

**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:
<b>I. Opracowania rozwiązań problemów wybranymi aplikacjami</b>					
1. Host i serwer, czyli o działaniu sieci	<ul style="list-style-type: none"> <li>– na podstawie podręcznika i z pomocą nauczyciela omawia podstawowe pojęcia związane z budową sieci komputerowych (np. <i>host, serwer, klient, LAN, WAN, WiFi</i>);</li> <li>– wie, czym są media transmisyjne;</li> <li>– wymienia nazwy najczęściej spotykanych topologii sieci lokalnej i na podstawie ilustracji (np. z podręcznika) wymienia ich podstawowe cechy.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia znaczenie podstawowych pojęć związanych z budową i działaniem sieci komputerowych;</li> <li>– na podstawie podręcznika omawia cechy poszczególnych mediów transmisyjnych;</li> <li>– samodzielnie wymienia nazwy najczęściej spotykanych topologii sieci lokalnej i wymienia ich podstawowe cechy;</li> <li>– wymienia jednostki przesyłu informacji i omawia zależności pomiędzy nimi;</li> <li>– wie, że do różnych zastosowań można używać różnych rodzajów kabla sieciowego.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia znaczenie wszystkich pojęć związanych z budową i działaniem sieci komputerowych;</li> <li>– oblicza maksymalną ilość danych przesyłanych w sieci na podstawie danych o czasie i przepustowości łącza;</li> <li>– z niewielką pomocą nauczyciela wymienia nazwy i cechy podstawowych kabli dla sieci lokalnych CAT 3, 5 i 5e, 6 i 7;</li> <li>– wymienia podstawowe dla transmisji danych cechy światłowodu.</li> </ul>	<ul style="list-style-type: none"> <li>– wybiera odpowiednią topologię i medium dla różnych sieci lokalnych;</li> <li>– identyfikuje topologię istniejącej sieci (np. w pracowni);</li> <li>– identyfikuje rodzaje mediów transmisyjnych;</li> <li>– prawidłowo stosuje poznane pojęcia dotyczące sieci lokalnych.</li> </ul>	<ul style="list-style-type: none"> <li>– omawia niewymienione w podstawie programowej zagadnienia związane z tematem.</li> </ul>

<p>2. Włączamy do ruchu, czyli konfiguracja lokalnej sieci komputerowej</p>	<ul style="list-style-type: none"> <li>– omawia funkcję routera w sieci lokalnej;</li> <li>– wie, że router należy odpowiednio skonfigurować;</li> <li>– podaje, jak znaleźć opis konfiguracji routera (np. instrukcję, samouczek w sieci);</li> <li>– identyfikuje nazwę i producenta routera.</li> </ul>	<ul style="list-style-type: none"> <li>– omawia funkcję DHCP w kontekście konfiguracji routera;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł wyjaśnia pojęcia: <i>DHCP</i>, <i>serwer</i>, <i>szyfrowanie</i> (np. <i>WPA2-PSK</i>) i ich znaczenie dla konfiguracji routera;</li> <li>– na podstawie instrukcji i z pomocą nauczyciela omawia lub przeprowadza konfigurację przykładowego routera.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie instrukcji omawia lub przeprowadza konfigurację przykładowego routera;</li> <li>– na podstawie instrukcji lub opcji Pomocy omawia lub przeprowadza konfigurację przykładowego routera bez wykorzystania DHCP.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia lub przeprowadza konfigurację przykładowego routera z wykorzystaniem DHCP i bez tego mechanizmu.</li> </ul>	<ul style="list-style-type: none"> <li>– dokonuje konfiguracji zaawansowanych funkcji routera (np. uprawnień poszczególnych użytkowników).</li> </ul>
<p>3. Nie tylko WiFi, czyli bezprzewodowy dostęp do internetu</p>	<ul style="list-style-type: none"> <li>– wie czym się różni WLAN od WiFi;</li> <li>– omawia zastosowanie złączy Bluetooth do połączenia urządzeń komputerowych;</li> <li>– z niewielką pomocą nauczyciela przeprowadza przesłanie pliku (np. z telefonu do laptopa) za pośrednictwem Bluetooth.</li> </ul>	<ul style="list-style-type: none"> <li>– wymienia nazwy i omawia przeznaczenie podstawowych mediów i urządzeń transmisyjnych w tym: WLAN, WiFi, Bluetooth;</li> <li>– odróżnia i prawidłowo przypisuje do danych mediów ich symbole graficzne.</li> </ul>	<ul style="list-style-type: none"> <li>– wyjaśnia, na czym polega łączenie urządzeń za pomocą fal radiowych i podaje, jakie warunki muszą być przy tym spełnione;</li> <li>– na podstawie opisów omawia różnice między mediami transmisyjnymi;</li> <li>– wyjaśnia znaczenie norm określających ramy działania poszczególnych mediów transmisyjnych.</li> </ul>	<ul style="list-style-type: none"> <li>– wymienia pasma radiowe przeznaczone dla mediów transmisyjnych stosowanych w sieciach bezprzewodowych;</li> <li>– opisuje możliwości wykorzystania mediów bezprzewodowych i samodzielnie przeprowadza konfigurację podłączanych do nich urządzeń (np. słuchawek, telefonu, laptopa).</li> </ul>	<ul style="list-style-type: none"> <li>– omawia zastosowania i działanie WiMAX;</li> <li>– samodzielnie łączy klika urządzeń jednocześnie za pomocą Bluetooth.</li> </ul>
<p>4. Zespół, czyli realizujemy projekty w chmurze</p>	<ul style="list-style-type: none"> <li>– omawia budowę przykładowej chmury, wymieniając jej podstawowe składniki.</li> </ul>	<ul style="list-style-type: none"> <li>– przedstawia i omawia przykładowy schemat struktury chmury informatycznej;</li> <li>– wymienia najczęściej spotykane w chmurach</li> </ul>	<ul style="list-style-type: none"> <li>– omawia przeznaczenie poszczególnych elementów chmury użytej na zajęciach;</li> <li>– porównuje działanie programów z chmury z ich</li> </ul>	<ul style="list-style-type: none"> <li>– biegle posługuje się programami i dyskami sieciowymi;</li> <li>– samodzielnie przygotowuje chmurę do pracy zespołu, m.in.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie planuje i organizuje pracę zespołu w chmurze;</li> <li>– posługuje się kilkoma chmurami (np. Google i OneDrive).</li> </ul>

		narzędzia (m.in. dysk, edytor, arkusz, kalendarz); – wie, że chmurę można wykorzystać do pracy zespołowej.	odpowiednikami z aplikacji 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.	konfiguruje kalendarz, udostępnia foldery i pliki.	
5. Tutorial, czyli jak stworzyć 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 podręcznika 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 podręcznika tworzy konspekt dokumentu; – na podstawie podręcznika 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 podręcznika tworzy spisy ilustracji i tabel.	– na podstawie podręcznika lub tutoriali zmienia domyślne opcje edytora; – na podstawie podręcznika lub tutoriali wykonuje automatyczne spisy	– samodzielnie zmienia domyślne opcje edytora; – samodzielnie wykonuje automatyczne spisy treści, tabel i ilustracji; – samodzielnie aktualizuje	– spełnia kryteria oceny bardzo dobrej dla innych edytorów (np. LibreOffice).

			treści, tabel i ilustracji; – na podstawie podręcznika lub tutoriali aktualizuje spisy po zmianach.	spisy po zmianach; – decyduje, które z opcji domyślnych zmienić i uzasadnia swój wybór.	
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 podręcznika używa opcji Recenzja w Word i wstawiania komentarzy w dokumencie PDF.	– na podstawie podręcznika lub tutoriali używa opcji Recenzja do wpisywania komentarzy w Word oraz opcji wstawiania komentarzy i zaznaczania fragmentów w dokumencie PDF; – na podstawie podręcznika 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 proponowania zmian w dokumencie; – analizuje wynik działania 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 podręcznika wyjaśnia, czym jest i do czego służy technika OLE.	– wyjaśnia, jakie obiekty mogą być wstawiane za pomocą techniki OLE; – na podstawie podręcznika lub opcji Pomoc umieszcza w dokumencie obiekty OLE; – na podstawie podręcznika wyjaśnia, czym są OLE2 i ActiveX; – wyjaśnia, dlaczego podczas używania ActiveX należy zachować szczególną ostrożność.	– na podstawie podręcznika 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).

<b>II. Rozwijanie kompetencji społecznych</b>					
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 tłumaczem języków.	– ocenia jakość aplikacji ratujące życie lub zdrowie; – ocenia jakość aplikacji ułatwiających funkcjonowanie osobom niepełnosprawnym i starszym; – wymienia cechy strony dostosowanej do potrzeb osób niepełnosprawnych; – używa urządzeń monitorujących stan organizmu.	– dobiera urządzenia 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 tłumacza 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 podręcznika 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 korzystając z podręcznika, umieszcza 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.
<b>III. Bazy danych</b>					
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 podręcznika 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 postępuje 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 <i>kwerenda</i> ; – wyjaśnia, na czym polega filtrowanie w informatyce.	– na podstawie podręcznika tworzy kwerendę, używając kreatora; – na podstawie podręcznika lub tutoriali modyfikuje kwerendę, dodając filtrowanie; – na podstawie podręcznika 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 różnych rodzajów formularzy i na podstawie podręcznika 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 podręcznika lub tutoriali tworzy w Excelu tabele z nagłówkami;	– samodzielnie tworzy w Excelu tabele z nagłówkami; – samodzielnie używa	– planuje i wykonuje czynności tworzenia bazy w Excelu; – dobiera kryteria	– czynności opisane dla Excels wykonuje także w innym arkuszu (np. LibreOffice Calc).

		<ul style="list-style-type: none"> <li>– na podstawie podręcznika lub tutoriali używa formularzy do wprowadzania danych do tabel;</li> <li>– na podstawie podręcznika lub tutoriali tworzy listę rozwijaną.</li> </ul>	<ul style="list-style-type: none"> <li>formularzy do wprowadzania danych do tabel;</li> <li>– samodzielnie tworzy listę rozwijaną;</li> <li>– stosuje sortowanie według wskazanych kryteriów;</li> <li>– wstawia narzędzia do paska Szybki dostęp.</li> </ul>	<ul style="list-style-type: none"> <li>sortowania;</li> <li>– przenosi tabele z bazy danych do Excela.</li> </ul>	
21. Wiedza w sieci, czyli internet mądrych ludzi	<ul style="list-style-type: none"> <li>– wyszukuje informacje w domyślnej wyszukiwarce przeglądarki internetowej;</li> <li>– omawia znaczenie zachowania praw autorskich i podstawy licencji CC.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika zmienia domyślną wyszukiwarkę w przeglądarce Firefox;</li> <li>– na podstawie podręcznika uszczegółowia dane w wyszukiwarce w celu zwiększenia prawdopodobieństwa dotarcia do szukanej informacji;</li> <li>– na podstawie podręcznika wyszukuje za pomocą obrazu.</li> </ul>	<ul style="list-style-type: none"> <li>– uzasadnia celowość korzystania z różnych wyszukiwarek;</li> <li>– pozyskuje informacje z baz danych, takich jak europeana.eu;</li> <li>– tłumaczy podstawy i cel działania botów indeksujących;</li> <li>– tłumaczy na przykładzie konieczność uszczegółowiania zapytań do wyszukiwarki;</li> <li>– korzysta z zaawansowanych opcji wyszukiwarek;</li> <li>– ocenia wiarygodność źródeł, wymieniając świadczące o niej cechy.</li> </ul>	<ul style="list-style-type: none"> <li>– planuje użycie odpowiedniej wyszukiwarki;</li> <li>– wybiera bazę danych do wyszukiwania konkretnych informacji;</li> <li>– trafnie dobiera słowa kluczowe, zarówno w opcjach głównych, jak i zaawansowanych wyszukiwarek;</li> <li>– zmienia wyszukiwarki domyślne w różnych przeglądarkach internetowych.</li> </ul>	<ul style="list-style-type: none"> <li>– omawia cechy różnych wyszukiwarek internetowych i ocenia ich przydatność do konkretnego zadania.</li> </ul>
22. Wirtualne serwery, czyli instalujemy Apache	<ul style="list-style-type: none"> <li>– omawia pojęcie <i>serwer</i>;</li> <li>– uruchamia środowisko XAMP.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika i doświadczeń z poprzednich lekcji instaluje i uruchamia środowisko XAMP.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie instaluje środowisko XAMP, wybierając wskazane komponenty;</li> <li>– omawia zawartość XAMP i jego zastosowanie.</li> </ul>	<ul style="list-style-type: none"> <li>– decyduje o wyborze komponentów na podstawie informacji o późniejszym wykorzystaniu XAMP.</li> </ul>	<ul style="list-style-type: none"> <li>– instaluje i korzysta także z innych środowisk symulujących działanie serwera oraz omawia ich cechy.</li> </ul>



23. Język zapytań, czyli poznajemy język SQL	– wyjaśnia, do czego służy język SQL.	– na podstawie podręcznika wymienia i krótko charakteryzuje podstawowe elementy składni SQL – DML, DDL, DCL, DQL; – na podstawie podręcznika lub innych publikacji omawia typy danych języka SQL.	– na podstawie podręcznika 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 podręcznika omawia definicję MySQL jako otwarto-źródłowego systemu zarządzania bazami danych; – na podstawie podręcznika uruchamia phpMyAdmin i jego narzędzia z pakietu XAMP; – na podstawie podręcznika i z niewielką pomocą nauczyciela tworzy nową bazę i tabele w MySQL; – na podstawie podręcznika i z niewielką pomocą nauczyciela tworzy zapytania w SQL.	– na podstawie podręcznika 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 podręcznika omawia składnię polecenia SELECT; – na podstawie podręcznika omawia działanie klauzuli WHERE	– samodzielnie omawia składnię polecenia SELECT; – samodzielnie omawia działanie klauzuli WHERE i operatorów logicznych;	– wykorzystuje zapytanie SELECT i operatory logiczne podczas tworzenia zapytań; – łączy tabele w celu przeszukania większej ich	– operuje MySQL i układa programy w SQL w przykładach innych niż poznane w trakcie lekcji.

		<p>i operatorów logicznych;</p> <ul style="list-style-type: none"> <li>– na podstawie podręcznika omawia efekt łączenia tabel w celu wyszukiwania informacji;</li> <li>– na podstawie podręcznika analizuje umieszczone w nim przykłady w SQL.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia efekt łączenia tabel w celu wyszukiwania informacji;</li> <li>– samodzielnie analizuje umieszczone w podręczniku przykłady w SQL;</li> <li>– sporadycznie korzystając ze źródeł wiedzy, wykonuje ćwiczenia z podręcznika.</li> </ul>	<p>liczby;</p> <ul style="list-style-type: none"> <li>– wyjaśnia, jak szukać informacji w tabelach niepołączonych ze sobą.</li> </ul>	
26. Podzapytania, czyli instrukcje modyfikujące dane w języku SQL	<ul style="list-style-type: none"> <li>– na podstawie podręcznika wyjaśnia pojęcie <i>podzapytania</i>.</li> </ul>	<ul style="list-style-type: none"> <li>– wyjaśnia pojęcie <i>podzapytania</i>;</li> <li>– na podstawie podręcznika omawia znaczenie podzapytań.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika wykonuje opisane w nim ćwiczenie;</li> <li>– wyjaśnia istotę podzapytań wierszowych i skorelowanych;</li> <li>– wyjaśnia istotę podzapytań o charakterze tabeli tymczasowej;</li> <li>– omawia przykłady programów SQL realizujących podzapytania.</li> </ul>	<ul style="list-style-type: none"> <li>– układa programy podzapytań w SQL do przykładów z podręcznika lub podanych przez nauczyciela.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie układa programy podzapytań w SQL.</li> </ul>
27. Modyfikacje, czyli zarządzanie bazą danych	<ul style="list-style-type: none"> <li>– na podstawie podręcznika wyjaśnia działanie poleceń: UPDATE, ALTER TABLE, DELETE, DELETE FROM.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika omawia składnię poleceń: UPDATE, ALTER TABLE, DELETE, DELETE FROM.</li> </ul>	<ul style="list-style-type: none"> <li>– z pomocą nauczyciela lub na podstawie podręcznika modyfikuje elementy bazy danych.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie, według zaleceń modyfikuje elementy (zmienia strukturę, usuwa rekordy, tabele) i strukturę bazy danych.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie określa zakres modyfikacji bazy ze względu na zaistniałe potrzeby i modyfikuje elementy i strukturę bazy danych.</li> </ul>
28. Bezpieczeństwo bazy, czyli tworzymy kopie zapasowe	<ul style="list-style-type: none"> <li>– omawia zasady dbania o bezpieczeństwo baz danych;</li> <li>– podaje, czym jest kopia zapasowa.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika omawia istotę najczęściej spotykanych ataków na bazę (pasywne i aktywne);</li> </ul>	<ul style="list-style-type: none"> <li>– omawia na przykładach definicje ataków pasywnych i aktywnych;</li> <li>– omawia zasady, jakimi powinien się kierować</li> </ul>	<ul style="list-style-type: none"> <li>– tworzy kopię bezpieczeństwa, używając odpowiednich poleceń;</li> <li>– wyjaśnia pojęcie przyrostowej kopii</li> </ul>	<ul style="list-style-type: none"> <li>– planuje zabezpieczenie bazy danych.</li> </ul>

		– wymienia zasady, jakimi powinien się kierować administrator bazy w celu zabezpieczenia jej przed nieuprawnionym dostępem lub skasowaniem danych.	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.	bezpieczeństwa; – odzyskuje dane z kopii bezpieczeństwa.	
<b>IV. Programowanie stron internetowych</b>					
29. HTML, czyli przeglądarka interpretuje język programowania stron	– omawia przeznaczenie języka HTML; – omawia rolę znaczników w języku HTML.	– omawia rolę przeglądarki internetowej w kontekście języka HTML; – instaluje, uruchamia i korzysta ze wskazanego edytora (np. Notepad+) skonfigurowanego dla HTML5; – korzysta z opisów najczęściej używanych znaczników (np. z podręcznika).	– na podstawie podręcznika lub innych wiarygodnych źródeł konfiguruje edytor do układania programów w HTML5; – na podstawie podręcznika lub innych wiarygodnych źródeł układa proste programy stron, np. wyświetlające tekst z zachowaniem prawidłowej 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	– na podstawie podręcznika uruchamia programy z przykładów; – na podstawie przykładów omawia rolę list,	– na podstawie podręcznika lub innych wiarygodnych źródeł tworzy w kodzie HTML listy uporządkowane	– samodzielnie tworzy w kodzie HTML listy uporządkowane i nieuporządkowane oraz omawia ich znaczenie;	– tworzy rozbudowane strony internetowe w języku HTML5.

	(np. z przykładu z podręcznika).	definicji i tabel; – tłumaczy istotę i przeznaczenie hipertekstu.	i nieuporządkowane oraz omawia ich znaczenie; – na podstawie podręcznika lub innych wiarygodnych źródeł tworzy listy definicji z dodanymi nagłówkami; – na podstawie podręcznika lub innych wiarygodnych źródeł tworzy tabele w HTML.	– samodzielnie tworzy listy definicji z dodanymi nagłówkami; – samodzielnie tworzy tabele w HTML; – planuje użycie odpowiednich konstrukcji do projektowanej strony; – używa atrybutów w konstrukcji hipertekstu.	
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 podręcznika wstawia przygotowaną przez nauczyciela grafikę do gotowego kodu strony; – na podstawie podręcznika 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 podręcznika lub innych wiarygodnych źródeł omawia zasady łączenia CSS z HTML; – na podstawie podręcznika lub innych wiarygodnych źródeł układa proste pliki stylów CSS, formatujące sposób wyświetlania tekstu; – na podstawie podręcz-	– 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	– tworzy rozbudowane strony internetowe w języku HTML5 z zastosowaniem CSS.

			<p>nika lub innych wiarygodnych źródeł wyjaśnia znaczenie nagłówków i bloków w stosowaniu CSS;</p> <p>– na podstawie podręcznika lub innych wiarygodnych źródeł wyjaśnia znaczenie sekcji i selektora.</p>	<p>znaczenie sekcji i selektora;</p> <p>– projektuje użycie CSS do formatowania stylu tekstu na stronie.</p>	
33. Pliki stylów, czyli CSS w akcji	<p>– objaśnia działanie i znaczenie odsyłaczy na stronie w HTML;</p> <p>– omawia rolę menu na stronie.</p>	<p>– na podstawie podręcznika lub innych wiarygodnych źródeł omawia znaczenie selektorów i ich atrybutów w pliku CSS;</p> <p>– na podstawie podręcznika lub innych wiarygodnych źródeł omawia pojęcie <i>walidacja</i>.</p>	<p>– na podstawie podręcznika lub innych wiarygodnych źródeł tworzy pliki HTML i CSS z menu z przyciskami, wykorzystując przykłady;</p> <p>– na podstawie podręcznika lub innych wiarygodnych źródeł używa pseudoklas z przykładu do określenia interakcji klawisza z kursorem;</p> <p>– na podstawie podręcznika lub innych wiarygodnych źródeł rozmieszcza elementy strony, tworząc jej prosty layout.</p>	<p>– samodzielnie tworzy pliki HTML i CSS z menu z przyciskami, wykorzystując przykłady;</p> <p>– samodzielnie używa pseudoklas z przykładu do określenia interakcji klawisza z kursorem;</p> <p>– samodzielnie rozmieszcza elementy strony, tworząc jej prosty layout;</p> <p>– zmienia wygląd i atrybuty menu z przykładu;</p> <p>– zmienia położenie elementów strony;</p> <p>– stosuje różną orientację elementów menu;</p> <p>– przeprowadza walidację strony.</p>	<p>– tworzy rozbudowane strony internetowe w języku HTML5 z zastosowaniem CSS zawierające menu i efekty interakcji z kursorem.</p>
34. Treści mogą się zmieniać, czyli elementy dynamiczne na stronie internetowej	<p>– wskazuje elementy istniejących stron, które zmieniają się dynamicznie;</p> <p>– na podstawie podręcznika wskazuje różnice</p>	<p>– omawia sposoby wklejania w kod strony gotowych odnośników do elementów dynamicznych;</p> <p>– na podstawie podręcz-</p>	<p>– tworzy element dynamiczny zewnętrzny za pomocą darmowego edytora aplikacji (np. learningapps.com);</p> <p>– pobiera i umieszcza na</p>	<p>– planuje umieszczenie na stronie widżetów i innych elementów dynamicznych;</p> <p>– dokonuje prawidłowego wyboru widżetów;</p>	<p>– tworzy strony z własnymi elementami dynamicznymi.</p>

	między statyczną a dynamiczną stroną internetową.	nika umieszcza w kodzie strony elementy dynamiczne.	stronie kod uruchamiający widżety lub aplikacje internetowe z learningapps.com.	– projektuje elementy dynamiczne (np. krzyżówki) w learningapps.com.	
35. Widoczna w internecie, czyli jak opublikować stronę	<ul style="list-style-type: none"> <li>– omawia funkcje domeny internetowej;</li> <li>– wskazuje przynajmniej jedną firmę (stronę internetową) zajmującą się rejestracją domen.</li> </ul>	<ul style="list-style-type: none"> <li>– omawia proces wczytywania strony internetowej do przeglądarki i rolę domeny;</li> <li>– omawia znaczenie niektórych domen (np. .com, .pl, .edu);</li> <li>– podaje, jakie usługi musi uruchomić lub zamówić, by strona była widoczna w sieci.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł opisuje na przykładzie proces rezerwacji domeny;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł zakłada konta na darmowym serwerze z usługą hostingową (np. cba.pl);</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł wysyła pliki strony do serwera www.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie opisuje na przykładzie proces rezerwacji domeny;</li> <li>– samodzielnie zakłada konta na darmowym serwerze z usługą hostingową (np. cba.pl);</li> <li>– samodzielnie wysyła pliki strony do serwera www;</li> <li>– sprawdza zajętość domen, którymi jest zainteresowany;</li> <li>– korzysta z klienta FTP nie tylko do przesyłania plików strony.</li> </ul>	– podaje, jak zarządzać domeną i usługą hostingową na przykładzie swojego konta.
36. CMS, czyli system zarządzania treścią strony internetowej	<ul style="list-style-type: none"> <li>– omawia definicję CMS-u;</li> <li>– podaje, do czego służy CMS i wymienia kilka najpopularniejszych (w tym Joomla i WordPress).</li> </ul>	<ul style="list-style-type: none"> <li>– omawia proces przygotowań do instalacji CMS-u;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł instaluje środowisko serwerowe (np. Laragon), a w nim CMS (np. WordPress);</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł loguje się do panelu sterowania CMS-em (np. WordPress).</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie instaluje środowisko serwerowe (np. Laragon), a w nim CMS (np. WordPress);</li> <li>– samodzielnie loguje się do panelu sterowania CMS-em (np. WordPress).</li> </ul>	<ul style="list-style-type: none"> <li>– dobiera odpowiedni do tematu strony CMS i uzasadnia swój wybór;</li> <li>– wyjaśnia konieczność instalacji bazy danych dla CMS-a;</li> <li>– używa wirtualnego środowiska serwera do testowania działania różnych stron internetowych.</li> </ul>	– buduje ciekawe strony internetowe za pomocą różnych CMS-ów.

<p>37. Panel i skórki, czyli tworzymy stronę w CMS</p>	<ul style="list-style-type: none"> <li>– wyjaśnia znaczenie szablonu (skórki) w systemie CMS;</li> <li>– przegląda oferowane w sieci skórki dla WordPressa.</li> </ul>	<ul style="list-style-type: none"> <li>– odróżnia motywy od szablonów;</li> <li>– omawia znaczenie i funkcję motywów w szablonach;</li> <li>– na podstawie podręcznika wypełnia treścią stronę zbudowaną na bazie szablonu.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł instaluje w CMS-ie wskazany szablon;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł dobiera motyw i zmienia go w razie potrzeby;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł modyfikuje motyw skórki;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł wypełnia treścią stronę, tworząc nowe wpisy;</li> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł umieszcza na stronie elementy graficzne.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie instaluje w CMS-ie wskazany szablon;</li> <li>– samodzielnie dobiera motyw i zmienia go w razie potrzeby;</li> <li>– samodzielnie modyfikuje motyw skórki;</li> <li>– samodzielnie wypełnia treścią stronę, tworząc nowe wpisy;</li> <li>– samodzielnie umieszcza na stronie elementy graficzne;</li> <li>– trafnie i z uzasadnieniem dobiera szablon i motyw do treści przyszłej strony;</li> <li>– administruje stroną, wykorzystując panel sterowania;</li> <li>– dodaje i usuwa użytkowników CMS-a i świadomie nadaje im uprawnienia;</li> <li>– umieszcza na stronie filmy i inne elementy multimedialne.</li> </ul>	<ul style="list-style-type: none"> <li>– planuje i realizuje własne projekty stron na bazie CMS.</li> </ul>
<p>38. Szybkie i łatwe, czy programy do tworzenia stron internetowych</p>	<ul style="list-style-type: none"> <li>– podaje, jak zapisać tekstowy dokument Word w formacie HTML;</li> <li>– wymienia kilka systemów darmowych blogów.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika lub innych wiarygodnych źródeł eksportuje do formatu HTML dokumenty Word i Excel;</li> <li>– na podstawie</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie eksportuje do formatu HTML dokumenty Word i Excel;</li> <li>– samodzielnie zakłada konto w systemie darmowych blogów wskazanym przez</li> </ul>	<ul style="list-style-type: none"> <li>– prawidłowo i estetycznie rozmieszcza elementy graficzne, wzory itp. w eksportowanym dokumencie;</li> <li>– testuje w środowisku serwerowym (np. Lara-</li> </ul>	<ul style="list-style-type: none"> <li>– prowadzi własnego bloga na ciekawy temat.</li> </ul>

		podręcznika lub innych wiarygodnych źródeł zakłada konto w systemie darmowych blogów wskazanym przez nauczyciela.	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.	gon) strony utworzone podczas eksportu dokumentów do HTML, zachowując układ folderów.	
39. Pierwsze skrypty, czyli poznajemy PHP	– omawia specyfikę języka PHP (skrypt umieszczony w HTML i na odwrót); – uruchamia środowisko Notepad++.	– na podstawie podręcznika omawia umieszczenie 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 podręcznika 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 podręcznika analizuje przykładowy 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 nieskomplikowane 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 przykładowych skryptów PHP zawierających instrukcje warunkowe i wyboru.	– samodzielnie omawia działanie przykładowych skryptów PHP zawierających instrukcje warunkowe i wyboru, uruchamia je i testuje.	– układa według specyfikacji skryptu PHP zawierające instrukcje warunkowe i wyboru.	– układa na podstawie własnych specyfikacji i pomysłów skryptu 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 podobnych instrukcji języka C++; – wyjaśnia różnice między instrukcjami pętli.	– omawia działanie przykładowych skryptów PHP zawierających pętle; – analizuje przykłady skryptów zawierających pętle.	– układa według specyfikacji skryptu PHP zawierające pętle.	– układa na podstawie własnych specyfikacji i pomysłów skryptu 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 podręcznika wyjaśnia różnice między funkcjami wbudowanymi 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 złożonych 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 skryptu 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-	– samodzielnie analizuje działanie przykładowych skryptów PHP zawierających tablice.	– układa według specyfikacji skryptu zawierające tablice.	– układa według własnych specyfikacji i pomysłów skryptu PHP zawierające tablice.

		a wielowymiarowymi; – uruchamia przykładowe skrypty PHP zawierające tablice i na podstawie podręcznika omawia ich działanie.			
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 zawierających 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.
<b>V. Algorytmy i programowanie</b>					
46. Powtarzanie w pętli i wywołanie siebie, czyli iteracja i rekurencja w algorytmach	– omawia na realnych przykładach różnicę między rekurencją a iteracją.	– na podstawie podręcznika lub innych wiarygodnych źródeł analizuje przykładowe algorytmy iteracyjne i rekurencyjne; – na podstawie podręcznika lub innych wiarygodnych źródeł rozpoznaje procesy rekurencyjne i iteracyjne.	– samodzielnie analizuje przykładowe algorytmy iteracyjne i rekurencyjne; – samodzielnie rozpoznaje procesy rekurencyjne i iteracyjne; – wskazuje w przykładowych 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 iteracyjności lub rekurencyjności 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 podręcznika lub innych wiarygodnych źródeł omawia budowę tablicy jednowymiarowej;	– na podstawie podręcznika lub innych wiarygodnych źródeł analizuje działanie algorytmu sortowania bąbelkowego	– samodzielnie omawia istotę metody sortowania bąbelkowego; – samodzielnie omawia działanie przykładowego	– układa algorytm sortowania bąbelkowego; – weryfikuje poprawność działania programu na przykładach.	– układa program sortujący metodą bąbelkową w innym języku niż C++ (np. Java).

	– na podstawie podręcznika lub innych wiarygodnych źródeł omawia istotę sortowania bąbelkowego.	w postaci listy kroków i schematu blokowego; – na podstawie podręcznika lub innych wiarygodnych źródeł analizuje przykład sprawdzający poprawność działania algorytmu.	algorytmu opartego o metodę sortowania bąbelkowego; – samodzielnie sprawdza działanie algorytmu na przykładach.		
48. Przez wstawianie, czyli jeszcze o porządkowaniu liczb	– na podstawie podręcznika lub innych wiarygodnych źródeł omawia istotę sortowania przez wstawianie.	– na podstawie podręcznika lub innych wiarygodnych źródeł analizuje działanie algorytmu sortowania bąbelkowego w postaci listy kroków i schematu przez wstawianie; – na podstawie podręcznika lub innych wiarygodnych 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 podręcznika lub innych wiarygodnych źródeł weryfikuje działanie przykładowych programów (np. z podręcznika).	– na podstawie podręcznika lub innych wiarygodnych źródeł analizuje przykładowe programy i wskazuje miejsca, w których są wykonywane działania najważniejsze dla metody.	– na podstawie podręcznika lub innych wiarygodnych źródeł układa programy sortujące; – na podstawie podręcznika 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 podręcznika lub innych wiarygodnych źródeł omawia na przykładzie istotę ciągu	– samodzielnie omawia na przykładzie istotę ciągu Fibonacciego; – samodzielnie analizuje	– układa program w języku C++ obliczający $n$ kolejnych elementów ciągu Fibonacciego;	– proponuje rozwiązanie problemu obliczania $n$ -tego elementu ciągu Fibonacciego.

		<p>Fibonacciego;</p> <p>– na podstawie podręcznika lub innych wiarygodnych źródeł analizuje działanie przykładowego algorytmu obliczającego kolejne elementy ciągu, zapisanego w postaci schematu blokowego.</p>	<p>działanie przykładowego algorytmu obliczającego kolejne elementy ciągu, zapisanego w postaci schematu blokowego;</p> <p>– układa algorytm obliczający określoną liczbę liczb ciągu Fibonacciego;</p> <p>– analizuje program w języku C++ ułożony według przykładowego algorytmu.</p>	<p>– testuje poprawność działania swojego programu na przykładzie.</p>	
<p>51. Szukamy lidera, czyli porządkowanie zbioru metodą binarnego wyszukiwania</p>	<p>– omawia, na czym polega metoda dziel i zwyciężaj;</p> <p>– rozumie, na czym polega umieszczanie danych w tablicy.</p>	<p>– na podstawie podręcznika opisuje zastosowanie wyszukiwania binarnego w sortowaniu.</p>	<p>– omawia na przykładzie działanie algorytmu sortowania zbioru z wykorzystaniem wyszukiwania binarnego.</p>	<p>– testuje algorytm sortowania zbioru z wykorzystaniem wyszukiwania binarnego na dowolnych danych;</p> <p>– układa program na podstawie algorytmu.</p>	<p>– stosuje algorytm dla tablicy wielowymiarowej.</p>
<p>52. Przecinając oś, czyli wyznaczenie miejsc zerowych funkcji metodą połowienia</p>	<p>– podaje definicję miejsca zerowego funkcji i wskazuje je na wykresie.</p>	<p>– na podstawie podręcznika omawia metodę połowienia.</p>	<p>– analizuje algorytm metody połowienia z podręcznika.</p>	<p>– na podstawie opisu metody opracowuje algorytm metody połowienia i na jego podstawie układa program w języku C++;</p> <p>– testuje algorytm i program.</p>	<p>– omawia inne metody znajdowania miejsca zerowego, np. metodę analityczną.</p>
<p>53. Prawie robi różnicę, czyli obliczanie przybliżonej wartości pierwiastka kwadratowego</p>	<p>– podaje, czym jest przybliżenie wartości pierwiastka i zna konsekwencje jego stosowania.</p>	<p>– na podstawie podręcznika omawia metodę Newtona-Robsona.</p>	<p>– omawia podstawy matematyczne metody Newtona-Robsona;</p> <p>– analizuje omówiony w podręczniku algorytm i program metody Newtona-Robsona.</p>	<p>– na podstawie opisu metody samodzielnie opracowuje algorytm metody Newtona-Robsona;</p> <p>– układa program znajdujący przybliżoną</p>	<p>– omawia inne metody obliczania wartości pierwiastka kwadratowego.</p>

				wartość pierwiastka kwadratowego i go testuje.	
54. Szybkie potęgowanie liczb w wersji iteracyjnej, czyli jak obliczyć potęgę o wykładniku naturalnym	<ul style="list-style-type: none"> <li>– podaje właściwości potęgi liczby 0;</li> <li>– własnymi słowami opisuje, czym jest potęgowanie liczb i jak się je wykonuje;</li> <li>– uruchamia program z podręcznika i testuje jego działanie;</li> <li>– podaje, na czym polega iteracyjne podejście do rozwiązywania problemów.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika omawia algorytm iteracyjny metody szybkiej potęgowania i porównuje ją z metodą kolejnego mnożenia przez podstawę potęgi.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia metodę szybkiego potęgowania w wersji iteracyjnej;</li> <li>– analizuje algorytm i program w języku C++ metody szybkiego potęgowania.</li> </ul>	<ul style="list-style-type: none"> <li>– omawia na przykładzie zalety metody szybkiego potęgowania;</li> <li>– układa algorytm metody szybkiego potęgowania i na jego podstawie program w języku C++ w wersji iteracyjnej.</li> </ul>	<ul style="list-style-type: none"> <li>– porównuje podejście iteracyjne i rekurencyjne do obliczania potęgi metodą szybką.</li> </ul>
55. Szybkie potęgowanie liczb w wersji rekurencyjnej, czyli jak obliczyć potęgę o wykładniku naturalnym	<ul style="list-style-type: none"> <li>– uruchamia program z podręcznika i testuje jego działanie;</li> <li>– podaje, na czym polega rekurencyjne podejście do rozwiązywania problemów.</li> </ul>	<ul style="list-style-type: none"> <li>– na podstawie podręcznika omawia rekurencyjne podejście do metody potęgowania szybkiego.</li> </ul>	<ul style="list-style-type: none"> <li>– samodzielnie omawia metodę szybkiego potęgowania w wersji rekurencyjnej;</li> <li>– analizuje algorytm i program w języku C++ lub Java dla metody szybkiego potęgowania rekurencyjnego.</li> </ul>	<ul style="list-style-type: none"> <li>– układa program metody szybkiego potęgowania i na jego podstawie program w języku C++ lub Java w wersji rekurencyjnej.</li> </ul>	<ul style="list-style-type: none"> <li>– porównuje podejście iteracyjne i rekurencyjne do obliczania potęgi metodą szybką.</li> </ul>
56. Suma binarna, czyli dodawanie w systemie dwójkowym	<ul style="list-style-type: none"> <li>– omawia podstawy systemu dwójkowego;</li> <li>– dodaje do siebie pisemnie niewielkie liczby binarne.</li> </ul>	<ul style="list-style-type: none"> <li>– zamienia reprezentacje liczb binarnych na dziesiętne i odwrotnie.</li> </ul>	<ul style="list-style-type: none"> <li>– układa algorytm dodawania binarnego i go testuje.</li> </ul>	<ul style="list-style-type: none"> <li>– układa program w języku C++ realizujący algorytm dodawania liczb binarnych i go testuje.</li> </ul>	<ul style="list-style-type: none"> <li>– układa program na podstawie algorytmu także w innym języku (np. Java).</li> </ul>
57. Różnica binarna, czyli odejmowanie w systemie dwójkowym	<ul style="list-style-type: none"> <li>– odejmuje pisemnie niewielkie liczby binarne.</li> </ul>	<ul style="list-style-type: none"> <li>– opisuje metodę odejmowania liczb binarnych i tłumaczy rolę pożyczki na przykładzie.</li> </ul>	<ul style="list-style-type: none"> <li>– układa algorytm odejmowania binarnego i go testuje.</li> </ul>	<ul style="list-style-type: none"> <li>– układa program w języku C++ realizujący algorytm odejmowania liczb binarnych i go testuje.</li> </ul>	<ul style="list-style-type: none"> <li>– układa program na podstawie algorytmu także w innym języku (np. Java).</li> </ul>

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. obliczenia 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 Monte Carlo.	– układa program na podstawie algorytmu także w innym języku (np. Java).