

WYMAGANIA EDUKACYJNE Z INFORMATYKI – KLASA IV LO

Szkoła: I Liceum Ogólnokształcące w Łosicach (Klasa IV)

Opracował: Jarosław Mitek na podstawie materiałów z Wydawnictwa Nowa Era

Zakres kształcenia: Rozszerzony (2 godziny tygodniowo)

Program / Podręcznik: WSiP (Informatyka. Zakres rozszerzony – przygotowanie do matury z informatyki)

ZASADY OCENIANIA I SKALA WYMAGAŃ (PRZYGOTOWANIE MATURALNE)

Ocena osiągnięć edukacyjnych ucznia odbywa się w ramach pięciu poziomów wymagań:

- **Ocena dopuszczająca (wymagania konieczne):** Uczeń zna formę i strukturę arkusza maturalnego z informatyki; rozwiązuje najprostsze zadania testowe/teoretyczne i częściowe podzadania w arkuszu.
- **Ocena dostateczna (wymagania konieczne + podstawowe):** Uczeń samodzielnie rozwiązuje typowe zadania maturalne (np. proste kwerendy w Accessie, podstawowa analiza w Excelu, czytelna implementacja prostego algorytmu w języku Java).
- **Ocena dobra (wymagania konieczne + podstawowe + rozszerzające):** Uczeń sprawnie łączy wiedzę algorytmiczną ze złożonymi strukturami danych, operuje na plikach tekstowych (wczytywanie/przetwarzanie danych maturalnych) i uzyskuje min. 50-60% punktów z arkuszy próbnych.
- **Ocena bardzo dobra (wymagania od koniecznych do dopełniających):** Uczeń bezbłędnie optymalizuje kod, buduje zaawansowane relacyjne bazy danych i formuły w arkuszu kalkulacyjnym, samodzielnie rozwiązuje pełne arkusze maturalne CKE na poziomie powyżej 75-80%.
- **Ocena celująca (wymagania pełne, w tym wykraczające):** Uczeń rozwiązuje najtrudniejsze zadania maturalne z lat ubiegłych, proponuje optymalne algorytmy o minimalnej złożoności czasowej i pamięciowej, pomaga innym uczniom i osiąga najwyższe wyniki w próbnych egzaminach maturalnych (powyżej 90%).

I. ALGORYTM I PROGRAMOWANIE (JAVA)

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
- opisuje budowę i zasady wykonywania programu komputerowego w wybranym języku Java - używa podstawowych typów danych (int, float/double, string, bool) oraz instrukcji sterujących (if/else, for, while), - wczytuje i wypisuje dane z/do konsoli, - podaje definicje algorytmu i złożoności obliczeniowej.	- implementuje algorytmy wyszukiwania elementów (min, max, wyszukiwanie liniowe), - operuje na jednowymiarowych tablicach/listach i napisach (funkcje operujące na stringach), - wczytuje i zapisuje dane z/do plików tekstowych ('dane.txt', 'wyniki.txt'), - implementuje algorytmy na liczbach całkowitych (rozkład na czynniki pierwsze, badanie pierwszości, NWD/NWW).	- implementuje i porównuje algorytmy sortowania (przez wybieranie, zliczanie, bąbelkowe, szybkie/QuickSort), - wykorzystuje metody numeryczne (metoda wyznaczania miejsc zerowych funkcji – metoda połowienia, wyznaczanie pola – metoda trapezów/prosto kątów), - implementuje wyszukiwanie binarne w ciągu uporządkowanym, - stosuje rekrutację w algorytmach (np. potęgowanie, ciąg Fibonacciego, szyfr Cezara).	- implementuje algorytmy tekstowe i szyfrujące (szyfr przestawieniowy, RSA, wyszukiwanie wzorca w tekście), - wykorzystuje zaawansowane struktury danych (tablice dwuwymiarowe, tablice asocjacyjne, stos, kolejka), - optymalizuje algorytmy pod kątem złożoności czasowej $O(n)$ i pamięciowej, - samodzielnie rozwiązuje złożone Zadanie 4 (programistyczne) z arkusza CKE.	- tworzy bezbłędny, zoptymalizowany kod rozwiązujący zestaw zadań z arkuszy maturalnych z maksymalną liczbą punktów, - stosuje programowanie dynamiczne lub zaawansowaną analizę grafów, jeśli wymaga tego zadanie nietypowe.

II. BAZY DANYCH I SYSTEMY RELACYJNE (MS ACCESS / SQL)

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
- wyjaśnia pojęcia: baza danych, tabela, rekord, pole, klucz podstawowy (główny), - tworzy prostą tabelę w środowisku MS Access lub bazy SQL, - importuje proste dane z plików tekstowych (CSV, TXT).	- definiuje i tworzy relacje między tabelami (1:1, 1:N, N:M), - tworzy podstawowe kwerendy wybierające z filtrowaniem i sortowaniem, - zapisuje proste zapytania w języku SQL (instrukcje `SELECT`, `WHERE`, `ORDER BY`).	- tworzy kwerendy z polami obliczeniowymi oraz funkcjami agregującymi (`SUM`, `AVG`, `COUNT`, `MIN`, `MAX`), - stosuje grupowanie danych w kwerendach / SQL (`GROUP BY`, `HAVING`), - tworzy kwerendy łączące dane z kilku relacyjnych tabel (`INNER JOIN`, `LEFT JOIN`).	- tworzy kwerendy z podzapytaniami i kwerendy krzyżowe w MS Access / SQL, - rozwiązuje w całości i bezbłędnie zadanie maturalne z baz danych (część praktyczna), - analizuje i modyfikuje strukturę bazy danych celem wyeliminowania redundancji (normalizacja).	rozwiązuje najtrudniejsze, wielostopniowe zadania z baz danych z poprzednich lat w CKE w czasie ograniczonego limitu maturalnego.

III. ZAAWANSOWANA ANALIZA DANYCH W ARKUSZU KALKULACYJNYM (MS EXCEL)

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
- wprowadza i formatuje dane w arkuszu kalkulacyjnym, - używa podstawowych formuł arytmetycznych i funkcji (`SUMA`, `ŚREDNIA`),	- importuje duże pliki danych maturalnych (.txt, .csv) do arkusza kalkulacyjnego, - stosuje adresowanie względne, bezwzględne i	wykorzystuje zaawansowane funkcje wyszukiwania i odwołania (`WYSZUKAJ.PI ONOWO`, `INDEKS`, `PODAJ.POZYC JE`),	- tworzy złożone formuły tablicowe i zagnieżdżone warunki logiczne rozwiązujące problem maturalny, - dobiera odpowiednie	optymalizuje arkusz kalkulacyjny pod kątem wydajności przeliczeń i czytelności prezentowanych wyników dla zadań

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
tworzy prosty wykres na podstawie podanych danych.	mieszane, używa funkcji logicznych ('JEŻELI', 'ORAZ', 'LUB') oraz statystycznych ('LICZ.JEŻELI', 'SUMA.JEŻELI')	- tworzy tabele przestawne do szybkiej analizy wielowymiarowej - modeluje proste algorytmy iteracyjne i symulacje bezpośrednio w komórkach arkusza.	metody wizualizacji i tworzy czytelne wykresy dwuosiowe / specjalistyczne, samodzielnie i bezbłędnie rozwiązuje pełne zadanie z arkusza kalkulacyjnego CKE.	wykraczających poza standardowy schemat.

IV. TEORIA INFORMATYKI I SYSTEMY KOMPUTEROWE (CZEŚĆ TEORETYCZNA CKE)

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
- przelicza liczby między systemem dziesiętnym a dwójkowym (binarnym), - wymienia podstawowe elementy architektury komputera i sieci komputerowych, - rozpoznaje reprezentację danych w komputerze (bity, bajty).	- dokonuje konwersji między systemami: binarnym, ósemkowym, dziesiętnym i szesnastkowym, - wyjaśnia działanie podstawowych bramek logicznych (AND, OR, NOT, XOR), - wyjaśnia zasady reprezentacji	- analizuje algorytmy przedstawione w postaci schematów blokowych i pseudokodu (Zadanie 1 i 2 na maturze), - wyznacza wynik działania podanego algorytmu dla określonych danych wejściowych, - wyjaśnia	- wyznacza złożoność obliczeniową i liczbę operacji dominujących dla podanego pseudokodu, - wykrywa błędy w pseudokodzie i proponuje poprawki, - rozwiązuje bezbłędnie zadania z części teoretycznej (część I arkusza CKE).	przeprowadza formalne dowody poprawności algorytmów oraz uzasadnia ich optymalność na poziomie akademickim/olimpijskim.

Konieczne (dopuszczający)	Podstawowe (dostateczny)	Rozszerzające (dobry)	Dopełniające (bardzo dobry)	Wykraczające (celujący)
	liczb ujemnych (U2) i stałoprzecinkow ych.	zagadnienia kompresji danych, szyfrowania z kluczem publicznym i prywatnym oraz pojęcie podpisu cyfrowego.		