

**PRZEDMIOTOWY SYSTEM OCENIANIA**  
**Z INFORMATYKI**  
**DLA I LICEUM OGÓLNOKSZTAŁCĄCEGO**  
**im. Ks. Elżbiety w Szczecinku**

**opracowanie : Dariusz Nyk**

## **Zasady oceniania:**

1. Uczeń oceniany jest zgodnie z przyjętymi wymaganiami w myśl zasad sprawiedliwości
2. Ocenie podlegają
  - Praca na lekcji
    - Ćwiczenia praktyczne
    - Odpowiedzi ustne
    - Jakość pracy i aktywność na lekcji
    - Współpraca w grupie
  - Sprawdziany wiadomości i umiejętności po każdej zrealizowanej partii materiału
  - Praca samodzielna ucznia w domu
3. Osiągnięcia w konkursach szkolnych i międzyszkolnych.
4. Jeżeli uczeń opuścił pracę klasową z przyczyn losowych, to powinien napisać ją w ciągu dwóch tygodni od dnia powrotu do szkoły (poza swoimi lekcjami).
5. Uczeń jest zobowiązany poprawić ocenę niedostateczną z pracy klasowej w ciągu 2 tygodni od dnia oddania sprawdzonych prac. Może ją poprawić poza swoimi lekcjami. Ocena z poprawy, bez względu na jej wartość, wpisywana jest do dziennika obok poprawianej oceny
6. Uczeń, który nie poprawił oceny w wyznaczonym terminie lub uzyskał z poprawy ocenę niedostateczną traci prawo do następnych poprawek tej pracy.
7. Uczeń, który unika pisania pracy klasowej (sprawdzianu) lub jej poprawy, nie przychodzi w określone dni lub mimo obecności w szkole odmawia napisania tej pracy otrzymuje stopień niedostateczny.
8. Uczeń obecny na lekcji, odmawiający odpowiedzi ustnej, pisemnej, kartkówki, sprawdzianu itp. otrzymuje ocenę niedostateczną.
9. Uczniowie nieobecni na kartkówkach piszą je w możliwie najbliższym terminie.
10. Nie ma możliwości poprawiania ocen na tydzień przed klasyfikacją.
11. Każdy uczeń ma prawo do zaliczenia mu dodatkowych ocen za wykonane prace nadobowiązkowe o tematyce uzgodnionej i zaaprobowanej przez nauczyciela.
12. Korzystanie przez ucznia w czasie sprawdzianów, kartkówek, odpowiedzi ustnych i innych form sprawdzania wiedzy z niedozwolonych przez nauczyciela pomocy stanowi podstawę do wystawienia oceny niedostatecznej.
13. Jeżeli uczeń nie przestrzega regulaminu pracowni szkolnej może otrzymać ocenę niedostateczną za jego nieznaną (rubryka aktywność).

## I. Algorytmika

Ocena	Wymagania na ocenę
<b>dopuszczająca</b>	<ul style="list-style-type: none"><li>• Uczeń potrafi podać przykłady algorytmów z różnych dziedzin(przedmiotów szkolnych) i z życia praktycznego,</li><li>• definiuje proste sytuacje problemowe przez wyodrębnienie danych</li><li>• opisuje i analizuje działanie prostych algorytmów,</li><li>• zna sposoby zapisu algorytmów.</li><li>• zna klasyczne algorytmy : liniowe przeszukiwanie ciągu w poszukiwaniu wyróżnionego elementu , porządkowanie ciągu elementów (sortowanie), metoda „dziel i zwyciężaj”</li></ul>
<b>dostateczna</b>	<ul style="list-style-type: none"><li>• Uczeń potrafi omówić podstawowe elementy budowy algorytmów,</li><li>• definiuje sytuacje problemowe przez wyodrębnienie danych i wskazanie celu, wskazuje dodatkowe warunki, które musi spełnić rozwiązanie,</li><li>• umie zamienić słowny opis algorytmu na bardziej sformalizowany zapis</li><li>• zna klasyczne algorytmy : algorytmy z rozgałęzieniami , algorytmy rekurencyjne(np. algorytm Euklidesa, znajdowanie liczb Fibonacciego), schemat Hornera , algorytmy numeryczne.</li></ul>
<b>dobra</b>	<ul style="list-style-type: none"><li>• Uczeń zna metody algorytmiczne(rekurencje, iteracje),</li><li>• stosuje wybrane metody do rozwiązywania problemu,</li><li>• zna własności algorytmu,</li><li>• bada zgodność działania algorytmu ze specyfikacją zadania rozwiązywanego przez ten algorytm.</li></ul>
<b>bardzo dobra</b>	<ul style="list-style-type: none"><li>• Uczeń potrafi oceniać własności algorytmu: poprawność, skończoność, złożoność,</li><li>• potrafi przedstawić algorytm za pomocą sformalizowanych sposobów zapisu,</li><li>• rozwiązuje złożone algorytmy przy pomocy rozbijania go na fragmenty.</li></ul>
<b>celująca</b>	<ul style="list-style-type: none"><li>• Uczeń rozwiązuje algorytmy korzystając ze strukturalizacji problemu,</li><li>• stosuje rekurencje przy rozwiązywaniu problemów</li></ul>

## II. Bazy danych

Ocena	Wymagania na ocenę
<b>dopuszczająca</b>	<ul style="list-style-type: none"><li>• Uczeń umie opisywać sytuacje, w których skorzystamy z bazy danych,</li><li>• potrafi zdefiniować prostą bazę danych przy pomocy kreatora</li><li>• korzysta z gotowych obiektów bazy danych</li><li>• zna def. bazy danych i SZBD</li><li>• sortuje dane w bazie</li><li>• zna typy danych</li><li>• rozumie pojęcie klucza</li></ul>

<b>dostateczna</b>	<ul style="list-style-type: none"> <li>• Uczeń umie wprowadzać, aktualizować, przeglądać i wyszukiwać informacje w założonej przez siebie bazie danych,</li> <li>• definiuje prostą strukturę bazy danych</li> <li>• stosuje filtrowanie danych w bazie</li> <li>• potrafi utworzyć autoformularz, autoraport</li> <li>• zna rodzaje baz danych i relacji</li> </ul>
<b>dobra</b>	<ul style="list-style-type: none"> <li>• Uczeń potrafi opracować formularz do wprowadzania danych,</li> <li>• wyszukuje dane posługując się zapytaniem(kwerendami),</li> <li>• porządkuje dane.</li> <li>• Potrafi korzystać z przybornika i formantów</li> <li>• Stosuje kreatora odnośników</li> <li>• Tworzy raport i kwerendę</li> <li>• Korzysta z makr</li> </ul>
<b>bardzo dobra</b>	<ul style="list-style-type: none"> <li>• Uczeń potrafi zdefiniować strukturę relacyjnej bazy danych</li> <li>• modyfikuje formularz, stosuje formanty.</li> <li>• tworzy podformularze</li> <li>• stosuje bazę do korespondencji seryjnej</li> <li>• samodzielnie tworzy makra</li> </ul>
<b>celująca</b>	<ul style="list-style-type: none"> <li>• Modyfikuje makra w Visual Basic</li> </ul>

### **III. Metody i sposoby reprezentacji informacji**

<b>Ocena</b>	<b>Wymagania na ocenę</b>
<b>dopuszczająca</b>	<ul style="list-style-type: none"> <li>• Zna pojęcie systemu pozycyjnego</li> <li>• Zna system dwójkowy, ósemkowy i szesnastkowy</li> <li>• Wie na czym polega kompresja danych</li> <li>• Rozumie pojęcie szyfrowania</li> <li>• Zna znaczenie kodu ASCII</li> <li>• Określa dziedziny zastosowania kryptografii i szyfrowania</li> </ul>
<b>dostateczna</b>	<ul style="list-style-type: none"> <li>• Zamienia liczby z systemu dwójkowego na dziesiętny i odwrotnie</li> <li>• Zamienia z systemu ósemkowego na dwójkowy i odwrotnie</li> <li>• Zna reprezentacje znaków drukarskich, obrazów, dźwięków, animacji</li> <li>• Rozumie pojęcie kryptografii i kryptoanalizy</li> </ul>
<b>dobra</b>	<ul style="list-style-type: none"> <li>• Zamienia liczby z dowolnego systemu na dowolny</li> <li>• Zna reprezentację binarną liczb niecałkowitych i ujemnych</li> <li>• Zna algorytm Huffmana</li> <li>• Stosuje algorytmy szyfrowania przez podstawianie i przestawianie</li> <li>• Zna pojęcie podpisu elektronicznego</li> </ul>
<b>bardzo dobra</b>	<ul style="list-style-type: none"> <li>• Potrafi zaszyfrować i odszyfrować tekst stosując poznane metody szyfrowania i kryptografii i kryptoanalizy</li> <li>• Stosuje kryptografię z jawnym kluczem</li> </ul>
<b>celująca</b>	<ul style="list-style-type: none"> <li>• Stosuje szyfrowanie z kluczem prywatnym</li> <li>• Potrafi zastosować własny sposób szyfrowania</li> </ul>

#### IV. Języki programowania

Ocena	Wymagania na ocenę
<b>dopuszczająca</b>	<ul style="list-style-type: none"><li>• Uczeń rozumie pojęcie języków programowania,</li><li>• zna schemat rozwiązywania problemu,</li><li>• zna proste struktury danych, zna strukturę programu,</li><li>• stosuje proste polecenia wprowadzania i wyprowadzania danych,</li><li>• zna podstawowe polecenia języka,</li><li>• rozróżnia zmienną od stałej.</li><li>• Uczeń rozumie pojęcie programowania strukturalnego.</li></ul>
<b>dostateczna</b>	<ul style="list-style-type: none"><li>• Uczeń stosuje proste operacje arytmetyczne, logiczne na danych,</li><li>• pisze programy z wykorzystaniem poznanych poleceń(decyzyjnych, przypisania) języka,</li><li>• zna pojęcie funkcji i procedur.</li></ul>
<b>dobra</b>	<ul style="list-style-type: none"><li>• Uczeń zna i stosuje złożone struktury danych,</li><li>• zna i stosuje polecenia iteracyjne,</li><li>• stosuje zagnieżdżenia,</li><li>• rozwiązuje problemy przy pomocy strukturalizacji problemu,</li><li>• zna budowę modułów</li><li>• potrafi zapisać prosty algorytm w postaci programu wykonalnego</li></ul>
<b>bardzo dobra</b>	<ul style="list-style-type: none"><li>• Uczeń pisze programy z wykorzystaniem złożonych struktur danych,</li><li>• stosuje różne dostępne moduły</li><li>• pisze programy graficzne,</li><li>• stosuje pliki jako źródło danych i miejsce przechowywania danych</li><li>• stosuje rekurencję przy rozwiązywaniu złożonych problemów.</li><li>• potrafi zapisać złożone algorytmy w postaci programu wykonalnego</li></ul>
<b>celująca</b>	<ul style="list-style-type: none"><li>• Uczeń zna i stosuje programowanie obiektowe, buduje własne moduły i wykorzystuje je.</li><li>• Uczeń zna i stosuje dwa języki programowania.</li></ul>

Ustala się następujący tryb i warunki uzyskania wyższej niż przewidywana oceny rocznej (semestralnej).

Możliwe jest to w przypadku gdy:

- a) uczeń systematycznie uczęszczał na zajęcia z informatyki (nie ma godzin nieusprawiedliwionej nieobecności);
- b) aktywnie uczestniczył w lekcjach
- c) starannie prowadził zeszyt przedmiotowy
- d) systematycznie odrabiał prace domowe
- e) przewidywana ocena roczna(semestralna) jest o jeden stopień niższa od oczekiwanej

Jeżeli uczeń spełnia powyższe warunki nauczyciel może wówczas ustalić dodatkowy termin w celu ponownego sprawdzenia wiadomości ucznia w formie pisemnej lub ustnej.