

2009

Zespół Szkół Społecznych
STO

Krzysztof Kadowski

[PROGRAM NAUCZANIA PRZEDMIOTÓW INFORMATYCZNYCH]

•Technologia informacyjna. •Systemy operacyjne. •Programowanie strukturalne.
•Programowanie obiektowe. •Algorytmy i struktury danych. •Multimedia i grafika
komputerowa. •Sieci komputerowe. •Bazy danych.

1. Wstęp – charakterystyka programu

Szybkie tempo rozwoju dziedziny nauki jaką jest informatyka powoduje, że coraz więcej informacji i treści musi być przekazywanych uczniowi w celu przygotowania go do pracy z nowymi technologiami. Bieżące aktualizowanie i modyfikowanie zagadnień powoduje ciągle rozszerzanie zbioru umiejętności potrzebnych uczniowie do łatwego wdrożenia się w naukę na uczelni wyższej.

Specyfika szkoły społecznej oraz możliwość zwiększenia liczby godzin w siatce godzin na dodatkowe przedmioty powoduje, że niniejszy program nauczania posiada budowę modułową. Każdy moduł jest oddzielnym przedmiotem realizowanym w różnej ilości godzin w cyklu zależnej od rozległości poruszanej tematyki. Nastąpił tu podział na poszczególne przedmioty, które częściowo pokrywają się z podziałem treści na uczelniach wyższych. W niniejszym programie zostały zawarte cele i treści z podstawy programowej przedmiotu informatyka rozszerzone o dodatkowe wiadomości z zakresu systemów operacyjnych, algorytmiki, programowania strukturalnego i obiektowego, multimediów, sieci komputerowych oraz baz danych. Podyktowane jest to chęcią zgłaszających się do tego typu klasy uczniów do przygotowania się na studia informatyczne różnych specjalności.

Bardzo ważnym elementem programu jest nacisk na samodzielne rozwiązywanie problemów przez uczniów poprzez uczestnictwo w warsztatach i projektach realizowanych wspólnie z patronami klasy: firmą Studio 55 i Wydziałem Elektroniki i Informatyki Politechniki Koszalińskiej.

Ważnym elementem programu jest również potrzeba korzystania przez uczniów nie tylko z podręczników na poziomie szkoły ponadgimnazjalnym ale również z literatury fachowej i materiałów w formie skryptów oraz materiałów przygotowywanych na platformę e-learningową klasy. Platforma ta stanowi w tym wypadku źródło wiedzy oraz narzędzie do kontroli i sprawdzania postępów ucznia.

Uczestnictwo uczniów w projektach ma dodatkowo wdrażać ich do pracy zespołowej, która jest obecnie szeroko stosowaną formą pracy nad projektami informatycznymi również w przemyśle. Praca w grupie spełnia dodatkową rolę – uświadamia uczniowi, że jego stopień zaangażowania i praca mają wpływ na wyniki całego projektu.

Korelacje między przedmiotami informatycznymi i innymi przedmiotami takimi jak matematyka czy fizyka powodują, że uczeń posiada świadomość iż rozwiązywanie faktycznych i rzeczywistych problemów za pomocą środków informatycznych nie tylko utrwała jego wiedzę ale przygotowuje do późniejszej działalności i pracy nad zadaniami, które może spotkać w firmie czy na uczelni wyższej.

W niniejszym programie położono nacisk na treści wskazane w standardach wymagań egzaminacyjnych, jako że ich realizacja zadecyduje o powodzeniu uczniów na egzaminie maturalnym.

2. Przedmioty wchodzące w skład programu i proponowany przydział godzin

Tabela 1 Tygodniowy przydział godzin przedmiotów informatycznych

Przedmiot	Skrót	Semestr					
		I	II	III	IV	V	VI
Technologia informacyjna	IT	1	2				
Systemy operacyjne	OS	2	1				
Algorytmy i struktury danych	ASD		1	1	1		
Programowanie strukturalne	PS	2	1				
Programowanie obiektowe	PO			2	2		
Multimedia i grafika komputerowa	MGK			1	1		
Sieci komputerowe	SK			1	1	1	
Bazy danych	BD					2	2
Warsztaty specjalistyczne: programowanie lub grafika	W		1*	1*	1*	1	2
Razem		5	4	5	5	4	4

* przedmioty realizowane w formie zajęć dodatkowych warsztatów (nie wliczają się w siatkę godzin)

3. Podstawa programowa

Poniżej cytowana jest podstawa programowa dla liceów ogólnokształcących.

Cele edukacyjne

1. Przygotowanie do Świadomego wyboru kierunku i zakresu dalszego kształcenia informatycznego.
2. Zdolność do samodzielnego korzystania z komputera dla realizacji części zadań edukacyjnych oraz innych celów poznawczych.

Zadania szkoły

1. Stworzenie warunków do poznania wybranych zagadnień, pojęć i metod informatyki jako dyscypliny naukowej oraz jej najważniejszych zastosowań.
2. Kształcenie samodzielności intelektualnej, odpowiedzialności za własny rozwój, gotowości do podejmowania i rozwiązywania złożonych zadań, z uwzględnieniem środków i metod informatyki.
3. Rozwijanie umiejętności pracy zespołowej przez realizacji projektów grupowych.

Treści nauczania

1. Algorytmika i programowanie:
 - a) metodyczna analiza i modelowanie umiarkowanie złożonych problemów i procesów z różnych dziedzin;
 - b) przegląd algorytmów klasycznych;
 - c) wybrane techniki projektowania algorytmów i struktur danych: programowanie strukturalne,
 - d) zastępujące, abstrakcja danych, metoda kolejnych uściśleń;
 - e) elementy analizy algorytmów;
 - f) indywidualna i zespołowa realizacja projektów programistycznych w wybranym języku wysokiego poziomu.
2. Bazy danych:
 - a) podstawowe formy organizacji informacji w bazach danych,
 - b) budowa relacyjnych baz danych,
 - c) wyszukiwanie informacji w relacyjnych bazach danych z użyciem języka zapytań,
 - d) projektowanie prostych relacyjnych baz danych.
3. Multimedia. Sieci komputerowe:
 - a) sprawne i świadome korzystanie z multimediów i tworzenie własnych materiałów multimedialnych,
 - b) przetwarzanie informacji w różnej postaci (w tym wizualnej i dźwiękowej),
 - c) budowa i działanie sieci komputerowych,
 - d) tworzenie i publikowanie własnych materiałów w sieci.
4. Tendencje w rozwoju informatyki i jej zastosowań.

Osiągnięcia

1. Formułowanie sytuacji problemowej, jej modelowanie i rozwiązywanie z użyciem metod informatycznych.
2. Ocenianie poprawności i efektywności rozwiązań i ich testowanie. Tworzenie dokumentów rozwiązań.
3. Wyszukiwanie informacji w bazach danych i projektowanie prostych baz danych.
4. Tworzenie opracowań multimedialnych.
5. Sprawne korzystanie z usług sieci komputerowych w pracy z informacjami swoimi i obcymi.
6. Planowanie pracy i nadzór nad przebiegiem wykonywania projektów realizowanych zespołowo z wykorzystaniem programów komputerowych.

4. Szczegółowe cele edukacyjne i wychowawcze

Na lekcjach przedmiotów informatycznych cele edukacyjne i wychowawcze powinny być realizowane spójnie. Uczeń powinien być zapoznany z etycznymi podstawami użytkowania komputerów i oprogramowania oraz aspektami poszanowania praw autorskich.

4.1 Cele edukacyjne

4.1.1 Systemy operacyjne

Uczeń powinien:

- wiedzieć czym jest system operacyjny i znać jego funkcje,
- znać podział na systemy serwerowe, biurkowe, wieloprocesorowe, rozproszone, kieszonkowe oraz znać ich właściwości,
- znać struktury systemów operacyjnych i komputerowych,
- umieć zarządzać procesami w różnych systemach operacyjnych: Windows, Linux/Unix.
- umieć zarządzać pamięcią w systemach i znać pojęcia przydziału pamięci, stronicowania, pamięci wirtualnej, metody dostępu do plików,
- znać różne systemy plików i ich właściwości,
- znać sprzęty i interfejsy I/O,
- znać strukturę pamięci masowych i metody pracy z nimi,
- znać topologie i typy systemów rozproszonych,
- znać metody zabezpieczania systemów, wykrywania włamań i kryptograficzne metody ochrony danych, umieć tworzyć kopie zapasowe danych,
- umieć wydawać komendy i sterować systemami z rodziny Windows i Unix,

4.1.2 Programowanie strukturalne

Uczeń powinien:

- znać pojęcie, właściwości i formy zapisu algorytmów,
- znać i stosować pojęcia optymalizacji, złożoności obliczeniowej, czasowej i pesymistycznej algorytmu,
- znać reprezentacje danych w komputerze,
- znać typy i podział języków programowania i kompilatorów,
- znać i stosować różne typy danych,
- znać instrukcje sterujące, warunkowe, wyboru i pętli w wybranym języku,
- umieć definiować funkcje, stosować różny zasięg zmiennych i metody przekazywania parametrów funkcji,
- umieć stosować iterację w podstawowych algorytmach iteracyjnych,
- umieć stosować typy tablicowe w rozwiązywaniu różnych problemów,
- znać pojęcie i umieć stosować rekurencję w wybranych algorytmach,
- umieć wykorzystywać struktury i tablice typu strukturalnego,
- umieć stosować typy wskaźnikowe, przekazywać parametry funkcji przez wskaźnik i stosować zmienne i struktury dynamiczne,
- umieć stosować funkcje I/O z pliku.

4.1.3 Programowanie obiektowe

Uczeń powinien:

- znać różnice między programowaniem strukturalnym i obiektowym,
- znać pojęcia obiektu, klasy, instancji,
- znać pojęcia hermetyzacji danych, dziedziczenia i polimorfizmu,
- znać typy, operatory i instrukcje sterujące,
- znać i stosować funkcje i moduły,
- umieć stosować typy: łańcuchy znaków, kolekcje, tablice,
- umieć zarządzać pamięcią za pomocą operatorów *new* i *delete*,
- tworzyć klasy, strumienie I/O, klasy zaprzyjaźnione i inne typy,
- znać typy i obsługę wyjątków w klasach,
- umieć tworzyć klasy dziedziczące i znać mechanizmy dziedziczenia pojedynczego, wielokrotnego i wielobazowego,
- znać i umieć stosować szablony,
- znać i stosować klasy strumieni,
- znać i stosować różne metody dostępu do plików,
- umieć przeprowadzać instalację środowiska języków Builder C++ oraz Java,
- umieć stosować funkcje i klasy graficzne w obu językach.

4.1.4 Technologia informacyjna

Wymogi wg. programu nauczania nr DKOS-5002-17/05.

4.1.5 Algorytmy i struktury danych

Uczeń powinien:

- znać pojęcie abstrakcyjnych typów danych,
- znać pojęcie enkapsulacja, dziedziczenie, wskaźniki, polimorfizm,
- umieć stosować szablony z C++,
- znać pojęcia i umieć przeprowadzić analizę złożoności algorytmów,
- znać notację O,
- znać i umieć stosować w wybranych problemach listy jedno-, dwukierunkowe, cykliczne, z przeskokami, samoorganizujące się, tablice rzadkie,
- znać i umieć stosować stosy i kolejki,
- umieć stosować metody rekurencyjne,
- znać i stosować drzewa binarne różnego rodzaju,
- znać i stosować grafy,
- znać i stosować metody sortowania,
- znać i stosować metody kompresji danych,
- znać i stosować podstawowe metody numeryczne w zagadnieniach matematycznych, fizycznych, statystycznych itp.

4.1.6 Sieci komputerowe

Uczeń powinien:

- znać historię rozwoju sieci komputerowych,
- znać sprzęt do budowy sieci komputerowych,
- znać i umieć budować sieci komputerowe wg. różnych topologii,
- znać obszary funkcjonalne różnych typów sieci LAN, MAN, WAN,
- znać metody przyłączania stacji, serwerów do różnych typów sieci,
- znać pojęcie i właściwości warstwy fizycznej sieci,
- znać pojęcie i właściwości warstwy łącza danych,
- znać właściwości różnych rodzajów sieci Ethernet,
- znać budowę ramki różnych rodzajów sieci,
- znać pojęcie protokołu,
- znać właściwości i różnych protokołów sieciowych,
- znać pojęcia, właściwości sprzętu i zasady współdziałania sieci WAN,
- znać budowę i właściwości linii dzierżawionych,
- znać i umieć stosować technologie Dial-Up,
- znać i umieć stosować oprogramowanie do obsługi sieci w różnych systemach operacyjnych,
- znać pojęcia i umieć stosować narzędzia administracyjne w różnych systemach operacyjnych,
- umieć stosować narzędzia do testowania wydajności sieci, poprawności działania i zabezpieczeń,
- znać pojęcia ochrony danych w sieci, procedury instalacji zabezpieczeń sieciowych, techniki konserwacji oraz ochrony sprzętu,
- znać narzędzia i umieć wykonywać kopie zapasowe danych w sieci,
- umieć tworzyć i konfigurować usługi serwera domeny, www, baz danych, plików, wydruku, ftp, aplikacji na różnych systemach danych.

4.1.7 Bazy danych

Uczeń powinien:

- znać różne rodzaje systemów baz danych,
- znać pojęcia normalizacji danych, rekordu, pola,
- znać i umieć stosować schemat budowy relacyjnej bazy danych,
- umieć projektować relacyjne bazy danych,
- znać typy danych w bazach danych,
- umieć tworzyć relacje i łączyć pola tabel,
- umieć organizować dane za pomocą kwerend,
- umieć stosować kwerendy wyszukiwujące, aktualizujące, parametryczne,
- umieć modyfikować dane w bazie,
- umieć eksportować i importować dane,
- umieć stosować zapytania w języku SQL,

- umieć stosować mechanizmy ochrony danych w bazach danych,
- umieć tworzyć interfejsy bazy danych w postaci formularzy,
- umieć budować raporty baz danych,
- umieć budować bazy typu desktop,
- umieć konfigurować bazy klient/serwer,
- umieć tworzyć bazy klient/serwer na różnych platformach,
- znać pojęcia związane z hurtowniami danych,
- znać pojęcia związane z zastosowaniem baz danych w handlu elektronicznym.

4.1.8 Multimedia i grafika komputerowa

Uczeń powinien:

- znać i charakteryzować formaty zapisu grafiki i dźwięku,
- znać różnice między grafiką rastrową i wektorową,
- znać zasady posługiwania się kolorem, planem, światłem,
- znać zasady tworzenia grafiki 2D i 3D w różnych programach,
- znać zasady tworzenia animacji w różnych technikach,
- umieć tworzyć i wykorzystywać grafikę do budowy serwisów www,
- umieć tworzyć grafikę do różnych rodzajów wydruków poligraficznych,
- znać metody projektowania grafiki wielkoformatowej,
- umieć wykorzystywać narzędzia do kalibracji koloru.

4.2 Cele wychowawcze

Cele wychowawcze to:

- wpojenie uczniowi konieczności przestrzegania regulaminów pracowni komputerowych oraz zasad BHP w pracy z urządzeniami komputerowymi,
- wdrożenie ucznia do pracy zespołowej,
- przygotowanie ucznia do świadomego wyboru dalszego zakresu kształcenia,
- wpojenie poszanowania własności intelektualnej oraz przestrzegania praw autorskich,
- kształtowanie umiejętności wyszukiwania i wykorzystania informacji w różnych mediach,
- motywowanie do samokształcenia i planowania samorozwoju,
- motywowanie do poszerzania wiadomości z wykorzystaniem dostępnych narzędzi i zasobów,
- wykazania zagrożeń z stosowania nowoczesnych technologii informacyjnych.

5. Procedury osiągnięcia celów

5.1 Cele edukacyjne

Osiągnięcie celów edukacyjnych zależy przede wszystkim od metod nauczania stosowanych w procesie dydaktycznym. Do większości materiału z zakresu informatyki można zastosować metody projektowe, burzę mózgów, polemikę – czyli metody poszukujące. Część materiału, szczególnie na wstępie i wprowadzeniu do przedmiotu można przeprowadzić metodami podającymi cel nakierowania ucznia do dalszego poszukiwania materiałów. W klasie informatycznej o małej ilości uczniów istnieje możliwość pracy indywidualnej z uczniem opartej na wykonywaniu ćwiczeń prowadzących do realizacji zamierzonych celów edukacyjnych.

Wśród metod podających najważniejszą metodą stosowaną w tej klasie będzie wykład poparty przykładami oraz dodatkowymi informacjami, wskazówkami oraz testami, które do każdej części tematycznej będą przygotowane w formie elektronicznej na platformie e-learningowej klasy. Dodatkowe wskazówki i projekty mają za zadanie wzbudzić u ucznia chęć poszukiwania w celu rozwiązania określonych problemów.

Testy sprawdzające przyswojenie materiału w formie zautomatyzowanych testów różnych typów z automatycznymi wskazówkami do błędów, automatyczną oceną oraz możliwością wglądu ucznia i jego rodzica do wyników powoduje usystematyzowanie procesu oraz wprowadzenie elementów samokontroli własnego rozwoju.

Prac w grupach nad projektami informatycznymi spowoduje wdrożenie ucznia do rozwiązywania rzeczywistych problemów z zastosowaniem różnych technik informatycznych, zmusi go do komunikacji z

uczestnikami grupy i prowadzącym w celu wymiany doświadczeń i pomysłów oraz kontrolowania postępów a co za tym idzie do posługiwania się językiem informatycznym.

5.2 Cele wychowawcze

Prawidłowa konstrukcja procesu edukacyjnego nie tylko pełni rolę w osiągnięciu celów dydaktycznych ale również w wychowaniu. Organizacja pracy ucznia, samoorganizacja rozwoju motywuje ucznia do samodzielnego szukania rozwiązań i prezentacji własnych dokonań. Praca w grupie kształtuje więź emocjonalną a przez to odpowiedzialność za grupę, swoją rolę w grupie i umiejętności dyskusji, podejmowania kompromisów i współpracy.

Ważnym czynnikiem jest poszanowanie własności intelektualnych oraz przestrzeganie praw autorskich zawartych w różnego rodzaju dokumentach i umowach międzynarodowych. Korzystanie z licencjonowanego oprogramowania lub opartego na licencjach otwartych i możliwość wyboru między tymi aplikacjami kształtuje postawy poprawne etycznie.

Ważne jest również wpajanie i egzekwowanie zasad i regulaminów korzystania z urządzeń oraz zasad BHP w celu świadomego i poprawnego użytkowania sprzętów i aplikacji.

6. Wyposażenie pracowni

Pracownia szkolna nie jest jedynym miejscem odbywania przez ucznia zajęć. W klasie informatycznej część zajęć odbywać się będzie w zakładach produkcyjnych oraz laboratoriach uczelni wyższej. Dlatego część sprzętu i oprogramowania użyta w procesie dydaktycznym pochodzić będzie z zewnątrz szkoły. Przede wszystkim będą to urządzenia do kalibracji kolorów, wydruków wielkoformatowych, komputery specjalistyczne i stacje graficzne oparte na systemach Unix i Mac OS, oprogramowanie graficzne takie jak 3DMax, Macromedia Flash, Adobe Photoshop, itp.

W samej szkolnej pracowni komputerowej urządzenia i oprogramowanie potrzebne do przeprowadzenia procesu dydaktycznego wykazane są w tabeli drugiej.

Tabela 2 Oprogramowanie i sprzęt komputerowy w pracowni komputerowej

Przedmiot	Zestaw sprzętu i oprogramowania
Technologia informacyjna	Stanowiska PC, OS Windows i Linux, MS Office, Open Office
Systemy operacyjne	Stanowiska PC z bootladerem dla systemów Windows i Linux
Programowanie strukturalne	Stanowiska PC, kompilatory języka C, środowisko programistyczne Borland Builder C++ wersja 6.0 Personal, GCC dla Linux, Dev C++
Programowanie obiektowe	Stanowiska PC, kompilatory języka C++, środowisko programistyczne Borland Builder C++ wersja 6.0 Personal, GCC dla Linux, Dev C++, Java Software Development Kit 8.0
Algorytmy i struktury danych	Stanowiska PC, środowisko programistyczne Borland Builder C++ 6.0 wersja Personal, Dia,
Sieci komputerowe	Stanowiska PC, OS Windows, OS Linux, dostęp do serwera, stanowiska do budowy serwerów, osprzęt sieciowy: HUB, switch, okablowanie, router, modem
Bazy danych	Stanowiska PC, baza danych typu desktop MS Access, Open Office Kexi, serwer www, serwer baz danych MS SQL Server, MySQL, Postgres, Interbase, Oracle.
Multimedia i grafika komputerowa	Stanowiska PC i Mac. Oprogramowanie Corel Draw, Adobe Photoshop, GIMP, Macromedia Flash, Dia, Nvu, MS Power Point, Open Office Impress

7. Treści nauczania

7.1 Technologia informacyjna

Wg. Programu nauczania nr DKOS-5002-17/05.

1. Lekcja wprowadzająca. BHP pracy z urządzeniami w pracowni informatycznej.
2. Omówienie tematyki zajęć TI.
3. Rozwój technologii informatycznych w życiu człowieka.
4. Czy komputer może stać się wirtualnym biurkiem?
5. Czy program komputerowy jest narzędziem? Licencjonowanie oprogramowania i prawa autorskie.
6. Jak edytować tekst? Wprowadzenie do edytorów tekstu.
 - i. Układ dokumentu i drukowanie.
 - ii. Sekcje. Nagłówki i stopki dokumentu. Formatowanie numerów stron. Podgląd i drukowanie dokumentu.
 - iii. Edytowanie dokumentu i sprawdzanie pisowni. Sprawdzanie pisowni i gramatyki. Dzielenie wyrazów, zmiana języka słownika.
 - iv. Zmiana wyglądu tekstu i praca z dokumentem długim.
 - v. Nagłówki rozdziałów, style obiektów. Praca z konspektem. Formatowanie stylów.
 - vi. Operacje na blokach tekstu i obiektach. Zarządzanie dokumentami głównymi i podrzędnymi. Wstawianie zakładki i odsyłaczy.
 - vii. Generowanie i modyfikowanie spisu treści. Tworzenie i modyfikowanie indeksu. Przypisy. Autopodsumowanie dokumentu.
 - viii. Prezentowanie informacji w tabelach i kolumnach. Formatowanie tabel. Operacje na danych w tabelach.
 - ix. Wykorzystanie grafiki, wykresów oraz edycja wzorów.
 - x. Tworzenie diagramów. Wstawianie i modyfikowanie rysunku. Położenie rysunku względem otaczającego go tekstu. Tworzenie obiektów WordArt.
 - xi. Rysowanie i modyfikowanie kształtów. Prosta obróbka grafiki. Podpisywanie grafiki. Wstawianie wykresu. Modyfikowanie wyglądu wykresu.
 - xii. Importowanie danych do wykresu. Edycja wzorów.
 - xiii. Współpraca w grupie. Śledzenie zmian w dokumencie i zarządzanie nimi. Wyświetlanie i przeglądanie komentarzy. Zabezpieczanie dokumentu hasłem.
 - xiv. Tworzenie listów seryjnych i etykiet. Korespondencja seryjna. Przygotowanie danych zmiennych listu seryjnego.

7. Jak używać arkusza kalkulacyjnego?

- i. Opcje arkusza. Tworzenie i nazywanie zakresów komórek. Tworzenie etykiet. Formaty komórek. Tworzenie własnych formatów. Tworzenie stylów.
- ii. Adresowanie względne i bezwzględne. Tworzenie list danych. Dodawanie serii wartości.
- iii. Stosowanie formatowania warunkowego. Podsumowywanie danych za pomocą sum częściowych.
- iv. Filtrowanie list danych. Tworzenie filtrów zaawansowanych. Sprawdzanie poprawności danych.
- v. Dodawanie funkcji. Edycja formuł. Szacowanie formuł. Tworzenie prostych formuł warunkowych.
- vi. Łączenie danych. Importowanie danych z plików. Tworzenie kwerend sieci. Import tabel baz danych.
- vii. Funkcje logiczne: jeżeli, oraz, lub.
- viii. Funkcje daty i czasu: czas, data, dni.360, dzień, dzień.tyg, dziś, edate, miesiąc, rok, weeknum, workday.
- ix. Funkcje matematyczne, statystyczne i informacyjne.
- x. Sporządzanie wykresów danych.
- xi. Tabele przestawne.
- xii. Wprowadzenie do makr. Rejestrowanie makra. Uruchamianie makr.

8. Jak tworzyć i obrabiać obrazy?

- i. Podstawowe pojęcia: format plików graficznych, rozszerzenia, kompresja, grafika rastrowa i wektorowa.

- ii. Tworzenie grafiki wektorowej.
 - iii. Kreślenie krzywych i figur 2D.
 - iv. Efekty w grafice wektorowej.
 - v. Elementy grafiki 3D.
 - vi. Grafika rastrowa – podstawy.
 - vii. Tworzenie obrazów, pędzle i efekty.
 - viii. Filtry w grafice rastrowej.
 - ix. Obróbka zdjęć. Retusz.
9. Podstawy sieci komputerowych.
- i. Architektura sieci.
 - ii. Urządzenia sieciowe.
 - iii. Serwer – zadania i funkcje.
 - iv. Bezpieczeństwo w sieci.
10. Jak stworzyć prezentację multimedialną?
- i. Zadania i zasady tworzenia prezentacji.
 - ii. Wstawianie tła, dobór czcionki i grafiki.
 - iii. Efekty przejść.
 - iv. Elementy formatowania tekstu.
 - v. Dźwięk i film w prezentacji.
 - vi. Przebieg prezentacji.
11. Jak gromadzić dane w bazie danych?
- i. Czym jest baza danych?
 - ii. Pola, rekordy, tabele.
 - iii. Podział baz danych.
 - iv. Relacje w bazach danych.
 - v. Tworzenie tabel.
 - vi. Tworzenie formularzy do baz danych.
 - vii. SQL – język zapytań.
 - viii. Tworzenie raportów.
 - ix. Automatyzacja pracy z bazą danych.
12. Moja strona WWW.
- i. Zasada działania serwera WWW.
 - ii. Tworzenie prostej strony WWW.
 - iii. Formatowanie tekstu i grafiki na stronie.
 - iv. Podstawy języka HTML.
 - v. Arkusze stylów.
 - vi. Odnośniki na stronie.

7.2 Systemy operacyjne

1. Podstawy systemów operacyjnych
 - i. Co to jest system operacyjny?
 - ii. Systemy serwerowe.
 - iii. Systemy biurkowe.
 - iv. Systemy wieloprocessorowe.
 - v. Systemy rozproszone.
 - vi. Systemy kieszonkowe.
2. Struktury systemów komputerowych
 - i. Działanie systemu komputerowego.
 - ii. Struktura I/O.
 - iii. Struktura i hierarchia pamięci.
 - iv. Ochrona sprzętowa.
3. Struktura systemów operacyjnych (OS)
 - i. Składowe systemu.
 - ii. Usługi OS.
 - iii. Wywołania systemowe.
 - iv. Programy OS.
 - v. Struktura OS.
 - vi. Maszyny wirtualne.
4. Zarządzanie procesami
 - i. Procesy. Koncepcja i planowanie.
 - ii. Działania na procesach i współdziałanie.
 - iii. Komunikacja między procesami.
 - iv. Modele wielowątkowości.
 - v. Wątki w różnych OS.
 - vi. Planowanie przydziału procesora.
 - vii. Algorytmy planowania procesora.
 - viii. synchronizacja procesów.
5. Zarządzanie pamięcią
 - i. Podstawy.
 - ii. Przydział pamięci.
 - iii. Stronicowanie.
 - iv. Segmentacja.
 - v. Pamięć wirtualna.
 - vi. Pojęcie pliku.
 - vii. Metody dostępu do pliku.
 - viii. Katalogi i ich struktura.
 - ix. Montowanie systemu plików.
 - x. Budowa systemów plików.

- xi. Zarządzanie przestrzenią, efektywność i wydajność.
- xii. Przykładowe systemy plików: FAT, FAT32, NTFS, NFS, EXT2, EXT3.

6. Systemy I/O

- i. Sprzęt I/O.
- ii. Interfejsy I/O.
- iii. Strumienie.
- iv. Struktura pamięci masowych.
- v. Zarządzanie obszarem dysku i obszarem wymiany.
- vi. Podłączanie i montowanie dysków.

7. Systemy rozproszone

- i. Topologia.
- ii. Typy sieci.
- iii. Protokoły komunikacyjne.
- iv. Projekty sieci.
- v. Zdalny dostęp do plików.
- vi. Porządkowanie zdarzeń.
- vii. Wykluczanie.

8. Bezpieczeństwo

- i. Uwierzytelnianie.
- ii. Zagrożenia programowe i systemowe.
- iii. Wykrywanie włamań.
- iv. Kryptografia.
- v. Poziomy bezpieczeństwa komputerowego.

9. Przykładowe OS

- i. Windows XP
 - Elementy systemu.
 - Podsystemy środowiskowe.
 - System plików.
 - Praca w sieci.
 - Interface.
- ii. Linux / Unix
 - Moduły jądra.
 - Zarządzanie procesami.
 - Zarządzanie pamięcią.
 - System I/O.
 - Komunikacja międzyprocesowa.
 - Struktura sieci.
 - Bezpieczeństwo.
 - Komendy.
 - Skrypty.

7.3 Programowanie strukturalne

1. Wstęp do programowania - algorytmy

- i. Algorytm – definicja, formy zapisu, właściwości.
- ii. Iteracje.
- iii. Poprawność algorytmu.
- iv. Złożoność obliczeniowa algorytmu.
- v. Podstawowe algorytmy.

2. Wstęp do programowania w C

- i. Reprezentacja danych w komputerze.
- ii. Zmienne i wyrażenia.
- iii. Proces wytwarzania programu. Kompilatory.
- iv. Podstawowe typy danych.
- v. Instrukcje sterujące.
- vi. Instrukcja warunkowa if..else.
- vii. Instrukcja wyboru switch.
- viii. Instrukcje pętli: for, while, do...while, break, continue.

3. Funkcje

- i. Definiowanie i wywołanie funkcji.
- ii. Zakres zmiennych.
- iii. Metody przekazywania parametrów funkcji.

4. Podstawowe algorytmy iteracyjne

- i. Znajdowanie najmniejszego / największego elementu w ciągu.
- ii. Liczba pierwsza.
- iii. NWD i NWW (algorytm Euklidesa).
- iv. Obliczanie pierwiastka kwadratowego (metoda Newtona-Raphosa).
- v. Obliczanie pola obszaru ograniczonego wykresem (metody całkowania numerycznego).
- vi. Znajdowanie przybliżonej wartości miejsca zerowego funkcji.
- vii. Metody Monte Carlo.
- viii. Obliczanie wartości liczby π .
- ix. Modelowanie ruchów Browna.

5. Typ tablicowy

- i. Charakterystyka tablic jednowymiarowych i wielowymiarowych.
- ii. Metody przeszukiwania tablic.
- iii. Zamiana systemów liczbowych.
- iv. Sito Eratostenesa.
- v. Metody sortowania: bąbelkowa, przez wstawiania, przez wybór.
- vi. Szyfr Cezara.

6. Rekurencja

- i. Funkcje rekurencyjne.
- ii. Silnia.
- iii. Potęga liczby naturalnej.
- iv. Wartości ciągów rekurencyjnych.
- v. Ciąg Fibonacciego.
- vi. Schemat Hornera.
- vii. Wieże Hanoi.
- viii. Metoda „Dziel i zwyciężaj”.
- ix. Przeszukiwanie binarne.
- x. Sortowanie szybkie i przez scalanie.

7. Struktury

- i. Definicja struktury.
- ii. Tablice o elementach typu strukturalnego.
- iii. Struktury jako argumenty funkcji.

8. Wskaźniki

- i. Deklaracja zmiennych wskaźnikowych.
- ii. Przekazywanie parametrów funkcji przez wskaźnik.
- iii. Wskaźniki w tablicach.

9. Zmienne i struktury dynamiczne

- i. Tablica dynamiczna jedno i wielowymiarowa.
- ii. Lista jednokierunkowa i dwukierunkowa.

10. Praca z plikami

- i. Zapis i odczyt z pliku.
- ii. Przeszukiwanie plików.

7.4 Programowanie obiektowe

1. Wstęp do programowania obiektowego w C++

- i. Język C++.
- ii. C++ jako język programowania obiektowego.
- iii. Obiekty, klasy i instancje.
- iv. Klasy w języku C++.
- v. Hermetyzacja danych.
- vi. Dziedziczenie.
- vii. Polimorfizm.

2. Podstawowe pojęcia języka C++

- i. Typy, operatory i instrukcje sterujące.
- ii. Funkcje i moduły.
- iii. Łańcuchy znaków.
- iv. Kolekcje.
- v. Obsługa wyjątków.
- vi. Wskaźniki, tablice i C-łańcuchy.
- vii. Zarządzanie pamięcią za pomocą operatorów new i delete.

3. Programowanie klas

- i. Pierwsza klasa: Fraction.
- ii. Operatory klas.
- iii. Optymalizacja efektywności kodu.
- iv. Referencje i stałe.
- v. Strumienie wejścia i wyjścia.
- vi. Klasy zaprzyjaźnione i inne typy.
- vii. Obsługa wyjątków w klasach.

4. Dziedziczenie i polimorfizm

- i. Dziedziczenie pojedyncze.
- ii. Funkcje wirtualne.
- iii. Polimorfizm.
- iv. Dziedziczenie wielokrotne.
- v. Pałapki projektowania z użyciem dziedziczenia.

5. Składowe dynamiczne i statyczne

- i. Składowe dynamiczne.
- ii. Inne aspekty składowych dynamicznych.
- iii. Dziedziczenie i klasy o składowych dynamicznych.
- iv. Klasy zawierające klasy.

6. Szablony

- i. Terminologia.
- ii. Szablony klas.

7. Standardowa biblioteka wejścia i wyjścia

- i. Standardowe klasy strumieni.
- ii. Dostęp do plików.

8. Inne właściwości języka

- i. Operacje na wektorach.
- ii. Algorytmy STL.
- iii. Ograniczenia wartości numerycznych.
- iv. Wskaźniki funkcji.
- v. Pliki nagłówkowe w językach C i C++.

9. Wprowadzenie do języka Java

- i. Java jako narzędzie programistyczne.
- ii. Zalety Javy.
- iii. Koncepcje języka Java.
- iv. Java i Internet.

10. Środowisko Java

- i. Instalowanie pakietu Java Software Development Kit.
- ii. Konfiguracja ścieżki dostępu.
- iii. Instalowanie bibliotek i dokumentacji.
- iv. Różne środowiska programistyczne.
- v. Praca ze zintegrowanym środowiskiem programistycznym.
- vi. Kompilowanie i uruchamianie programów przy użyciu edytora tekstu.
- vii. Aplikacje graficzne.

11. Podstawy programowania w Javie

- i. Prosty program napisany w Javie.
 - ii. Typy danych.
 - iii. Operatory.
 - iv. Funkcje i stałe matematyczne.
 - v. Łańcuchy.
 - vi. Formatowanie wyjścia.
 - vii. Instrukcje sterujące.
 - viii. Tablice.
12. Obiekty i klasy
- i. Wprowadzenie do programowania zorientowanego obiektowo.
 - ii. Obiekty.
 - iii. Relacje pomiędzy klasami.
 - iv. Tworzenie własnych klas.
 - v. Pola i metody statyczne.
 - vi. Przeładowanie.
13. Dziedziczenie
- i. Hierarchie dziedziczenia.
 - ii. Polimorfizm.
 - iii. Rzutowanie.
 - iv. Klasy abstrakcyjne.
 - v. Wskaźniki do metod.

7.5 Algorytmy i struktury danych

1. Wstęp do algorytmów i programowania obiektowego

- i. Abstrakcyjne typy danych.
- ii. Enkapsulacja.
- iii. Dziedziczenie.
- iv. Wskaźniki.
- v. Polimorfizm.
- vi. C++ a programowanie obiektowe.
- vii. Standardowa biblioteka szablonów.
- viii. Wektory w standardowej bibliotece szablonów.
- ix. Struktury danych a programowanie obiektowe.
- x. Przykład zastosowania: plik z dostępem swobodnym.

2. Analiza złożoności

- i. Złożoność obliczeniowa i asymptotyczna.
- ii. O-notacja.
- iii. Właściwości O-notacji.
- iv. Notacje.
- v. Możliwe problemy.
- vi. Przykłady złożoności.
- vii. Określanie złożoności asymptotycznej.
- viii. Złożoność optymistyczna, średnia i pesymistyczna.

3. Listy

- i. Listy jednokierunkowe.
- ii. Listy dwukierunkowe.
- iii. Listy cykliczne.

4. Stosy i kolejki

- i. Stosy.
- ii. Kolejki.
- iii. Kolejki priorytetowe.

5. Rekurencja

- i. Definicje rekurencyjne.

- ii. Wywołania funkcji a implementacja rekurencji.
- iii. Anatomia wywołania rekurencyjnego.

6. Drzewa binarne

- i. Drzewa, drzewa binarne i binarne drzewa poszukiwania.
- ii. Implementacja drzew binarnych.
- iii. Wyszukiwanie w drzewie binarnym.
- iv. Przechodzenie po drzewie.
- v. Wstawianie i usuwanie.

7. Drzewa wielokierunkowe

7.1. Rodzina B-drzew.

- i. B-drzewa.
- ii. Drzewa bitowe.

8. Grafy

- i. Reprezentacje grafów.
- ii. Przechodzenie po grafach.
- iii. Najkrótsza droga.
- iv. Problem najkrótszych dróg typu "wszystkie-do-wszystkich".
- v. Wykrywanie cykli.
- vi. Drzewa rozpinające.

9. Sortowanie

- i. Podstawowe algorytmy sortowania: sortowanie przez wstawianie, przez wybieranie, bąbelkowe.
- ii. Drzewa decyzyjne.
- iii. Efektywne algorytmy sortowania: sortowanie Stella, przez kopcowanie, szybkie (quicksort), poprzez scalanie, pozycyjne.

10. Kompresja danych

- i. Warunki kompresji danych.
- ii. Kodowanie Huffmana.
- iii. Metoda Shannona-Fano.

11. Podstawowe metody numeryczne

- i. Macierze i ich właściwości.
- ii. Prawa macierzy.

- iii. Rozwiązywanie układów równań liniowych.
- iv. Eliminacja pełna Jordana-Gausa.
- v. Eliminacja Gausa.
- vi. Rozkład Cholesky'ego.
- vii. Metody iteracyjne.
- viii. Transformaty Fouriera.

7.6 Sieci komputerowe

1. Podstawy sieci

- i. ABC sieci.
- ii. Ewolucja sieci.
- iii. Model referencyjny OSI.
- iv. Podstawy sieci.
- v. Sprzętowe elementy składowe.
- vi. Programowe elementy składowe.
- vii. Składanie elementów w sieć.

2. Typy i topologie sieci LAN.

- i. Urządzenia przyłączane do sieci LAN.
- ii. Typy serwerów.
- iii. Typy sieci.
- iv. Topologie sieci lokalnych.
- v. Hierarchie.
- vi. Obszary funkcjonalne sieci LAN.
- vii. Przyłączanie stacji.
- viii. Przyłączanie serwera.
- ix. Przyłączanie do sieci WAN.
- x. Przyłączanie do szkieletu.

3. Warstwa fizyczna

- i. Funkcje warstwy fizycznej.
- ii. Nośniki transmisji fizycznej.

4. Niezupełnie-fizyczna warstwa fizyczna

- i. Spektrum elektromagnetyczne.
- ii. Bezprzewodowe sieci LAN.
- iii. Technologie transmisji.
- iv. Podczerveń.

5. Warstwa łączy danych

- i. Ramki.
- ii. Składniki typowej ramki.

- iii. Sterowanie łączem logicznym w standardzie IEEE 802.2.
- iv. Zasady sterowania dostępem do nośnika.
- v. Wybór technologii LAN.

6. Tworzenie sieci lokalnych

- i. Różne rodzaje sieci Ethernet.
- ii. Obsługiwany sprzęt.
- iii. Ramka Ethernetu IEEE 802.3.
- iv. Szybsze sieci Ethernet.
- v. Nośniki Fast Ethernetu.
- vi. Gigabit Ethernet.
- vii. Token Ring.
- viii. Struktura ramki Token Ring.
- ix. Topologia Tonek Ring.

7. Protokoły sieciowe

- i. Stosy protokołów.
- ii. Protokół Internetu, wersja 4 (Ipv4).
- iii. Analiza TCP/IP.
- iv. Protokół Internetu, wersja 6 (IPv6).
- v. Struktury adresów.

8. Sieci WAN

- i. Funkcjonowanie technologii WAN.
- ii. Korzystanie z urządzeń transmisji.
- iii. Wybór sprzętu komunikacyjnego.
- iv. Adresowanie międzysieciowe.
- v. Zapewnianie adresowania unikatowego.
- vi. Współdziałanie międzysieciowe z wykorzystaniem różnych protokołów.
- vii. Korzystanie z protokołów trasowania.
- viii. Wybór protokołu.

9. Modemy i technologie Dial-Up

- i. Sposób działania modemu.
- ii. Bity i body.

- iii. Typy modulacji modemów.
- iv. Modemy a Microsoft Networking.

10. Sieciowe systemy operacyjne

- i. Historia sieciowych systemów operacyjnych.
- ii. Novell dominuje rynek.
- iii. Tradycyjne usługi sieciowych systemów operacyjnych.

11. Administrowanie siecią

- i. Administrowanie siecią - cóż to oznacza?
- ii. Zarządzanie kontami sieciowymi.
- iii. Konta użytkowników.
- iv. Konta grup.
- v. Zarządzanie zasobami.
- vi. Zasoby sprzętowe.
- vii. Wydzielone obszary dysku.
- viii. Pliki i katalogi.
- ix. Instalowanie/aktualizowanie oprogramowania.
- x. Drukowanie w sieci.
- xi. Narzędzia zarządzania.
- xii. Narzędzia zarządzania Microsoftu.
- xiii. Zarządzanie Linux.

12. Zarządzanie siecią

- i. Wydajność sieci.
- ii. Warstwa fizyczna.
- iii. Problemy rozróżniania adresów.
- iv. Współdziałanie międzysieciowe.
- v. Ping.
- vi. Traceroute.
- vii. Analizatory sieci.

13. Bezpieczeństwo danych

- i. Planowanie w celu zwiększenia bezpieczeństwa sieci oraz danych.
- ii. Poziomy bezpieczeństwa.

- iii. Grupy robocze, domeny i zaufanie.
- iv. Bezdyskowe stacje robocze.
- v. Szyfrowanie.
- vi. Ochrona antywirusowa.

14. Serwery usług - konfiguracja

- i. Serwer domeny BIND.
- ii. Serwer WWW Apache.
- iii. Serwer baz danych MySQL, Postgres.
- iv. Serwer plików NFS.
- v. Serwer wydruku CUPS.
- vi. Serwer FTP vsFTPD.
- vii. Serwer aplikacji.

7.7 Multimedia i grafika komputerowa

1. Sprzęt grafiki komputerowej, w tym: urządzenia we/wy, zasada działania monitora, i realizacja koloru, funkcje karty graficznej, współdziałanie z P.O. Proces wyświetlania obrazu na ekranie.
2. Modele koloru.
3. Formaty plików obrazowych. Kompresja.
4. Przetwarzanie obrazów rastrowych: cele, metody. Skalowanie, histogramy, macierze sąsiedztwa, filtry, progowanie, dithering.
5. Wykrywanie konturów, wypełnianie konturów, ścienianie. (kod łańcuchowy).
6. Algorytmy rastrowe rysowania linii (Bresenhama).
7. Rysowanie połączeń krzywymi, algorytmy aproksymacji i interpolacji.
8. Reprezentacja punktu we współrzędnych jednorodnych na płaszczyźnie i w przestrzeni, elementarne macierze przekształceń, składanie przekształceń.
9. Reprezentacja danych dla obrazów wektorowych 2-D, algorytmy wektorowe (przecinania odcinków, obcinania kadrem prostokątnym, wielobokiem).
10. Rzutowanie równoległe i perspektywiczne. Przekształcanie układu danych do układu obserwatora.
11. Reprezentacje obiektów w 3-D.
12. Przesłanianie niewidocznych krawędzi.
13. Model oświetlenia, cieniowanie obiektów.
14. Złożone metody wizualizacji: algorytm śledzenia promieni.

7.8 Bazy danych

1. Wprowadzenie do baz danych.
2. Architektura systemu bazy danych.
3. Modele danych. Model relacyjny.
4. Projektowanie baz danych. Projektowanie koncepcyjne. Diagram obiektowo-związkowy.
5. Projektowanie logiczne.
6. Projektowanie aplikacji bazodanowych.
7. Zależności funkcyjne.
8. Normalizacja schematu. Postaci normalne.
9. Języki manipulacji danymi. Języki zapytań. Optymalizacja zapytań.
10. Język oparty na algebrze relacji. Język oparty na rachunku predykatów. Query By Example.
11. Składnia języka SQL.
12. Przetwarzanie transakcyjne.
13. Rozproszone bazy danych.
14. Obiektowe bazy danych.
15. Poufność i bezpieczeństwo baz danych.

8. Kontrola i ocena osiągnięć ucznia

Uczniowie powinni być poinformowani o obowiązujących kryteriach oceniania. Uczeń powinien być świadomy jaki materiał musi być przez niego opanowany aby otrzymać daną ocenę. Nauczyciel określa wymagania na poszczególne oceny z zarysowaniem wymagań koniecznych i podstawowych stanowiących bazę do rozszerzenia wiadomości przy czym wymagania konieczne powinny stanowić 40% wszystkich wymagań. Kolejne stopnie stanowią kolejne 20% progi.

Metody oceniania:

- praca na lekcjach,
- sprawdziany wiadomości bez użycia komputera,
- sprawdziany praktyczne przy komputerze,
- komputerowe testy wiadomości różnego rodzaju,
- praca w projektach,
- współpraca z innymi nauczycielami,
- referaty i przygotowane przez ucznia wykłady i materiały,
- ocena osiągnięć w konkursach i olimpiadach.