Wymagania edukacyjne z informatyki ZR w szkole ponadpodstawowej kl. 5Fb

Elementy analizy algorytmów				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wymienia własności algorytmów. Potrafi przeanalizować przebieg algorytmu zapisanego w postaci listy kroków lub w postaci schematu blokowego dla przykładowych danych i ocenić w ten sposób jego poprawność.	Zna i omawia własności algorytmów. Potrafi ocenić poprawność działania algorytmu i jego zgodność ze specyfikacją. Określa liczbę prostych działań zawartych w algorytmie.	Rozumie, co to jest złożoność czasowa algorytmu i potrafi określić liczbę operacji wykonywanych na elementach zbioru w wybranym algorytmie sortowania. Rozróżnia złożoność czasową i pamięciową.	Wie, jak ocenić złożoność pamięciową algorytmu. Potrafi porównać złożoność obliczeniową różnych algorytmów tego samego zadania dla tych samych danych. Wie, kiedy algorytm jest efektywny.	Określa złożoność czasową i pamięciową wybranych algorytmów. Zna odpowiednie wzory. Określa efektywność algorytmów.

Reprezentacja danych w komputerze				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna pojęcie systemu pozycyjnego. Wie, co to jest system binarny. Korzystając z przykładów, potrafi obliczyć wartość dziesiętną liczby zapisanej w systemie dwójkowym.	Wyjaśnia, co to jest system binarny, i potrafi dokonać zamiany liczby z systemu dziesiętnego na binarny i odwrotnie.	Zna system szesnastkowy i potrafi wykonać konwersję liczb dziesiętnych na liczby w systemie szesnastkowym i odwrotnie. Zna zależność między systemem binarnym i szesnastkowym.	Potrafi wykonać dowolną konwersję pomiędzy systemem dziesiętnym, dwójkowym i szesnastkowym. Potrafi napisać program obliczający wartość dziesiętną liczby dwójkowej.	Potrafi napisać program (w wersji iteracyjnej i rekurencyjnej) realizujący algorytm zamiany liczby dziesiętnej na postać binarną. Potrafi napisać program realizujący algorytm umożliwiający zamianę liczb z systemu szesnastkowego na dziesiętny i odwrotnie.

Kompresja i szyfrowanie danych				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wie, co to jest kompresja danych. Zna przynajmniej jeden algorytm kompresji danych. Wie, czym jest szyfrowanie danych. Zna przynajmniej jeden algorytm szyfrowania danych.	Rozumie, na czym polega kompresja danych i w jakim celu się ją wykonuje. Wymienia rodzaje kompresji. Omawia jeden przykładowy algorytm kompresji. Koduje tekst, używając alfabetu Morse'a. Wymienia przykładowe algorytmy szyfrowania. Potrafi zaszyfrować i odszyfrować prosty tekst.	Wie, co to jest współczynnik kompresji. Omawia rodzaje kompresji: kompresję stratną i bezstratną. Podaje przykłady algorytmów kompresji stratnej i bezstratnej. Omawia algorytm statyczny i słownikowy. Omawia przynajmniej dwa algorytmy szyfrowania: szyfr podstawieniowy i przestawieniowy.	Potrafi policzyć współczynnik kompresji. Wyjaśnia różnicę pomiędzy algorytmem statycznym a słownikowym. Stosuje algorytm słownikowy do kompresji ciągu znaków. Omawia przykładowy szyfr z kluczem. Stosuje szyfr Vigenère'a do zaszyfrowania ciągu znaków. Omawia wykorzystanie algorytmów szyfrowania w podpisie elektronicznym.	Wyszukuje dodatkowe informacje na temat kompresji i szyfrowania danych. Omawia inne algorytmy kompresji i szyfrowania. Potrafi zapisać wybrany algorytm kompresji lub szyfrowania w postaci programu. Zapoznaje się samodzielnie z kodem Huffmana i pokazuje na przykładzie jego zastosowanie. Omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat. Wyjaśnia, w jaki sposób tworzy się podpis elektroniczny.

Algorytmy i zależności funkcyjne w arkuszu kalkulacyjnym				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna zasady tworzenia formuł i stosowania funkcji arkusza kalkulacyjnego. Zna i stosuje zasady adresowania względnego i bezwzględnego w arkuszu kalkulacyjnym. Zapisuje w arkuszu kalkulacyjnym algorytm liniowy i z warunkami. Stosuje wybrane funkcje arkusza kalkulacyjnego. Potrafi utworzyć wykres w arkuszu kalkulacyjnym.	Zapisuje w arkuszu kalkulacyjnym algorytm z warunkami zagnieżdżonymi. Zna i stosuje zasady adresowania mieszanego w arkuszu kalkulacyjnym. Potrafi zrealizować iterację w arkuszu kalkulacyjnym. Rysuje wykres funkcji liniowej i kwadratowej.	Potrafi zrealizować pętlę zagnieżdżoną w arkuszu kalkulacyjnym. Wie, co to jest fraktal i w jaki sposób się go tworzy. Podaje przykłady fraktali. Rysuje wykres wybranej funkcji trygonometrycznej. Dobiera odpowiedni typ wykresu do prezentowanych danych.	Rysuje wykres funkcji liniowej, wielomianu, wybranej funkcji trygonometrycznej i funkcji logarytmicznej. Zna możliwości zastosowania algorytmów iteracyjnych w rysowaniu fraktali, m.in.: śnieżynki Kocha, dywanu i trójkąta Sierpińskiego. Stosuje wybrane możliwości arkusza kalkulacyjnego do rozwiązywania zadań z różnych dziedzin.	Potrafi samodzielnie zobrazować wybraną zależność funkcyjną w arkuszu kalkulacyjnym, np. algorytm rozwiązywania układu równań liniowych metodą wyznaczników. Rozumie, w jaki sposób narysować paprotkę Barnsleya w arkuszu kalkulacyjnym. Korzystając z Internetu i innych źródeł, wyszukuje dodatkowe informacje na temat geometrii fraktalnej, m.in. dotyczące jej zastosowań.

Realizacja projektu programistycznego				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Uczestniczy czynnie w projekcie grupowym, wykonując proste zadania, np. wprowadza dane do bazy i je aktualizuje. Bierze udziałw testowaniu projektu.	Zna wszystkie etapy projektowania systemów informatycznych. Uczestniczy czynnie w poszczególnych etapach projektu, wykonując zlecone zadania szczegółowe. Planuje temat projektu.	Wie, co to jest system informatyczny. Potrafi omówić zakres prac na każdym etapie. Realizuje projekt na zadany (lub samodzielnie wybrany) temat zgodnie z etapami projektowania. Uczestniczy czynnie w analizie systemu informacyjnego, przygotowuje dokumentację. Przygotowuje założenia w postaci dokumentów edytora tekstu, korzystając z szablonów. Współpracuje przy projektowaniu tabel, formularzy i raportów.	Realizuje projekt zgodnie z zamierzoną organizacją pracy zespołowej i wytyczonymi wcześniej etapami projektowania. Wykonuje trudniejsze prace związane z projektowaniem systemu. Przeprowadza analizęsystemu informacyjnego. Projektuje złożone kwerendy, formularze, raporty. Uczestniczy we wdrażaniu systemu informatycznego.	Potrafi pełnićfunkcję koordynatora projektu. Przydziela zadania szczegółowe, dba o ich prawidłowe wykonanie, nadzoruje pracęinnych, dba o dobrą atmosferę w grupie.

Wprowadzanie danych i wyprowadzanie wyników				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Zna i stosuje podstawowe sposoby wprowadzania danych i wyprowadzania wyników w wybranym języku programowania.	Potrafi poprawnie zadeklarować dane potrzebne do rozwiązania zadania. Wie, co określa typ danych i format danych. Potrafi odpowiednio sformatować wyprowadzane wyniki.	Tworzy własne typy danych (typ okrojony, typ wyliczeniowy) i stałe. Zna procedury i funkcje umożliwiające generowanie danych losowych. Stosuje w programach generowanie losowych danych.	Poprawnie formatuje wyprowadzane wyniki. Omawia stos jako przykład struktury danych. Wykonuje operacje na stosie. Implementuje stos z wykorzystaniem tablicy. Zna i omawia zastosowanie stosu (odwrotną notacjępolską– ONP).	Dobiera sposób wprowadzania danych do rozwiązywanego zadania. Potrafi sprawdzićpoprawność danych wprowadzanych do programu. Definiuje własne typy danych, potrzebne do rozwiązania danego zadania. Korzystając z możliwości generowania danych losowych, tworzy własne gry komputerowe.

Programowanie modularne i obiektowe				
2	**3**	**4**	**5**	**6**
Uczeń:	Uczeń:	Uczeń:	Uczeń:	Uczeń:
Wymienia modele programowania. Omawia model programowania strukturalnego. Wie, jakie są korzyści z definiowania procedur i funkcji. Potrafi zdefiniować procedury i funkcje w wybranym języku programowania. Wie, co to jest moduł.	Omawia szczegółowo modele programowania. Zna zasady programowania modularnego. Analizując przykładowe programy, tworzy własne moduły.	Wie, na czym polega programowanie obiektowe. Zna podstawowe pojęcia programowania obiektowego: klasa, obiekt, pola, metody.	Rozumie i potrafi zastosować typ obiektowy. Definiuje klasy. Deklaruje pola prywatne. Omawia cechy programowania obiektowego: dziedziczenie i polimorfizm. Wie, czym są metody wirtualne. Poznaje zastosowanie tych cech, analizując gotowe programy. Modyfikuje programy według wskazówek nauczyciela.	Tworzy własne programy, stosując poznane zasady programowania modularnego i obiektowego. Korzysta z fachowej literatury. Wie, na czym polega programowanie zdarzeniowe. Rozwiązuje zadania z olimpiady informatycznej i bierze w niej udział.