**Plan wynikowy z wymaganiami edukacyjnymi przedmiotu informatyka**

**dla klasy II liceum ogólnokształcącego i technikum w zakresie rozszerzonym,**

**uwzględniający kształcone umiejętności i treści podstawy programowej**

Uwaga!

W planie pominięto podstawowe umiejętności, które uczeń powinien już posiadać, np. zachowywanie plików projektów, wczytywanie dokumentów do edycji i posługiwanie się systemem operacyjnym.

Kryteria danej oceny opracowano z założeniem, że zostały spełnione kryteria ocen niższych.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Temat** | **Ocena dopuszczająca Uczeń:** | **Ocena dostateczna  Uczeń:** | **Ocena dobra  Uczeń:** | **Ocena bardzo dobra  Uczeń:** | **Ocena celująca Uczeń:** |
| **I. Opracowania rozwiązań problemów wybranymi aplikacjami** | | | | | |
| 1. Host i serwer, czyli o działaniu sieci | – na podstawie podręcz- nika i z pomocą nauczyciela omawia podstawowe pojęcia związane z budową sieci komputerowych (np. *host*, *serwer*, *klient*, *LAN*, *WAN*, *WiFi*);  –wie, czym są media transmisyjne;  – wymienia nazwy najczęściej spotykanych topologii sieci lokalnej i na podstawie ilustracji (np. z podręcznika) wymienia ich podstawowe cechy. | – samodzielnie omawia znaczenie podstawowych pojęć związanych z budową i działaniem sieci komputerowych;  – na podstawie podręcz- nika omawia cechy poszczególnych mediów transmisyjnych;  – samodzielnie wymienia nazwy najczęściej spotykanych topologii sieci lokalnej i wymienia ich podstawowe cechy;  – wymienia jednostki przesyłu informacji i omawia zależności pomiędzy nimi;  – wie, że do różnych zastosowań można używać różnych rodzajów kabla sieciowego. | – samodzielnie omawia znaczenie wszystkich pojęć związanych z budową i działaniem sieci komputerowych;  – oblicza maksymalną ilość danych przesyłanych w sieci na podstawie danych o czasie i przepustowości łącza;  – z niewielką pomocą nauczyciela wymienia nazwy i cechy podstawowych kabli dla sieci lokalnych CAT 3, 5 i 5e, 6 i 7;  – wymienia podstawowe dla transmisji danych cechy światłowodu. | – wybiera odpowiednią topologię i medium dla różnych sieci lokalnych;  – identyfikuje topologię istniejącej sieci (np. w pracowni);  – identyfikuje rodzaje mediów transmisyjnych;  – prawidłowo stosuje poznane pojęcia dotyczące sieci lokalnych. | – omawia niewymienione w podstawie programowej zagadnienia związane z tematem. |
| 2. Włączamy do ruchu, czyli konfiguracja lokalnej sieci komputerowej | – omawia funkcję routera w sieci lokalnej;  – wie, że router należy odpowiednio skonfigurować;  – podaje, jak znaleźć opis konfiguracji routera (np. instrukcję, samouczek w sieci);  – identyfikuje nazwę i producenta routera. | – omawia funkcję DHCP w kontekście konfiguracji routera;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł wyjaśnia pojęcia: *DHCP*, *serwer*, *szyfrowanie* (np. WPA2-PSK) i ich znaczenie dla konfiguracji routera;  – na podstawie instrukcji i z pomocą nauczyciela omawia lub przeprowa- dza konfigurację przykładowego routera. | – na podstawie instrukcji omawia lub przeprowadza konfigurację przykładowego routera;  – na podstawie instrukcji lub opcji Pomocy omawia lub przeprowadza konfigurację przykładowego routera bez wykorzystania DHCP. | – samodzielnie omawia lub przeprowadza konfigurację przykładowego routera z wykorzystaniem DHCP i bez tego mechanizmu. | – dokonuje konfiguracji zaawansowanych funkcji routera (np. uprawnień poszczególnych użytkowników). |
| 3. Nie tylko WiFi, czyli bezprzewodowy dostęp do internetu | – wie czym się różni WLAN od WiFi;  – omawia zastosowanie złączy Bluetooth do połączenia urządzeń komputerowych;  – z niewielką pomocą nauczyciela przeprowadza przesłanie pliku (np. z telefonu do laptopa) za pośrednictwem Bluetooth. | – wymienia nazwy i omawia przeznaczenie podstawowych mediów i urządzeń transmisyjnych w tym: WLAN, WiFi, Bluetooth;  – odróżnia i prawidłowo przypisuje do danych mediów ich symbole graficzne. | – wyjaśnia, na czym polega łączenie urządzeń za pomocą fal radiowych i podaje, jakie warunki muszą być przy tym spełnione;  – na podstawie opisów omawia różnice między bezprzewodowymi mediami transmisyjnymi;  – wyjaśnia znaczenie norm określających ramy działania poszczególnych mediów transmisyjnych. | – wymienia pasma radiowe przeznaczone dla mediów transmisyjnych stosowanych w sieciach bezprzewodowych;  – opisuje możliwości wykorzystania mediów bezprzewodowych i samodzielnie przepro- wadza konfigurację podłączanych do nich urządzeń (np. słuchawek, telefonu, laptopa). | – omawia zastosowania i działanie WiMAX;  – samodzielnie łączy klika urządzeń jednocześnie za pomocą Bluetooth. |
| 4. Zespół, czyli realizujemy projekty w chmurze | – omawia budowę przykładowej chmury, wymieniając jej podstawowe składniki. | – przedstawia i omawia przykładowy schemat struktury chmury informatycznej;  – wymienia najczęściej spotykane w chmurach narzędzia (m.in. dysk, edytor, arkusz, kalendarz);  – wie, że chmurę można wykorzystać do pracy zespołowej. | – omawia przeznaczenie poszczególnych elementów chmury użytej na zajęciach;  – porównuje działanie programów z chmury z ich odpowiednikami z apli- kacji komputerowych;  – posługuje się chmurą w stopniu wystarczającym do wykonywania podstawowych czynności, takich jak edycja dokumentu, kopiowanie plików itp.;  – wie, które aplikacje można wykorzystać do organizacji pracy zespołu. | – biegle posługuje się programami i dyskami sieciowymi;  – samodzielnie przygotowuje chmurę do pracy zespołu, m.in. konfiguruje kalendarz, udostępnia foldery i pliki. | – samodzielnie planuje i organizuje pracę zespołu w chmurze;  – posługuje się kilkoma chmurami (np. Google i OneDrive). |
| 5. Tutorial, czyli jak tworzyć pomoce i instrukcje obsługi | – wie czym są style i szablony;  – zmienia styl w trakcie edycji dokumentu. | – prawidłowo dobiera style do treści zawartej w dokumencie;  – wie, że można samodzielnie definiować style i szablony;  – uruchamia edycję, wykorzystując szablon oferowany przez edytor. | – dobiera styl i szablon zgodnie z planowaną zawartością i tematem dokumentu;  – na podstawie podręcz- nika definiuje szablon i styl dokumentu;  – odnajduje w sieci (np. w chmurze) szablony dla różnych dokumentów. | – samodzielnie definiuje szablon i styl dokumentu;  – projektuje szablon zgodnie z treścią przyszłego dokumentu;  – modyfikuje i projektuje nowe style. | – projektuje szablony dla różnych edytorów (np. LibreOffice Draw) i ich używa. |
| 6. Rozbudowana struktura, czyli korzystamy z konspektu w edytorze tekstu | – rozumie i omawia pojęcie konspektu;  – podaje, czym jest akapit i jaką pełni funkcję w edycji i formatowaniu tekstu. | – na podstawie podręcz- nika tworzy konspekt dokumentu;  – na podstawie podręcz- nika dzieli dokument na sekcje i kolumny. | – samodzielnie tworzy konspekt dokumentu;  – samodzielnie dzieli dokument na sekcje i kolumny;  – tworzy spis treści na podstawie konspektu. | – uzasadnia stosowanie podziału dokumentu tekstowego na sekcje lub kolumny;  – stosuje różny podział na niektórych stronach (sekcjach) dokumentu. | – stosuje podziały i sekcje w różnych edytorach tekstu. |
| 7. Broszura, czyli jak projektować duże dokumenty | – wstawia do dokumentów SmartArt i Kształty. | – uzasadnia wybór danego kształtu lub ilustracji SmartArt;  – na podstawie podręcz- nika tworzy spisy ilustracji i tabel. | – na podstawie podręcz- nika lub tutoriali zmienia domyślne opcje edytora;  – na podstawie podręcz- nika lub tutoriali wyko- nuje automatyczne spisy treści, tabel i ilustracji;  – na podstawie podręcz- nika lub tutoriali aktuali- zuje spisy po zmianach. | – samodzielnie zmienia domyślne opcje edytora;  – samodzielnie wykonuje automatyczne spisy treści, tabel i ilustracji;  – samodzielnie aktualizuje spisy po zmianach;  – decyduje, które z opcji domyślnych zmienić i uzasadnia swój wybór. | – spełnia kryteria oceny bardzo dobrej dla innych edytorów (np. LibreOffice). |
| 8. Recenzja, czyli proponujemy poprawki w tekście | – odczytuje i interpretuje dokumenty, w których zastosowano tryb recenzji;  – odczytuje notatki i zaznaczenia w Adobe Acrobat Reader DC. | – na podstawie podręcz- nika używa opcji Recenzja w Word i wstawiania komentarzy w dokumencie PDF. | – na podstawie podręcz- nika lub tutoriali używa opcji Recenzja do wpisywania komentarzy w Word oraz opcji wstawiania komentarzy i zaznaczania fragmentów w dokumencie PDF;  – na podstawie podręcz- nika lub tutoriali uruchamia opcję śledzenia zmian. | – samodzielnie używa opcji Recenzja do wpisywania komentarzy w Word oraz opcji wstawiania komentarzy i zaznaczania fragmentów w dokumencie PDF;  – samodzielnie uruchamia opcję śledzenia zmian;  – świadomie używa opcji Recenzja do proponowa- nia zmian w dokumencie;  – analizuje wynik działa- nia opcji porównywania dokumentów. | – spełnia kryteria oceny bardzo dobrej dla innych edytorów (np. LibreOffice);  – przedstawia dokument z innego przedmiotu lub projektu, w którym użył opcji Recenzja dla dokumentu Word lub Notatka dla PDF. |
| 9. OLE, czyli łączymy dokumenty w dokumentacji pracy zespołu | – na podstawie podręcz- nika wyjaśnia, czym jest i do czego służy technika OLE. | – wyjaśnia, jakie obiekty mogą być wstawiane za pomocą techniki OLE;  – na podstawie podręcz- nika lub opcji Pomoc umieszcza w dokumencie obiekty OLE;  – na podstawie podręcz- nika wyjaśnia, czym są OLE2 i ActiveX;  – wyjaśnia, dlaczego podczas używania ActiveX należy zachować szczególną ostrożność. | – na podstawie podręcz- nika i opcji Pomocy przetwarza obiekty OLE w edytorach tekstu itp. | – samodzielnie przetwarza obiekty OLE w edytorach tekstu itp. | – spełnia kryteria oceny bardzo dobrej dla innych edytorów (np. LibreOffice). |
| **II. Rozwijanie kompetencji społecznych** | | | | | |
| 10. Informatyka pokonuje schody, czyli nikt nie powinien być wykluczony | – wymienia cechy aplikacji ratujących życie lub ułatwiających funkcjonowanie osobom niepełnosprawnym. | – znajduje i instaluje wiarygodne aplikacje ratujące życie lub zdrowie;  – wymienia urządzenia peryferyjne do monitorowania parametrów organizmu;  – posługuje się translatorem języków. | – ocenia jakość aplikacji ratujące życie lub zdrowie;  – ocenia jakość aplikacji ułatwiających funkcjono- wanie osobom niepełno- sprawnym i starszym;  – wymienia cechy strony dostosowanej do potrzeb osób niepełnosprawnych;  – używa peryferiów monitorujących stan organizmu. | – dobiera peryferia i aplikacje pomagające w ratowaniu życia lub zdrowia;  – omawia cechy dobranych przez siebie aplikacji i porównuje ich jakość;  – pokazuje przykłady stron przystosowanych do potrzeb osób niepełnosprawnych. | – projektuje strony przystosowane do potrzeb osób niepełnosprawnych;  – korzysta z translatora w trybie offline. |
| 11. Platformy uczą, czyli rola e-learningu w nauce i pracy | – wymienia podstawowe różnice między nauczaniem zdalnym a klasycznym. | – wymienia zalety i wady e-learningu;  – umie skorzystać z gotowych lekcji  e-learningowych. | – opisuje przykładową strukturę lekcji e-learningowej;  – opisuje przykładowy cykl nauczania e-learningowego. | – wymienia właściwości wybranego systemu do tworzenia i prowadzenia kursów e-learningowych;  – porównuje zalety i wady e-learningu z nauczaniem tradycyjnym;  – znajduje w sieci kursy e-learningowe na dany temat i odczytuje, na jakich warunkach można w nich uczestniczyć. | – układa scenariusz lekcji e-learningowej i wskazuje różnice w porównaniu z tradycyjną lekcją. |
| 12. Możesz być administratorem, czyli jak zarządzać platformą e-learningową | – omawia warunki, jakie musi spełniać platforma e-learningowa. | – uruchamia aplikację Classroom;  – wymienia elementy, jakie powinny się znaleźć w strukturze kursu e-learningowego;  – na podstawie podręcz- nika tworzy lekcje w Classroom. | – projektuje przykładową strukturę kursu e-learningowego;  – sporadycznie korzystając z podręcznika, tworzy lekcję i dodaje temat zajęć w Classroom. | – projektuje strukturę kursu e-learningowego na zadany temat i tworzy ją w Classroom;  – testuje poprawność działania kursu. | – układa kurs e-learningowy w innej platformie niż Classroom. |
| 13. Zasoby i testy, czyli wypełniamy kursy treścią | – wymienia rodzaje zasobów, które można umieszczać w kursach e-learningowych. | – objaśnia rolę poszczególnych rodzajów zasobów kursów e-learningowych;  – wymienia rodzaje pytań testowych, jakie mogą się znaleźć w testach online. | – dzieli zasoby kursów według różnych kryteriów;  – sporadycznie korzysta- jąc z podręcznika, umiesz- cza w Classroom gotowe zasoby edukacyjne i testy;  – zaprasza użytkowników do korzystania z kursu. | – samodzielnie tworzy część zasobów, które umieści w Classroom;  – tworzy krótkie testy w Classroom. | – umieszcza zasoby w innej platformie niż Classroom. |
| 14. Dzielimy się wiedzą, czyli współtworzymy zasoby udostępniane na platformach | – uzasadnia potrzebę współdzielenia zasobów chmury lub platformy e-learningowej. | – z pomocą nauczyciela lub na podstawie podręcznika udostępnia zasoby umieszczone w chmurze lub platformie. | – samodzielnie udostępnia zasoby na platformie lub chmurze w wybrany przez siebie sposób. | – samodzielnie udostępnia zasoby, korzystając z różnych sposobów oferowanych przez oprogramowanie chmury lub platformy. | – wykonuje czynności opisane w podręczniku także w innych chmurach i platformach. |
| **III. Bazy danych** | | | | | |
| 15. Jak gromadzić informacje, czyli komputerowe bazy danych | – podaje przykłady baz danych, z których korzysta np. w swoim telefonie;  – definiuje pojęcie bazy danych. | – wymienia wszystkie elementy bazy danych opisane w podręczniku. | – omawia przeznaczenie poszczególnych elementów bazy danych. | – samodzielnie opisuje czynności występujące w trakcie projektowania bazy danych. | – opisuje zależności między poszczególnymi elementami bazy danych. |
| 16. Tabele i formularze, czyli jak utworzyć bazę danych | – wymienia przykładowe dane, jakie powinna zawierać baza uczestników projektu. | – uzasadnia wybór danych dla bazy uczestników projektu;  – na podstawie podręcz- nika tworzy tabele bazy danych zgodnie z danymi zdefiniowanymi dla uczestników projektu;  – rozumie znaczenie tabel w bazie danych. | – samodzielnie tworzy kolejne tabele projektu;  – korzystając z kreatora odnośników, tworzy je dla pól tabeli, wybierając je spośród wskazanych w podręczniku;  – umieszcza dane w tabelach;  – omawia znaczenie klucza;  – prawidłowo posługuje się pojęciami związanymi z bazami danych. | – tworzy klucze;  – tworzy odnośniki dla pól tabel;  – tworzy formularze. | – czynności opisane dla Access wykonuje także w innym systemie baz danych (np. LibreOffice Base). |
| 17. Relacje i pytania, czyli jak uczynić bazę użyteczną | – wyjaśnia znaczenie relacji między tabelami bazy. | – wyjaśnia różnice między rodzajami relacji. | – tworzy klucz zewnętrzny;  – używa kreatora relacji między tabelami;  – wypełnia pola tabel powiązanych za pomocą kreatora;  – tworzy relacje za pomocą narzędzia Relacje na podstawie opisu w podręczniku. | – tworzy relacje za pośrednictwem kreatora i narzędzia Relacje;  – tłumaczy sens tworzenia relacji i efekt ich działania w przykładzie z ćwiczeń. | – czynności opisane dla Access wykonuje także w innym systemie baz danych (np. LibreOffice Base). |
| 18. Kto pyta, nie błądzi, czyli jak korzystać z baz danych Access | – wyjaśnia znaczenie pojęcia *kwerenda*;  – wyjaśnia, na czym polega filtrowanie w informatyce. | – na podstawie podręcz- nika tworzy kwerendę, używając kreatora;  – na podstawie podręcz- nika lub tutoriali modyfikuje kwerendę, dodając filtrowanie;  – na podstawie podręcz- nika lub tutoriali sortuje dane i tworzy raport za pomocą kreatora. | – samodzielnie tworzy kwerendę, używając kreatora;  – samodzielnie modyfikuje kwerendę, dodając filtrowanie;  – samodzielnie sortuje dane i tworzy raport za pomocą kreatora. | – planuje kwerendę i ją opracowuje;  – prawidłowo decyduje o wyborze filtrowania, sposobie sortowania i raportowania;  – wykonuje eksport tabel do innych formatów (np. Excel). | – czynności opisane dla Access wykonuje także w innym systemie baz danych (np. LibreOffice Base). |
| 19. Formularze, czyli ułatwiamy sobie wprowadzenie informacji do bazy danych | – z pomocą nauczyciela omawia przykład zamieszczony w podręczniku. | – wymienia nazwy róż- nych rodzajów formularzy i na podstawie podręcz- nika omawia ich cechy i zastosowanie;  – na podstawie opisów tworzy różne rodzaje formularzy za pomocą kreatora. | – przy tworzeniu formularzy korzysta z różnych rodzajów list. | – konstruuje formularze z widoku projektu, wykorzystując formaty;  – używa przycisków nawigacyjnych w formularzu. | – czynności opisane dla Access wykonuje także w innym systemie baz danych (np. LibreOffice Base). |
| 20. Czy to możliwe, czyli baza danych w arkuszu Excel | – omawia analogie w tworzeniu tabel w Access i Excel. | – na podstawie podręcz- nika lub tutoriali tworzy w Excelu tabele  z nagłówkami;  – na podstawie podręcz- nika lub tutoriali używa formularzy do wprowa- dzania danych do tabel;  – na podstawie podręcz- nika lub tutoriali tworzy listę rozwijaną. | – samodzielnie tworzy w Excelu tabele  z nagłówkami;  – samodzielnie używa formularzy do wprowa- dzania danych do tabel;  – samodzielnie tworzy listę rozwijaną;  – stosuje sortowanie wegług wskazanych kryteriów;  – wstawia narzędzia do paska Szybki dostęp. | – planuje i wykonuje czynności tworzenia bazy w Excelu;  – dobiera kryteria sortowania;  – przenosi tabele z bazy danych do Excela. | – czynności opisane dla Excela wykonuje także w innym arkuszu (np. LibreOffice Calc). |
| 21. Wiedza w sieci, czyli internet mądrych ludzi | – wyszukuje informacje w domyślnej wyszukiwarce przeglądarki internetowej;  – omawia znaczenie zachowania praw autorskich i podstawy licencji CC. | – na podstawie podręcz- nika zmienia domyślną wyszukiwarkę w przeglądarce Firefox;  – na podstawie podręcz- nika uszczegółowia dane w wyszukiwarce w celu zwiększenia prawdopo- dobieństwa dotarcia do szukanej informacji;  – na podstawie podręcz- nika wyszukuje za pomocą obrazu. | – uzasadnia celowość korzystania z różnych wyszukiwarek;  – pozyskuje informacje z baz danych, takich jak europeana.eu;  – tłumaczy podstawy i cel działania botów indeksujących;  – tłumaczy na przykładzie konieczność uszczegółowiania zapytań do wyszukiwarki;  – korzysta z zaawansowanych opcji wyszukiwarek;  – ocenia wiarygodność źródeł, wymieniając świadczące o niej cechy. | – planuje użycie odpowiedniej wyszukiwarki;  – wybiera bazę danych do wyszukiwania konkretnych informacji;  – trafnie dobiera słowa kluczowe, zarówno w opcjach głównych, jak i zaawansowanych wyszukiwarek;  – zmienia wyszukiwarki domyślne w różnych przeglądarkach internetowych. | – omawia cechy różnych wyszukiwarek internetowych i ocenia ich przydatność do konkretnego zadania. |
| 22. Wirtualne serwery, czyli instalujemy Apache | – omawia pojęcie *serwer*;  – uruchamia środowisko XAMP. | – na podstawie podręcz- nika i doświadczeń z poprzednich lekcji instaluje i uruchamia środowisko XAMP. | – samodzielnie instaluje środowisko XAMP, wybierając wskazane komponenty;  – omawia zawartość XAMP i jego zastosowanie. | – decyduje o wyborze komponentów na podstawie informacji o późniejszym wykorzystaniu XAMP. | – instaluje i korzysta także z innych środowisk symulujących działanie serwera oraz omawia ich cechy. |
| 23. Język zapytań, czyli poznajemy język SQL | – wyjaśnia, do czego służy język SQL. | – na podstawie podręcz- nika wymienia i krótko charakteryzuje podsta- wowe elementy składni SQL – DML, DDL, DCL, DQL;  – na podstawie podręcz- nika lub innych publikacji omawia typy danych języka SQL. | – na podstawie podręcz- nika omawia podstawowe elementy składni SQL;  – omawia przeznaczenie niektórych typów danych wykorzystywanych w MySQL. | – samodzielnie charakteryzuje podstawowe elementy składni SQL. | – omawia nieopisane w podręczniku elementy składni SQL. |
| 24. Zadajemy pytania, czyli kwerendy w języku SQL | – na podstawie wiedzy z poprzednich lekcji omawia definicję i zastosowanie kwerendy;  – na podstawie podręcznika uruchamia elementy XAMP niezbędne do korzystania z bazy danych;  – przy pomocy nauczyciela i na podstawie opisu wykonuje ćwiczenia z podręcznika;  – na podstawie podręcznika lub innych źródeł omawia różnicę między SQL a MySQL. | – na podstawie podręcz- nika omawia definicję MySQL jako otwarto- źródłowego systemu zarządzania bazami danych;  – na podstawie podręcz- nika uruchamia phpMyAdmin i jego narzędzia z pakietu XAMP;  – na podstawie podręcz- nika i z niewielką pomocą nauczyciela tworzy nową bazę i tabele w MySQL;  – na podstawie podręcz- nika i z niewielką pomocą nauczyciela tworzy zapytania w SQL. | – na podstawie podręcz- nika wykonuje ćwiczenie i wyjaśnia poszczególne etapy tworzenia zapytań;  – omawia przykładowy kod tworzenia tabeli w SQL;  – na podstawie tabeli omawia operatory wykorzystywane podczas wyszukiwania informacji w bazie. | – samodzielnie tworzy bazę i zapytania w języku SQL;  – samodzielnie korzysta z MySQL. | – samodzielnie tworzy programy w SQL inne niż w przykładach podanych w podręczniku. |
| 25. Więcej pytań, czyli jeszcze o kwerendach w języku SQL | – uruchamia w XAMP przykładowe programy SQL z podręcznika lub przedstawione przez nauczyciela. | – na podstawie podręcz- nika omawia składnię polecenia SELECT;  – na podstawie podręcz- nika omawia działanie klauzuli WHERE i operatorów logicznych;  – na podstawie podręcz- nika omawia efekt łączenia tabel w celu wyszukiwania informacji;  – na podstawie podręcz- nika analizuje umieszczone w nim przykłady w SQL. | – samodzielnie omawia składnię polecenia SELECT;  – samodzielnie omawia działanie klauzuli WHERE i operatorów logicznych;  – samodzielnie omawia efekt łączenia tabel w celu wyszukiwania informacji;  – samodzielnie analizuje umieszczone w podręcz- niku przykłady w SQL;  – sporadycznie korzystając ze źródeł wiedzy, wykonuje ćwiczenia z podręcznika. | – wykorzystuje zapytanie SELECT i operatory logiczne podczas tworzenia zapytań;  – łączy tabele w celu przeszukania większej ich liczby;  – wyjaśnia, jak szukać informacji w tabelach niepołączonych ze sobą. | – operuje MySQL i układa programy w SQL w przykładach innych niż poznane w trakcie lekcji. |
| 26. Podzapytania, czyli instrukcje modyfikujące dane w języku SQL | – na podstawie podręcznika wyjaśnia pojęcie *podzapytania*. | – wyjaśnia pojęcie *podzapytania*;  – na podstawie podręcznika omawia znaczenie podzapytań. | – na podstawie podręcz- nika wykonuje opisane w nim ćwiczenie;  – wyjaśnia istotę podzapytań wierszowych i skorelowanych;  – wyjaśnia istotę podzapytań o charakterze tabeli tymczasowej;  – omawia przykłady programów SQL realizujących podzapytania. | – układa programy podzapytań w SQL do przykładów z podręcznika lub podanych przez nauczyciela. | – samodzielnie układa programy podzapytań w SQL. |
| 27. Modyfikacje, czyli zarządzanie bazą danych | – na podstawie podręcznika wyjaśnia działanie poleceń: UPDATE, ALERT TABLE, DELETE, DELETE FROM. | – na podstawie podręcz- nika omawia składnię poleceń: UPDATE, ALERT TABLE, DELETE, DELETE FROM. | – z pomocą nauczyciela lub na podstawie podręcznika modyfikuje elementy bazy danych. | – samodzielnie, według zaleceń modyfikuje elementy (zmienia strukturę, usuwa rekordy, tabele) i strukturę bazy danych. | – samodzielnie określa zakres modyfikacji bazy ze względu na zaistniałe potrzeby i modyfikuje elementy i strukturę bazy danych. |
| 28. Bezpieczeństwo bazy, czyli tworzymy kopie zapasowe | – omawia zasady dbania o bezpieczeństwo baz danych;  – podaje, czym jest kopia zapasowa. | – na podstawie podręcz- nika omawia istotę najczęściej spotykanych ataków na bazę (pasywne i aktywne);  – wymienia zasady, jakimi powinien się kierować administrator bazy w celu zabezpieczenia jej przed nieuprawnionym dostępem lub skasowaniem danych. | – omawia na przykładach definicje ataków pasywnych i aktywnych;  – omawia zasady, jakimi powinien się kierować administrator bazy w celu zabezpieczenia jej przed nieuprawnionym dostępem lub skasowaniem danych;  – omawia działanie poleceń: CHECK TABLE, REPAIR TABLE, RESET MASTER, PURGE MASTER, SHOW BINLOG EVENTS;  – omawia składnię poleceń do odzyskiwania danych z kopii bezpieczeństwa. | – tworzy kopię bezpieczeństwa, używając odpowiednich poleceń;  – wyjaśnia pojęcie przyrostowej kopii bezpieczeństwa;  – odzyskuje dane z kopii bezpieczeństwa. | – planuje zabezpieczenie bazy danych. |
| **IV. Programowanie stron internetowych** | | | | | |
| 29. HTML, czyli przeglądarka interpretuje język programowania stron | – omawia przeznaczenie języka HTML;  – omawia rolę znaczników w języku HTML. | – omawia rolę przeglądar- ki internetowej w kontek- ście języka HTML;  – instaluje, uruchamia i korzysta ze wskazanego edytora (np. Notepad+) skonfigurowanego dla HTML5;  – korzysta z opisów naj- częściej używanych znacz- ników (np. z podręcznika). | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł konfiguruje edytor do układania programów w HTML5;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł układa proste programy stron, np. wyświetlające tekst z zachowaniem prawid- łowej struktury programu;  – samodzielnie objaśnia rolę CSS w projektowaniu wyglądu strony. | – dobiera odpowiedni edytor do edycji programów w HTML5 i uzasadnia swój wybór;  – konfiguruje edytor;  – tworzy proste strony internetowe zawierające sformatowany tekst. | – tworzy rozbudowane strony internetowe w języku HTML5. |
| 30. Budujemy stronę, czyli tabele, listy i inne elementy dobrej strony | – omawia znaczenie list w programie HTML5;  – uruchamia w przeglądarce gotowy program strony w HTML5 (np. z przykładu z podręcznika). | – na podstawie podręcz- nika uruchamia programy z przykładów;  – na podstawie przykła- dów omawia rolę list, definicji i tabel;  – tłumaczy istotę i przeznaczenie hipertekstu. | – na podstawie podręcz- nika lub innych wiary- godnych źródeł tworzy w kodzie HTML listy uporządkowane i nieuporządkowane oraz omawia ich znaczenie;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł tworzy listy definicji z dodanymi nagłówkami;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł tworzy tabele w HTML. | – samodzielnie tworzy w kodzie HTML listy uporządkowane i nieuporządkowane oraz omawia ich znaczenie;  – samodzielnie tworzy listy definicji z dodanymi nagłówkami;  – samodzielnie tworzy tabele w HTML;  – planuje użycie odpowiednich konstrukcji do projektowanej strony;  – używa atrybutów w konstrukcji hipertekstu. | – tworzy rozbudowane strony internetowe w języku HTML5. |
| 31. Tabele i grafika, czyli kolejne składowe stron internetowych w HTML | – wymienia zasady przygotowania grafiki do publikacji na stronie z uwzględnieniem wymiarów i rozdzielczości. | – na podstawie podręcz- nika wstawia przygoto- waną przez nauczyciela grafikę do gotowego kodu strony;  – na podstawie podręcz- nika tłumaczy znaczenie wymiarów i skalowania dla szybkości wczytywania strony. | – określa parametry zdjęcia, biorąc pod uwagę jego miejsce na stronie;  – zmienia parametry zdjęcia przeznaczonego na stronę za pomocą prostych narzędzi systemu Windows. | – planuje miejsce na grafikę;  – przygotowuje grafikę do publikacji na stronie;  – umieszcza zdjęcia w planowanym miejscu i formacie na stronie w HTML;  – tworzy odnośnik z elementu graficznego umieszczonego na stronie w HTML. | – tworzy rozbudowane strony internetowe w języku HTML5 z zastosowaniem elementów graficznych. |
| 32. Składnia stylów, czyli jak CSS pomaga w programowaniu wyglądu strony | – objaśnia rolę CSS w kreowaniu wyglądu strony. | – objaśnia istotę pliku stylów i jego połączenie z HTML. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia zasady łączenia CSS z HTML;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł układa proste pliki stylów CSS, formatujące sposób wyświetlania tekstu;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł wyjaśnia znaczenie nagłówków i bloków w stosowaniu CSS;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł wyjaśnia znaczenie sekcji i selektora. | – samodzielnie omawia zasady łączenia CSS z HTML;  – samodzielnie układa proste pliki stylów CSS, formatujące sposób wyświetlania tekstu;  – samodzielnie wyjaśnia znaczenie nagłówków i bloków w stosowaniu CSS;  – samodzielnie wyjaśnia znaczenie sekcji i selektora;  – projektuje użycie CSS do formatowania stylu tekstu na stronie. | – tworzy rozbudowane strony internetowe w języku HTML5 z zastosowaniem CSS. |
| 33. Pliki stylów, czyli CSS w akcji | – objaśnia działanie i znaczenie odsyłaczy na stronie w HTML;  – omawia rolę menu na stronie. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia znaczenie selektorów i ich atrybutów w pliku CSS;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia pojęcie *walidacja*. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł tworzy pliki HTML i CSS z menu z przyciskami, wykorzys- tując przykłady;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł używa pseudoklas z przykładu do określenia interakcji klawisza z kursorem;  – na podstawie podręcznika lub innych wiarygodnych źródeł rozmieszcza elementy strony, tworząc jej prosty layout. | – samodzielnie tworzy pliki HTML i CSS z menu z przyciskami, wykorzystując przykłady;  – samodzielnie używa pseudoklas z przykładu do określenia interakcji klawisza z kursorem;  – samodzielnie rozmiesz- cza elementy strony, tworząc jej prosty layout;  – zmienia wygląd i atrybu- ty menu z przykładu;  – zmienia położenie elementów strony;  – stosuje różną orientację elementów menu;  – przeprowadza walidację strony. | – tworzy rozbudowane strony internetowe w języku HTML5 z zastosowaniem CSS zawierające menu i efekty interakcji z kursorem. |
| 34. Treści mogą się zmieniać, czyli elementy dynamiczne na stronie internetowej | – wskazuje elementy istniejących stron, które zmieniają się dynamicznie;  – na podstawie podręcz- nika wskazuje różnice między statyczną a dynamiczną strona internetową. | – omawia sposoby wklejania w kod strony gotowych odnośników do elementów dynamicznych;  – na podstawie podręcz- nika umieszcza w kodzie strony elementy dynamiczne. | – tworzy element dynamiczny zewnętrzny za pomocą darmowego edytora aplikacji (np. learningapps.com);  – pobiera i umieszcza na stronie kod urucha- miający widżety lub aplikacje internetowe z learningapps.com. | – planuje umieszczenie na stronie widżetów i innych elementów dynamicz- nych;  – dokonuje prawidłowego wyboru widżetów;  – projektuje elementy dy- namiczne (np. krzyżówki) w learningapps.com. | – tworzy strony z własnymi elementami dynamicznymi. |
| 35. Widoczna w internecie, czyli jak opublikować stronę | – omawia funkcje domeny internetowej;  – wskazuje przynajmniej jedną firmę (stronę internetową) zajmującą się rejestracją domen. | – omawia proces wczytywania strony internetowej do przeglądarki i rolę domeny;  – omawia znaczenie niektórych domen (np. .com, .pl, .edu);  – podaje, jakie usługi musi uruchomić lub zamówić, by strona była widoczna w sieci. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł opisuje na przykładzie proces rezerwacji domeny;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł zakłada konta na darmowym serwerze z usługą hostingową (np. cba.pl);  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł wysyła pliki strony do serwera www. | – samodzielnie opisuje na przykładzie proces rezerwacji domeny;  – samodzielnie zakłada konta na darmowym serwerze z usługą hostingową (np. cba.pl);  – samodzielnie wysyła pliki strony do serwera www;  – sprawdza zajętość domen, którymi jest zainteresowany;  – korzysta z klienta FTP nie tylko do przesyłania plików strony. | – podaje, jak zarządzać domeną i usługą hostingową na przykładzie swojego konta. |
| 36. CMS, czyli system zarządzania treścią strony internetowej | – omawia definicję CMS-u;  – podaje, do czego służy CMS i wymienia kilka najpopularniejszych (w tym Joomla i WordPress). | – omawia proces przy- gotowań do instalacji CMS-u;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł instaluje środowisko serwerowe (np. Laragon), a w nim CMS (np. WordPress);  – na podstawie podręcz- nika lub innych wiarygodnych źródeł loguje się do panelu sterowania CMS-em (np. WordPress). | – samodzielnie instaluje środowisko serwerowe (np. Laragon), a w nim CMS (np. WordPress);  – samodzielnie loguje się do panelu sterowania CMS-em (np. WordPress). | – dobiera odpowiedni do tematu strony CMS i uzasadnia swój wybór;  – wyjaśnia konieczność instalacji bazy danych dla CMS-a;  – używa wirtualnego środowiska serwera do testowania działania różnych stron internetowych. | – buduje ciekawe strony internetowe za pomocą różnych CMS-ów. |
| 37. Panel i skórki, czyli tworzymy stronę w CMS | – wyjaśnia znaczenie szablonu (skórki) w systemie CMS;  – przegląda oferowane w sieci skórki dla WordPressa. | – odróżnia motywy od szablonów;  – omawia znaczenie i funkcję motywów w szablonach;  – na podstawie podręcznika wypełnia treścią stronę zbudowaną na bazie szablonu. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł instaluje w CMS-ie wskazany szablon;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł dobiera motyw i zmienia go w razie potrzeby;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł modyfikuje motyw skórki;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł wypełnia treścią stronę, tworząc nowe wpisy;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł umieszcza na stronie elementy graficzne. | – samodzielnie instaluje w CMS-ie wskazany szablon;  – samodzielnie dobiera motyw i zmienia go w razie potrzeby;  – samodzielnie modyfikuje motyw skórki;  – samodzielnie wypełnia treścią stronę, tworząc nowe wpisy;  – samodzielnie umieszcza na stronie elementy graficzne;  – trafnie i z uzasadnie- niem dobiera szablon i motyw do treści przyszłej strony;  – administruje stroną, wykorzystując panel sterowania;  – dodaje i usuwa użytkowników CMS-a i świadomie nadaje im uprawnienia;  – umieszcza na stronie filmy i inne elementy multimedialne. | – planuje i realizuje własne projekty stron na bazie CMS. |
| 38. Szybkie i łatwe, czy programy do tworzenia stron internetowych | – podaje, jak zapisać tekstowy dokument Word w formacie HTML;  – wymienia kilka systemów darmowych blogów. | – na podstawie podręcznika lub innych wiarygodnych źródeł eksportuje do formatu HTML dokumenty Word i Excel;  – na podstawie podręcznika lub innych wiarygodnych źródeł zakłada konto w systemie darmowych blogów wskazanym przez nauczyciela. | – samodzielnie eksportuje do formatu HTML dokumenty Word i Excel;  – samodzielnie zakłada konto w systemie darmowych blogów wskazanym przez nauczyciela;  – formatuje dokument Word I Excel w taki sposób, by po eksporcie do HTML powstała estetyczna i funkcjonalna strona;  – wypełnia treścią blog utworzony w darmowym systemie blogów. | – prawidłowo i estetycznie rozmieszcza elementy graficzne, wzory itp. w eksportowanym dokumencie;  – testuje w środowisku serwerowym (np. Lara- gon) strony utworzone podczas eksportu dokumentów do HTML, zachowując układ folderów. | – prowadzi własnego bloga na ciekawy temat. |
| 39. Pierwsze skrypty, czyli poznajemy PHP | – omawia specyfikę języka PHP (skrypt umieszczony w HTML i na odwrót);  – uruchamia środowisko Notepad++. | – na podstawie podręcz- nika omawia umieszcza- nie skryptów PHP w dokumencie HTML;  – uruchamia gotowe przykłady z podręcznika lub podane przez nauczyciela. | – omawia rolę funkcji w języku PHP, w tym: print(), echo(), printf();  – omawia działanie przykładowego skryptu, np. z podręcznika. | – układa prosty skrypt wyświetlający tekst;  – umieszcza skrypt na serwerze, uruchamia go w dokumencie HTML i testuje w przeglądarce. | – tworzy proste skrypty PHP. |
| 40. Stałe i zmienne, czyli typy danych w języku PHP | – na podstawie podręcz- nika wyjaśnia, czym są stałe i zmienne w języku PHP. | – wymienia znaki, jakie mogą być użyte w nazwach zmiennych i stałych w języku PHP;  – na podstawie podręcznika omawia typy zmiennych używanych w języku PHP;  – na podstawie podręcz- nika analizuje przykłado- wy skrypt PHP, w którym zastosowano zmienne i stałe;  – wymienia operatorów w języku PHP i omawia ich działanie. | – analizuje przykładowy skrypt ze wskazaniem funkcji, jakie pełnią zmienne i stałe;  – omawia sposób definiowania zmiennych i stałych w PHP;  – prawidłowo używa operatorów. | – układa nieskomplikowa- ne skrypty według specyfikacji, w których używa zmiennych i stałych. | – układa na podstawie własnych specyfikacji i pomysłów proste skrypty PHP. |
| 41. Instrukcje sterujące, czyli praktyczne wykorzystanie języka PHP | – wyjaśnia ogólnie, na czym polega działanie instrukcji warunkowych i wyboru. | – omawia działanie instrukcji warunkowych i wyboru w PHP przez analogię do podobnych instrukcji języka C++;  – na podstawie opisu omawia działanie przyk- ładowych skryptów PHP zawierających instrukcje warunkowe i wyboru. | – samodzielnie omawia działanie przykładowych skryptów PHP zawiera- jących instrukcje warunkowe i wyboru, uruchamia je i testuje. | – układa według specyfikacji skrypty PHP zawierające instrukcje warunkowe i wyboru. | – układa na podstawie własnych specyfikacji i pomysłów skrypty PHP zawierające instrukcje warunkowe i wyboru. |
| 42. Pętle, czyli jeszcze więcej możliwości wykorzystania języka PHP | – wyjaśnia ogólnie, na czym polega działanie pętli w algorytmie i w języku programowania. | – omawia działanie pętli: for, while, do while w PHP przez analogię do podob- nych instrukcji języka C++;  – wyjaśnia różnice między instrukcjami pętli. | – omawia działanie przykładowych skryptów PHP zawierających pętle;  – analizuje przykłady skryptów zawierających pętle. | – układa według specyfikacji skrypty PHP zawierające pętle. | – układa na podstawie własnych specyfikacji i pomysłów skrypty PHP zawierające pętle. |
| 43. Funkcje w języku PHP, czyli jak wzbogacić naszą stronę | – na podstawie podręcznika objaśnia, na czym polega użycie funkcji w programie komputerowym. | – na podstawie podręcz- nika wyjaśnia różnice między funkcjami wbudo- wanymi a własnymi;  – na podstawie podręcznika wymienia nazwy i określa rolę jaką odgrywają niektóre funkcje wbudowane. | – samodzielnie wyjaśnia różnice między funkcjami wbudowanymi a własnymi;  – samodzielnie wymienia nazwy i określa rolę, jaką odgrywają niektóre funkcje wbudowane;  – analizuje przykłady skryptów ułożonych w PHP z użyciem funkcji;  – testuje przykłady skryptów PHP zawierających funkcje. | – omawia rolę niektórych funkcji wbudowanych;  – tworzy skrypt PHP według specyfikacji z użyciem funkcji wbudowanych i testuje ich działanie. | – układa skrypty PHP zawierające funkcje własne. |
| 44. Tablice, czyli jak sortować dane w języku PHP | – omawia na realnym przykładzie istotę tablic. | – omawia różnice między zmiennymi tablicowymi w języku C++ a PHP;  – omawia różnice między tablicami jedno- a wielowymiarowymi;  – uruchamia przykładowe skrypty PHP zawierające tablice i na podstawie podręcznika omawia ich działanie. | – samodzielnie analizuje działanie przykładowych skryptów PHP zawierających tablice. | – układa według specyfikacji skrypty zawierające tablice. | – układa według własnych specyfikacji i pomysłów skrypty PHP zawierające tablice. |
| 45. Pobieranie i przetwarzanie danych, czyli obsługa formularza w języku PHP | – omawia istotę i cele stosowania formularzy na stronach internetowych. | – na podstawie opisu omawia sedno metod POST i GET stosowanych w budowaniu formularzy;  – uruchamia i testuje przykładowe skrypty z formularzami. | – samodzielnie analizuje działanie przykładowych skryptów PHP zawiera- jących formularze;  – proponuje zmiany w przykładowych formularzach w związku ze zmianą specyfikacji skryptu. | – układa według specyfikacji skrypty tworzące na stronie formularze, testuje je i modyfikuje;  – używa w swoich skryptach obu metod i uzasadnia ich wybór. | – układa według własnych specyfikacji i pomysłów skrypty PHP zawierające zaprojektowane przez siebie formularze. |
| **V. Algorytmy i programowanie** | | | | | |
| 46. Powtarzanie w pętli i wywołanie siebie, czyli iteracja i rekurencja w algorytmach | – omawia na realnych przykładach różnicę między rekurencją a iteracją. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje przykładowe algorytmy iteracyjne i rekurencyjne;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł rozpoznaje procesy rekurencyjne i iteracyjne. | – samodzielnie analizuje przykładowe algorytmy iteracyjne i rekurencyjne;  – samodzielnie rozpoz- naje procesy rekurencyjne i iteracyjne;  – wskazuje w przykłado- wych algorytmach miejsca, które decydują o iteracyjności lub rekurencyjności opisywanego procesu;  – analizuje przykładowy program. | – układa algorytmy z podejściem iteracyjnym i rekurencyjnym;  – układa programy na podstawie tych algorytmów;  – wskazuje instrukcje, które decydują o itera- cyjności lub rekuren- cyjności podejścia do realizacji algorytmu. | – rozwiązuje problemy obiema metodami i ocenia ich skuteczność. |
| 47. Sortowanie bąbelkowe, czyli każda liczba jest mniejsza od maksymalnej lub jej równa | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia budowę tablicy jedno- wymiarowej;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia istotę sortowania bąbelkowego. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje działanie algorytmu sortowania bąbelkowego w postaci listy kroków i schematu blokowego;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje przykład sprawdzający poprawność działania algorytmu. | – samodzielnie omawia istotę metody sortowania bąbelkowego;  – samodzielnie omawia działanie przykładowego algorytmu opartego o metodę sortowania bąbelkowego;  – samodzielnie sprawdza działanie algorytmu na przykładach. | – układa algorytm sortowania bąbelkowego;  – weryfikuje poprawność działania programu na przykładach. | – układa program sortujący metodą bąbelkową w innym języku niż C++ (np. Java). |
| 48. Przez wstawianie, czyli jeszcze o porządkowaniu liczb | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia istotę sortowania przez wstawianie. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje działanie algorytmu sortowania bąbelkowego w postaci listy kroków i schematu przez wstawianie;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje przykład sprawdzający poprawność działania algorytmu. | – samodzielnie omawiana przykładzie istotę metody sortowania przez wstawianie;  – samodzielnie omawia działanie przykładowego algorytmu opartego o metodę sortowania przez wstawianie;  – samodzielnie sprawdza działanie algorytmu na przykładach. | – układa algorytm sortowania przez wstawianie w postaci listy kroków i schematu blokowego;  – weryfikuje poprawność działania programu na przykładach. | – układa program sortujący metodą przez wstawianie w innym języku niż C++ (np. Java). |
| 49. Komputer porządkuje, czyli układamy programy sortujące | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł weryfikuje działanie przykładowych programów (np. z pod- ręcznika). | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje przykładowe programy i wskazuje miejsca, w których są wykonywane działania najważniejsze dla metody. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł układa programy sortujące;  – na podstawie podręcz- nika uzasadnia użycie danych instrukcji w przykładowych programach. | – samodzielnie układa programy sortujące według obu metod;  – samodzielnie weryfikuje poprawność działania tych programów;  – wskazuje instrukcje i rozkazy realizujące istotę metody. | – proponuje inne od przykładowych rozwiązania programowe;  – układa programy realizujące algorytmy sortowania bąbelkowego i przez wstawianie według własnego pomysłu. |
| 50. Fibonacci i jego wzór, czyli generujemy kolejne liczby ciągu | – wie, kim był i kiedy żył Fibonacci;  – zna jego zasługi dla rozwoju cywilizacji. | – na podstawie podręcz- nika lub innych wiarygod- nych źródeł omawia na przykładzie istotę ciągu Fibonacciego;  – na podstawie podręcz- nika lub innych wiarygod- nych źródeł analizuje działanie przykładowego algorytmu obliczającego kolejne elementy ciągu, zapisanego w postaci schematu blokowego. | – samodzielnie omawia na przykładzie istotę ciągu Fibonacciego;  – samodzielnie analizuje działanie przykładowego algorytmu obliczającego kolejne elementy ciągu, zapisanego w postaci schematu blokowego;  – układa algorytm obliczający określoną liczbę liczb ciągu Fibonacciego;  – analizuje program w języku C++ ułożony według przykładowego algorytmu. | – układa program w języku C++ obliczający *n* kolejnych elementów ciągu Fibonacciego;  – testuje poprawność działania swojego programu na przykładzie. | – proponuje rozwiązanie problemu obliczania *n*-tego elementu ciągu Fibonacciego. |
| 51. Szukamy lidera, czyli porządkowanie zbioru metodą binarnego wyszukiwania | – omawia, na czym polega metoda dziel i zwyciężaj;  – rozumie, na czym polega umieszczanie danych w tablicy. | – na podstawie podręcz- nika opisuje zastosowanie wyszukiwania binarnego w sortowaniu. | – omawia na przykładzie działanie algorytmu sortowania zbioru z wykorzystaniem wyszukiwania binarnego. | – testuje algorytm sortowania zbioru z wykorzystaniem wyszukiwania binarnego na dowolnych danych;  – układa pogram na podstawie algorytmu. | – stosuje algorytm dla tablicy wielowymiarowej. |
| 52. Przecinając oś, czyli wyznaczenie miejsc zerowych funkcji metodą połowienia | – podaje definicję miejsca zerowego funkcji i wskazuje je na wykresie. | – na podstawie podręcz- nika omawia metodę połowienia. | – analizuje algorytm metody połowienia z podręcznika. | – na podstawie opisu metody opracowuje algorytm metody połowienia i na jego podstawie układa program w języku C++;  – testuje algorytm i program. | – omawia inne metody znajdowania miejsca zerowego, np. metodę analityczną. |
| 53. Prawie robi różnicę, czyli obliczanie przybliżonej wartości pierwiastka kwadratowego | – podaje, czym jest przybliżenie wartości pierwiastka i zna konsekwencje jego stosowania. | – na podstawie podręcz- nika omawia metodę Newtona-Robsona. | – omawia podstawy matematyczne metody Newtona-Robsona;  – analizuje omówiony w podręczniku algorytm i program metody Newtona-Robsona. | – na podstawie opisu metody samodzielnie opracowuje algorytm metody Newtona- -Robsona;  – układa program znajdujący przybliżoną wartość pierwiastka kwadratowego i go testuje. | – omawia inne metody obliczania wartości pierwiastka kwadratowego. |
| 54. Szybkie potęgowanie liczb w wersji iteracyjnej, czyli jak obliczyć potęgę o wykładniku naturalnym | – podaje właściwości potęgi liczby 0;  – własnymi słowami opisuje, czym jest potęgowanie liczb i jak się je wykonuje;  – uruchamia program z podręcznika i testuje jego działanie;  – podaje, na czym polega iteracyjne podejście do rozwiązywania problemów. | – na podstawie podręcz- nika omawia algorytm iteracyjny metody szybkiej potęgowania i porównuje ją z metodą kolejnego mnożenia przez podstawę potęgi. | – samodzielnie omawia metodę szybkiego potęgowania w wersji iteracyjnej;  – analizuje algorytm i program w języku C++ metody szybkiego potęgowania. | – omawia na przykładzie zalety metody szybkiego potęgowania;  – układa algorytm metody szybkiego potęgowania i na jego podstawie program w języku C++ w wersji iteracyjnej. | – porównuje podejście iteracyjne i rekurencyjne do obliczania potęgi metodą szybką. |
| 55. Szybkie potęgowanie liczb w wersji rekurencyjnej, czyli jak obliczyć potęgę o wykładniku naturalnym | – uruchamia program z podręcznika i testuje jego działanie;  – podaje, na czym polega rekurencyjne podejście do rozwiązywania problemów. | – na podstawie podręcz- nika omawia rekurencyjne podejście do metody potęgowania szybkiego. | – samodzielnie omawia metodę szybkiego potęgowania w wersji rekurencyjnej;  – analizuje algorytm i program w języku C++ lub Java dla metody szybkiego potęgowania rekurencyjnego. | – układa program metody szybkiego potęgowania i na jego podstawie program w języku C++ lub Java w wersji rekurencyjnej. | – porównuje podejście iteracyjne i rekurencyjne do obliczania potęgi metodą szybką. |
| 56. Suma binarna, czyli dodawanie w systemie dwójkowym | – omawia podstawy systemu dwójkowego;  – dodaje do siebie pisemnie niewielkie liczby binarne. | – zamienia reprezentacje liczb binarnych na dziesiętne i odwrotnie. | – układa algorytm dodawania binarnego i go testuje. | – układa program w języku C++ realizujący algorytm dodawania liczb binarnych i go testuje. | – układa program na podstawie algorytmu także w innym języku (np. Java). |
| 57. Różnica binarna, czyli odejmowanie w systemie dwójkowym | – odejmuje pisemnie niewielkie liczby binarne. | – opisuje metodę odejmowania liczb binarnych i tłumaczy rolę pożyczki na przykładzie. | – układa algorytm odejmowania binarnego i go testuje. | – układa program w języku C++ realizujący algorytm odejmowania liczb binarnych i go testuje. | – układa program na podstawie algorytmu także w innym języku (np. Java). |
| 58. Iloczyn binarny, czyli mnożenie w systemie dwójkowym | – mnoży pisemnie niewielkie liczby binarne. | – opisuje własnymi słowami na przykładzie metodę mnożenia liczb binarnych. | – układa algorytm mnożenia binarnego i go testuje. | – układa program w języku C++ realizujący algorytm mnożenia liczb binarnych i go testuje. | – układa program na podstawie algorytmu także w innym języku (np. Java). |
| 59. Iloraz binarny, czyli dzielenie w systemie dwójkowym | – dzieli pisemnie niewielkie liczby binarne. | – opisuje własnymi słowami na przykładzie metodę dzielenia liczb binarnych. | – układa algorytm dzielenia binarnego i go testuje. | – układa program w języku C++ realizujący algorytm dzielenia liczb binarnych i go testuje. | – układa program na podstawie algorytmu także w innym języku (np. Java). |
| 60. Metoda Monte Carlo, czyli jak obliczyć przybliżoną wartość liczby *pi* | – omawia znaczenie liczby *pi* na przykładach (np. obliczania obwodu koła). | – opisuje własnymi słowami na przykładzie, korzystając z rysunku pomocniczego, metodę Monte Carlo obliczania kolejnych pozycji liczby *pi*;  – zna wzór na liczbę *pi* z metody Monte Carlo. | – układa algorytm w dowolnym zapisie obliczający kolejne pozycje liczby *pi* metodą Monte Carlo. | – układa program w języku C++ realizujący algorytm Monet Carlo. | – układa program na podstawie algorytmu także w innym języku (np. Java). |