

**WYMAGANIA EDUKACYJNE Z INFORMATYKI NIEZBĘDNE DO UZYSKANIA PRZEZ UCZNIA**

**POSZCZEGÓLNYCH ŚRÓDROCZNYCH I ROCZNYCH OCEN KLASYFIKACYJNYCH WYNIKAJĄCYCH Z REALIZOWANEGO PROGRAMU NAUCZANIA**

 **LICEUM 4-LETNIE KLASA 4.**

 **ZAKRES ROZSZERZONY**

Uczeń spełnia wszystkie wymagania edukacyjne dla poziomu podstawowego, a ponadto wymagania wyszczególnione poniżej

 Wokół komputera i sieci komputerowych

|  |
| --- |
| **Szczegółowe wymagania edukacyjne dla klas 4H** |
| Ocenę **niedostateczną** otrzymuje uczeń, który nie spełnia wymagań edukacyjnych niezbędnych do uzyskania oceny dopuszczającej. |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** |
|  Reprezentacja danych w komputerze- zadania maturalne |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| zna pojęcie systemu pozycyjnego; wie, co to jest system binarny;analizuje gotowy przykład obliczający wartość dziesiętną liczby zapisanej w systemie dwójkowym i na tej podstawie wykonuje podobne ćwiczenie;w zależności od problemu rozwiązuje go, stosując metodę wstępującą lub zstępującą; | definiuje pojęcie systemu pozycyjnego;wie na czym polega działanie procesora;potrafi dokonać konwersji liczby między systemem dziesiętnym a dwójkowym oraz dwójkowym a dziesiętnym;do realizacji rozwiązania problemu dobiera odpowiednią metodę lub technikę algorytmiczną i struktury danych; | zna podwójne nazwy pozycyjnych systemów liczbowych tj. dwójkowy – binarny, dziesiętny – decymalny, szesnastkowy - heksadecymalny;wyjaśnia, w jaki sposób procesor dodaje liczby;dokonuje konwersji liczb między systemem dziesiętnym i szesnastkowym oraz szesnastkowym i dziesiętnym;objaśnia dobrany algorytm, uzasadnia poprawność rozwiązania na wybranych przykładach danych i ocenia jego efektywność; | zna elementy uproszczonego modelu komputera zgodny z ideą von Neumanna;dokonuje konwersji liczb między systemem szesnastkowym i binarnym;ilustruje i wyjaśnia rolę pojęć, obiektów i operacji matematycznych w projektowaniu rozwiązań problemów informatycznych i z innych dziedzin, posługuje się pojęciem logarytmu;przedstawia sposoby reprezentowania w komputerze znaków, liczb, wartości logicznych, obrazów, dźwięków, animacji; | potrafi narysować uproszczony model komputera zgodny z ideą von Neumanna;samodzielnie potrafi dokonać zamiany między trzema systemami pozycyjnymi (dwójkowym, dziesiętnym i szesnastkowym) w jednym zadaniu;objaśnia sposoby wykonywania przez komputer działań arytmetycznych i operacji logicznych;wyjaśnia, jakie może być źródło błędów pojawiających się w obliczeniach komputerowych: błąd zaokrąglenia, błąd przybliżenia;dyskutuje na temat roli myślenia komputacyjnego i jego metod, takich jak: abstrakcja, reprezentacja danych, dekompozycja problemu, redukcja, myślenie rekurencyjne, podejście heurystyczne w rozwiązywaniu problemów z różnych dziedzin. |

 Wokół dokumentów komputerowych

|  |
| --- |
|  Bazy danych- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| wyjaśnia, na czym polega przetwarzanie danych;definiuje pojęcie baza danych;na przykładzie gotowego pliku bazy danych potrafi omówić jej strukturę – określić, jakie informacje są w niej pamiętane | podaje obszary zastosowań baz danych – na przykładach z najbliższego otoczenia – szkoły, instytucji naukowych, społecznych i gospodarczych;wyjaśnia pojęcia: *baza danych*, *rekord* i *pole*;rozumie organizację danych w relacyjnych bazach danych;potrafi przygotować schemat prostej relacyjnej bazy danych;tworzy prostą bazę danych, składającą się z dwóch tabel: planuje zawartość tabel; stosuje zasady tworzenia tabel | omawia etapy przygotowania bazy danych;określa odpowiednio typy danych;rozumie pojęcia relacji i klucza podstawowego; przygotowuje projekt formularza i raportu;tworzy tabele i korzysta z **Widoku projektu** | rozumie, co oznacza przetwarzanie danych w bazach danych;definiuje relacje między tabelami;potrafi uzasadnić, dlaczego warto umieszczać dane w kilku tabelach połączonych relacją;podczas rozwiązywania nowego problemu korzysta z doświadczeń zdobytych przy rozwiązaniu innego, podobnego problemu, rozwiązuje zadania maturalne wykorzystując program Access | analizuje problemy występujące w utworzonej bazie danych i znajduje ich rozwiązanie;samodzielnie przygotowuje projekt bazy danych (składającej z trzech tabel) i potrafi ją wykonać w programie do tworzenia baz danych, **rozwiązuje zadania maturalne wykorzystując program Access bezbłędnie**. |

|  |
| --- |
|  Formularze w relacyjnej bazie danych- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| wymienia poznane obiekty bazy danych: tabele, formularze, zapytania, raporty; wskazuje je, korzystając z gotowej bazy danych; korzysta z gotowych formularzy, wprowadzając przykładowe dane | potrafi dokonać modyfikacji wyglądu formularza;planuje i tworzy nowe formularze zgodnie z treścią ćwiczenia; korzysta z kreatora formularzy;wie na czym polega sortowanie danych;na podstawie przygotowanych formularzy ćwiczy wprowadzanie i aktualizację danych | potrafi wykonać sortowanie rekordów według wybranego pola;wie, czym różni się formularz standardowy od formularza z podformularzem;wie, jak utworzyć formularz z podformularzem;umie zaimportować dane z arkusza kalkulacyjnego do bazy danych | planuje i tworzy formularz z podformularzem) zgodnie z treścią ćwiczenia | samodzielnie projektuje wygląd formularzy, tworzy je i modyfikuje, korzystając z zaawansowanych możliwości modyfikacji formularzy, np. zmieniając różne własności**, także w zadaniach maturalnych**. |

|  |
| --- |
|  Wykonywanie operacji na relacyjnej bazie danych- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| zna zastosowanie filtrów do wyszukiwania danych;potrafi wyświetlić wynik gotowego zapytania i omówić, czego zapytanie dotyczy;wie do czego służą raporty;wyświetla gotowy raport i omawia, na podstawie jakich pól został utworzony;omawia zastosowanie korespondencji seryjnej | definiuje pojęcie kwerendy;tworzy kwerendę wybierającą w **Widoku projektu**;przygotowuje raporty do wydruku;zna sposób przygotowania korespondencji seryjnej z wykorzystaniem danych z bazy danych | modyfikuje gotowe zapytania;tworzy kwerendę parametryczną;potrafi utworzyć raport na podstawie kwerendy;umieszcza w korespondencji seryjnej pola z tabeli bazy danych;korzysta z gotowych szablonów listów seryjnych | samodzielnie modyfikuje i tworzy kwerendy oraz raporty;wie, jaka jest korelacja między edytorem tekstu a bazą danych podczas tworzenia korespondencji seryjnej;w edytorze tekstu przygotowuje listy seryjne i etykiety adresowe, korzystając z danych zapisanych w bazie danych | potrafi zaprojektować samodzielnie relacyjna bazę danych (składającą się z trzech tabel), ustala typy pól, projektuje wygląd formularzy; potrafi budować złożone kwerendy z dwóch lub więcej tabel połączonych;planuje i projektuje raporty;opracowuje własny szablon listu seryjnego |

|  |
| --- |
| Wykonywanie operacji w arkuszu kalkulacyjnym- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| z pomocą nauczyciela tworzy wykres funkcji liniowej, z pomocą nauczyciela pobiera surowe dane z czujników z pomocą nauczyciela planuje kolejne kroki symulacji w arkuszu porządkuje dane, aby móc utworzyć tabelę przestawną z pomocą nauczyciela tworzy ankietę w chmurze. | tworzy wykres funkcji liniowej przygotowuje dane do analizy korzysta z funkcji zaokrąglania wyników, tworzy wykresy przebiegu w czasie pobiera i importuje do arkusza wyniki ankiety | tworzy wykres funkcji kwadratowej zmienia wartości opracowuje pobrane dane, dobiera odpowiednie narzędzia przeprowadza symulację samodzielnie korzysta z Pomocy arkusza tworzy tabele przestawne stosuje zaawansowane kryteria filtrowania | tworzy złożone wykresy funkcji automatyzuje proces tworzenia wykresów, opracowuje wyniki wykorzystuje linie trendu w wykresach funkcji liniowej wprowadza dynamiczne tytuły osi wykresów prawidłowo dobiera pola do wyświetlania w tabeli przestawnej dokonuje wizualizacji danych z wykorzystaniem wykresów przebiegu w czasie korzysta z fragmentatorów tworzy raporty z danych z wykorzystaniem tabeli przestawnych i wykresów przebiegu w czasie. | przygotowuje trudniejsze wykresy, np. wykresy przestrzenne funkcji dwóch zmiennych samodzielnie wykonuje doświadczenia i eksperymenty analizuje wyniki dodatkowych doświadczeń i eksperymentów samodzielnie planuje i realizuje symulacje, np. o charakterze przyrodniczym samodzielnie wykorzystuje tabele przestawne do analizy różnych danych samodzielnie planuje i realizuje badanie na porządkuje dane i tworzy raport wybrany temat – przeprowadza ankietę, **rozwiązuje zadania maturalne w programie Excel** |

#  Wokół algorytmiki i programowania

|  |
| --- |
|  Algorytmy szyfrujące w językach C++ i Python- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| wie, że w edytorach tekstu wykorzystywane są algorytmy na tekstach – pokazuje przykłady wyszukiwania znaków w tekście, porównywania tekstów; potrafi omówić, posługując się przykładami i pomocami dydaktycznymi, wybrany algorytm na tekstach;analizuje gotowy program wykorzystujący dane tekstowe i objaśnia stosowanie zmiennych tekstowych;testuje gotowe programy dla różnych danych  | planuje kolejne kroki rozwiązania problemu porównywania tekstów, szukając rozwiązania;deklaruje zmienne typu tekstowego **char** i string;realizuje algorytm porównania dwóch tekstów;wie, na czym polega algorytm szukania wzorca w tekście;wie, jak odwoływać się do pojedynczego znaku łańcucha, wie, jak wyznaczyć długość łańcucha – potrafi zastosować funkcję length() oraz strlen() (C++) i funkcję len() (Python);analizuje funkcję realizującą np. algorytm porównywania tekstów i omawia działanie funkcji w tym zastosowane instrukcje; tworzy program wykorzystujący tę funkcję | przeprowadza analizę rozwiązania algorytmu szukania znaku w tekście;szuka wystąpień wzorca w tekście metodą naiwną, analizując i uzupełniając kolejne kroki algorytmu z wykorzystaniem podręcznika;definiuje pojęcie konkatencji;formułuje treść zadania do przedstawionego kodu źródłowego;definiuje funkcję szukającą i zliczającą wystąpienia znaków lub ciągów znaków w tekście;korzysta z instrukcji if…elif (Python) do utworzenia menu programu | opracowuje program zliczający wystąpienie znaku w tekście;korzysta z funkcji wyboru **switch()** (C++) do utworzenia menu programu;umieszcza w kodzie źródłowym funkcję getline() do wprowadzania napisów składających się z wyrazów oddzielonych spacjami, **rozwiązuje zadania maturalne wykorzystując podane funkcje** | tworzy zaawansowane programy wykorzystujące dane tekstowe i poznane funkcje;potrafi samodzielnie utworzyć algorytm i program realizujący porównywanie tekstów;tworzy rozbudowane menu wyboru z wykorzystaniem funkcji **switch()** (C++);tworzy rozbudowane menu wyboru z wykorzystaniem z instrukcji **if … elif** (Python**), wykorzystuje podane algorytmy w zadaniach maturalnych** |
|  Algorytmy na tekstach w językach C++ i Python- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| wie, czym jest szyfrowanie danych i w jakim celu się je stosuje;potrafi, korzystając z przykładu z podręcznika, przeanalizować prosty przykład szyfrowania;wie, jak się tworzy anagramy i podaje przykłady anagramów | zna pojęcia: *szyfr*, *szyfrowanie*, *deszyfrowanie*;omawia schemat procesu szyfrowania i deszyfrowania;zna przynajmniej jeden algorytm szyfrowania danych – szyfr Cezara;korzystając z podręcznika szyfruje i deszyfruje wiadomość, korzystając z szyfru Cezara;przedstawia sposób utworzenia anagramu;analizuje i rozumie działanie funkcji (podanych w podręczniku) realizujących wybrany algorytm szyfrowania | wie, czym zajmuje się kryptologia i kryptoanaliza;zna zasady programowania algorytmu szyfrowania przedstawieniowegostosuje szyfrowanie tekstu metodą Cezara i przestawieniową;pisze program w wybranym języku programowania tworzący anagramy;tworzy program realizujący algorytm szyfrowania szyfrem Cezara;formułuje algorytm deszfrowania podstawieniowego i pisze funkcję realizująca ten algorytm | pisze program realizujący algorytm szyfrowania przedstawieniowego ( opracowuje odpowiednie funkcje pomocnicze);pisze funkcję deszyfrowania przestawieniowego;objaśnia szyfrowanie symetryczne i asymetryczne | omawia dziedziny gospodarki, w których wykorzystywane jest szyfrowanie danych;opracowuje rozbudowane anagramy i potrafi je zaprogramować;wyszukuje dodatkowe informacje na temat szyfrowania danych;omawia, czym się zajmuje stenografia, samodzielnie wyszukując informacje na ten temat |

|  |
| --- |
|  Metoda połowienia, podejście zachłanne i rekurencja- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| 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);podaje przykłady zjawisk rekurencyjnych | analizuje algorytm wyszukiwania przez połowienie;potrafi dokonać analizy algorytmu wydawania reszty metodą zachłanną;pisze program realizujący algorytm obliczania silni ( w wersji iteracyjnej);zna rekurencyjną definicję funkcji obliczającej silnię liczby naturalnej | korzystając z opisów w podręczniku, definiuje funkcję wyszukiwania liczby w zbiorze uporządkowanym metoda połowienia;korzystając z podręcznika, definiuje funkcję realizującą algorytm wydawania reszty metodą zachłanną w wybranym języku programowania;zapisuje w postaci programu rekurencyjną realizację algorytmu obliczającego silnię liczby naturalnej;testuje programy dla różnych danych | pisze program realizujący algorytm wyszukiwania elementu w zbiorze uporządkowanym metodą połowienia;pisze program realizujący algorytm zachłanny wydawania reszty z wykorzystaniem odpowiedniej funkcji;potrafi zapisać w postaci programu rekurencyjną realizację algorytmu obliczającego liczby Fibonacciego;rozumie różnicę między rekurencją a iteracją | rozumie dokładnie technikę rekurencji (znaczenie stosu);potrafi ocenić, kiedy warto stosować iterację, a kiedy rekurencję;samodzielnie formułuje problem, do którego rozwiązania można zastosować rekurencję;zapisuje w postaci programu rekurencyjna wersję poznanego wcześniej algorytmu (np. algorytmu Euklidesa);korzysta samodzielnie z dodatkowej literatury fachowej |

|  |
| --- |
|  Algorytmy na liczbach w językach C++ i Python- zadania maturalne |
| Wymagania edukacyjne niezbędne do uzyskania **oceny dopuszczającej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dostatecznej** | Wymagania edukacyjne niezbędne do uzyskania **oceny dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny bardzo dobrej** | Wymagania edukacyjne niezbędne do uzyskania **oceny celującej** |
| **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** | **Uczeń:** |
| przeprowadza konwersję liczby z systemu dwójkowego na dziesiętny oraz z dziesiętnego na dwójkowy (bazując na przykładach z tematu A1);wyjaśnia, na czym polega wyznaczanie NWD i NWW | korzystając z fragmentu programów z podręcznika, pisze programy w wybranym języku programowania (C++, Python): obliczający wartość dziesiętną liczby dwójkowej i wyznaczający rozwinięcie dwójkowe liczby dziesiętnej;testuje programy dla różnych danych;opracowuje funkcję w języku programowania wyznaczającą NWD;opracowuje funkcję w  języku programowania wyznaczającą NWW | analizuje algorytm zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie (listę kroków i schemat blokowy), wyróżniając podproblemy;testuje działanie algorytmów: obliczania wartości dziesiętnej liczby zapisanej w dowolnym systemie i zapisywania liczby dziesiętnej w systemie liczbowym o określonej podstawie, korzystając z odpowiednich list kroków ;wie, jak można reprezentować ułamek zwykły w języku C++ (deklaruje rekordy za pomocą słowa kluczowego struct) lub w języku Python (omawia przykład klasy i tworzy zmienną danej klasy);zna wzory na obliczanie sumy ułamków zwykłych oraz mnożenie i dzielenie ułamków zwykłych | na podstawie list kroków (podanych w podręczniku) tworzy programy: obliczania wartości dziesiętnej liczby zapisanej w dowolnym systemie i zapisujący liczbę dziesiętną w systemie liczbowym o określonej podstawie;potrafi zastosować schemat Hornera do obliczenia wartości wielomianu;analizuje funkcje skracającą ułamki zwykłe i dodająca ułamki zwykłe oraz stosuje je do napisania programów: skracającego ułamki zwykłe oraz dodającego ułamki zwykłe;definiuje funkcje: odejmującą, mnożącą oraz dzielącą ułamki zwykłe; wywołuje funkcje w programie głównym i testuje programy dla różnych danych;zna algorytmy: wydawania reszty metodą zachłanną, pakowania plecaka, znajdowania miejsc zerowych funkcji, znajdowania najkrótszej drogi (grafy i drzewa);rozwiązuje zadania maturalne z wykorzystaniem poznanych algorytmów; | pisze program wykonujący konwersję liczb zapisanych w dowolnych podstawach;pisze program realizujący rekurencyjny algorytm zamiany liczby dziesiętnej na postać binarną;rozwiązuje zadania maturalne wykorzystując algorytmy na liczbach;wykorzystuje znajomość algorytmów: : wydawania reszty metodą zachłanną, pakowania plecaka, znajdowania miejsc zerowych funkcji, metody połowienia przedziału, znajdowania najkrótszej drogi (grafy i drzewa) w pisaniu programów;bierze udział w konkursach informatycznych i/lub olimpiadzie informatycznej |