze program realizujący algorytm zachłanny wydawania reszty z wykorzystaniem odpowiedniej funkcji; zapisuje w postaci programu rekurencyjną realizację algorytmu obliczającego liczby Fibonacciego; zapisuje w postaci programu rekurencyjną realizację algorytmu obliczającego silnię liczby naturalnej; definiuje funkcje: odejmującą, mnożącą oraz dzielącą ułamki zwykłe; wywołuje funkcje w programie głównym</p>	<p>zapisuje w postaci programu rekurencyjną wersję poznanego wcześniej algorytmu (np. algorytmu Euklidesa); samodzielnie formułuje problem, do którego można zastosować podejście zachłanne i pisze program rozwiązujący ten problem; rozumie różnice między iteracją a rekurencją; bierze udział w konkursach informatycznych i/lub olimpiadzie informatycznej</p>
<p>Elementy robotyki – projekty</p>				

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