

## Program praktyk

Zawód: Technik Programista

### **SYMBOL CYFROWY ZAWODU 351406**

Celem praktyki zawodowej jest zastosowanie i pogłębienie zdobytej wiedzy i umiejętności zawodowych w rzeczywistych warunkach pracy

Czas trwania praktyki: 280 godzin, w tym:

kl. 3: 3 tygodnie (120 godz.)

kl. 4: 4 tygodnie (160 godz.)

### **KWALIFIKACJE WYODRĘBNIONE W ZAWODZIE:**

INF.03. Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych

INF.04. Projektowanie, programowanie i testowanie aplikacji

### **PRAKTYKA ZAWODOWA klasa 4**

Czas trwania praktyki: 160 godzin 4 tygodnie

#### **Cele ogólne**

- 1) Tworzenie programów z zastosowaniem obiektowości;
- 2) Tworzenie programu z zastosowaniem dziedziczenia;
- 3) Tworzenie aplikacji webowych z zastosowaniem mySQL, html, PHP, JavaScript;
- 4) Wykonywanie zaawansowanych zapytań SQL-owych w bazach danych;
- 5) Tworzenie aplikacji mobilnych;
- 6) Tworzenie desktopowych aplikacji okienkowych;

#### **Cele operacyjne:**

- 1) Tworzenie programów z algorytmami sortowania, wyszukiwania, rekurencją;
- 2) Tworzenie programów z zastosowaniem wskaźników oraz danych dynamicznych;


- 3) Tworzenie złożonych typów danych;
- 4) Tworzenie programu czytającego i zapisującego dane do plików;
- 5) Zapoznanie się z tworzeniem programów skryptowych;
- 6) Tworzenie plików graficznych i multimedialnych;
- 7) Tworzenie zaawansowanych brył w programie 3D;

#### **MATERIAŁ NAUCZANIA W RAMACH PRAKTYKI ZAWODOWEJ w klasie VI**

Dział programowy	Tematy jednostek metodycznych	Wymagania programowe	
		Podstawowe Uczeń potrafi:	Ponadpodstawowe Uczeń potrafi:
<b>I. Organizacja pracy w ramach praktyki zawodowej</b>	1. Przepisy bezpieczeństwa i higieny pracy. Regulamin i zasad oceniania w ramach praktyki zawodowej. Wymagania edukacyjne.	<ul style="list-style-type: none"> <li>- stosować przepisy bezpieczeństwa i higieny pracy oraz ochrony przeciwpożarowej</li> <li>- stosować regulamin praktyki</li> <li>- wymienić zasady oceniania w ramach praktyki zawodowej</li> <li>- stosować zasady organizacji stanowiska teleinformatycznego zgodnie z zasadami ergonomii.</li> <li>- stosować zasady współpracy w zespole.</li> </ul>	<ul style="list-style-type: none"> <li>- określać warunki i organizację pracy zapewniające wymagany poziom ochrony zdrowia i życia przed zagrożeniami występującymi w środowisku pracy</li> <li>- stosować zasady współpracy w zespole</li> </ul>
<b>I. Tworzenie programów strukturalnych.</b>	1. Zastosowanie podstawowych pojęć w programowaniu strukturalnym. Wybrane biblioteki. Zapis i odczyt danych z pliku.	<ul style="list-style-type: none"> <li>- posługiwać się pętlami,</li> <li>- zagnieżdżać pętle,</li> <li>- tworzyć proste typy danych oraz umie z nich korzystać,</li> <li>- korzystać z instrukcji warunkowych,</li> <li>- określić i zastosować instrukcję OR, AND, XOR, negacja,</li> <li>- wykonać operacje na bitach,</li> <li>- korzystać z funkcji wypisujących na ekran oraz wczytujących z klawiatury,</li> <li>- korzystać z tablic,</li> <li>- utworzyć wskaźnik i połączyć ją ze zmienną,</li> <li>- poruszać się wskaźnikiem po tablicy,</li> <li>- uzyskać adres zmiennej,</li> <li>- wskaźnikiem przesyłać dane między funkcjami,</li> <li>- zapoznać się z podstawowymi funkcjami</li> </ul>	<ul style="list-style-type: none"> <li>- podzielić program na funkcje celem lepszej czytelności programu,</li> <li>- narysować schemat blokowy programu,</li> <li>- określić wielkość zmiennych,</li> <li>- określić co to są priorytety operatorów oraz umie podać ich zastosowanie, widzi zastosowanie dla wskaźników,</li> <li>- przesyłać tablice przez wskaźnik do funkcji,</li> <li>- zwracać wskaźnik z funkcji,</li> <li>- przekazywać do funkcji referencję,</li> <li>- podać różnicę między wskaźnikiem a referencją,</li> <li>- posługiwać się dokumentacją do biblioteki,</li> <li>- zapisać strukturę do pliku</li> <li>- zabezpieczyć program przed błędami związanymi z</li> </ul>

		<p>które są w bibliotece matematycznej,</p> <ul style="list-style-type: none"> <li>- zapoznanie się z podstawowymi funkcjami które są w bibliotece String,</li> <li>- dołączyć bibliotekę do projektu,</li> <li>- pierwiastkować i potęgować,</li> <li>- łączyć ciągi napisów,</li> <li>- posługiwać się funkcjami szukającymi w ciągu,</li> <li>- posługiwać się funkcjami zmieniającymi wartości ciągów,</li> <li>- zapisać dane ze zmiennej do pliku tekstowego,</li> <li>- odczytać dane z pliku tekstowego do zmiennej</li> <li>- utworzyć plik,</li> <li>- dopisywać dane do już istniejącego pliku,</li> </ul>	<p>otwieraniem i zapisywaniem do pliku,</p> <ul style="list-style-type: none"> <li>- odczytać dane z pliku tekstowego do tablicy,</li> </ul>
	<p>2. Zmienne dynamiczne. Wykorzystanie struktur danych do budowy własnych typów zmiennych</p>	<ul style="list-style-type: none"> <li>- utworzyć zmienną dynamiczną,</li> <li>- usunąć zmienną dynamiczną,</li> <li>- dać przykład wykorzystania zmiennych dynamicznych,</li> <li>- wykonywać podstawowe operacje na zmiennych dynamicznych (+, -, *, itp.),</li> <li>- utworzyć strukturę danych,</li> <li>- utworzyć dynamiczną zmienną na podstawie struktury danych,</li> <li>- wyodrębniać poszczególne elementy struktury,</li> <li>- użyć struktury do budowy pól bitowych</li> </ul>	<ul style="list-style-type: none"> <li>- przekazywać zmienne dynamiczne do funkcji,</li> <li>- zwracać zmienne dynamiczne z funkcji,</li> <li>- zaplanować aplikację z zastosowaniem struktur danych,</li> <li>- umieścić wskaźnik do zmiennej dynamicznej w definicji struktury,</li> </ul>
<p><b>II. Tworzenie programów z zastosowaniem obiektowości.</b></p>	<p>1. Tworzenie prostych obiektów.</p>	<ul style="list-style-type: none"> <li>- podzielić zagadnienie na klasy,</li> <li>- definiować pola klas,</li> <li>- definiować metody klas,</li> <li>- określać zakres widoczności pól i metod klasy, definiuje kwalifikatory dostępu,</li> <li>- definiować konstruktory, w tym konstruktor kopiujący i destruktor klasy,</li> <li>- implementuje funkcjonalność klasy</li> <li>- deklaruje obiekty i odwołuje się obiektem do składowych klasy</li> </ul>	<ul style="list-style-type: none"> <li>- planować aplikację z zastosowaniem hermetyzacji, dziedziczenia, polimorfizmu,</li> <li>- opisać obiekt na przykładzie z życia wziętym,</li> <li>- tworzyć obiekty statyczne i dynamiczne,</li> <li>- napisać przeciążenie konstruktora,</li> <li>- napisać przeciążoną metodę</li> </ul>

	2. Klasy pochodne.	<ul style="list-style-type: none"> <li>- budować hierarchię dziedziczenia klas w programie,</li> <li>- wydzielać metody i pola do odpowiednich klas w hierarchii dziedziczenia,</li> <li>- schować metody i pola, aby nie były widoczne w innych obiektach dziedziczących,</li> <li>- definiować klasy bazowe i pochodne,</li> <li>- stosować metody wirtualne, definiować klasy abstrakcyjne</li> <li>- stosować szkielet obsługi wyjątków z instrukcjami try i catch,</li> <li>- stosować instrukcję throw,</li> <li>- opracować listę możliwych błędów wykonania aplikacji,</li> </ul>	<ul style="list-style-type: none"> <li>- zaprezentować dziedziczenie na przykładzie z życia wziętym</li> <li>- stworzyć szablon</li> <li>- definiować obsługę dla błędów wykonania aplikacji,</li> <li>- wykorzystać wyjątki przy obsłudze plików,</li> </ul>
<b>III. Programowanie aplikacji okienkowych i mobilnych.</b>	1. Programowanie desktopowych aplikacji okienkowych i aplikacji mobilnych.	<ul style="list-style-type: none"> <li>- zainstalować i skonfigurować środowisko programistyczne,</li> <li>- utworzyć nowy projekt programu,</li> <li>- stosować formatki typowe dla aplikacji desktopowych,</li> <li>- używać elementów interfejsu użytkownika np. okno, kontrolki, dialog modalny i niemodalny,</li> <li>- programować okna aplikacji,</li> <li>- programować system menu aplikacji,</li> <li>- programować okno dialogowe aplikacji,</li> <li>- programować obsługę zdarzeń myszy i klawiatury,</li> <li>- programować aplikacje w jednym z systemów mobilnych,</li> <li>- programować przechowywanie danych i preferencji użytkownika w aplikacjach mobilnych,</li> <li>- używać elementów UI takich jak: przyciski, okienka dialogowe, nawigacja, paski narzędziowe, grafika, dźwięk,</li> <li>- programować aplikacje,</li> <li>- uruchamiać aplikacje mobilne</li> </ul>	<ul style="list-style-type: none"> <li>- dobrać środowisko programistycznego do określonych zadań,</li> <li>- doinstalować odpowiednie dodatki w środowisku programistycznym,</li> <li>- charakteryzować pojęcie framework,</li> <li>- korzystać z zaawansowanych zdarzeń,</li> <li>- skorzystać z zaawansowanych kontrolerek,</li> <li>- stosować frameworki typowe dla aplikacji desktopowych np. WPF,</li> <li>- pobierać i wysyłać dane z lub do Internetu w aplikacjach mobilnych,</li> <li>- programować aplikację mobilną korzystającą z bazy danych,</li> </ul>

	2. Testowanie aplikacji. Łączenie z bazą danych.	<ul style="list-style-type: none"> <li>- zastosować kontrolkę łączącą aplikację z bazą danych,</li> <li>- zapisywać i odczytywać dane z bazy danych</li> <li>- testować aplikację,</li> <li>- testować zdarzenia,</li> <li>- testować kontrolki,</li> </ul>	<ul style="list-style-type: none"> <li>- wykonywać bardziej złożone aplikacje powiązane z bazą danych,</li> <li>- zabezpieczyć połączenie z bazą danych, aby było ono stabilne i odporne na błędy,</li> <li>- wykorzystać widoki z bazy danych,</li> <li>- wykonać dokumentację testowania,</li> <li>- testować połączenie z bazą danych,</li> <li>- przygotować aplikacje mobilne</li> <li>- opublikować aplikację w sklepie</li> </ul>
<b>IV Programowanie aplikacji webowych.</b>	1. Wykorzystanie SQL. 	<ul style="list-style-type: none"> <li>- tworzyć tabele,</li> <li>- stosować polecenia języka SQL,</li> <li>- wyszukiwać informacje w bazie danych przy użyciu języka SQL,</li> <li>- zmieniać rekordy w bazie danych przy użyciu języka SQL,</li> <li>- usuwać rekordy w bazie danych przy użyciu języka SQL,</li> <li>- zakładać użytkowników i nadawać im uprawnienia,</li> <li>- połączyć zapytania SQL wraz z innym językiem np. PHP,</li> </ul>	<ul style="list-style-type: none"> <li>- definiować tabele w bazie danych na podstawie projektu,</li> <li>- tworzyć skrypty w strukturalnym języku zapytań,</li> <li>- kontrolować spójność bazy danych,</li> <li>- tworzyć kopię zapasową bazy danych,</li> <li>- przywracać dane z kopii zapasowej,</li> <li>- importować i eksportować tabele bazy danych,</li> <li>- diagnozować i naprawiać bazę danych,</li> </ul>
	2. Wykorzystanie języków skryptowych. Programy skryptowe po stronie klienta i serwera. Uruchamianie skryptów.	<ul style="list-style-type: none"> <li>- zainstalować środowisko pracy,</li> <li>- skonfigurować środowisko pracy,</li> <li>- poruszać się po systemie,</li> <li>- napisać program w wybranym języku skryptowym</li> <li>- uruchamiać napisany program,</li> <li>- znaleźć błędy w skryptach</li> <li>- implementować poznane algorytmy</li> <li>- stosować w programowaniu obsługę zdarzeń myszy i klawiatury</li> <li>- wykorzystać wybrane biblioteki</li> <li>- stosować składnię oraz podstawowe polecenia, instrukcje i funkcje</li> </ul>	<ul style="list-style-type: none"> <li>- analizować problemy programistyczne,</li> <li>- czytać dokumentację do zaawansowanych funkcji</li> <li>- stosować kaskadowość stylów,</li> <li>- wykonywać responsywne strony internetowe,</li> <li>- stosować zasady dostępności WCAG</li> <li>- posługiwać się narzędziami „Dla twórców witryn” które są wbudowane w przeglądarkę internetową,</li> <li>- podać różnicę między programem kompilowanym a skryptem,</li> <li>- przepisać program z wersji kompilowanej na skrypt</li> <li>- oszacować wynik skryptu,</li> <li>- testować skrypt,</li> </ul>

		wybranych języków skryptowych, - korzystać z funkcji do obsługi ciasteczek, - korzystać z funkcji do obsługi plików, - osadzić skrypt napisany w PHP na stronie internetowej, - odczytać dane z bazy danych, - korzystać ze standardów i dokumentów, - stosować znaczniki języka HTML - definiować strukturę dokumentu hipertekstowego korzystając ze znaczników sekcji - definiować hierarchię treści stosując znaczniki nagłówków i paragrafów, - definiować elementy strony internetowej: listy, tabele, obrazy, odnośniki, kontrolki, - wykonywać formularze na stronie internetowej, - projektować wygląd strony internetowej przy wykorzystaniu języka CSS, - osadzić skrypt napisany w JavaScript na stronie internetowej	- przekazać dane wynikowe z wyjścia jednego programu do wejścia drugiego,
	3. Tworzenie plików graficznych i multimedialnych.	- rozróżniać podstawowe pojęcia dotyczące grafiki, - dobierać oprogramowanie do obróbki grafiki komputerowej, - identyfikować różne formaty plików graficznych, - stosować różne modele barw, - korzystać z edytora grafiki wektorowej, - korzystać z edytora grafiki rastrowej, - wykonywać edycję plików graficznych na potrzeby stron internetowych, - projektować elementy graficzne dla strony internetowej, - osadzać tekst na grafice, dobierać jego krój i styl, - osadzać na stronie internetowej swoje grafiki,	- znaleźć różnicę między grafiką rastrową a wektorową, - korzystać z zaawansowanych funkcji graficznych, - działać na warstwach, - określać zasady komputerowego przetwarzania wideo i dźwięku przygotowanego na potrzeby strony internetowej, - importować materiały multimedialne do systemów CMS, - wykonać animacji na potrzeby strony internetowej, - wykonać materiały wideo na potrzeby strony internetowej,

		<ul style="list-style-type: none"> <li>- dobrać oprogramowanie do edycji obrazu ruchomego i dźwięku,</li> <li>- osadzać elementy multimedialne na stronie internetowej,</li> </ul>	<ul style="list-style-type: none"> <li>- edytować wideo i dźwięk na potrzeby strony internetowej</li> <li>- używać zaawansowanych funkcji do edycji wideo oraz dźwięku</li> </ul>
	4. Modelowanie brył.	<ul style="list-style-type: none"> <li>- dobrać oprogramowania do edycji,</li> <li>- stosować podstawy rysunku 2D i 3D</li> <li>- stosować podstawowe zasady rysowania,</li> <li>- stosować podstawy wymiarowania</li> <li>- stosować podstawowe funkcje,</li> <li>- stosować zasady oświetlania tekstury,</li> <li>- stosować zasady nakładania tekstury,</li> </ul>	<ul style="list-style-type: none"> <li>- zastosować zaawansowane funkcje,</li> <li>- zobrazować przestrzennie bryłę,</li> </ul>
	5. Systemy CMS.	<ul style="list-style-type: none"> <li>- zainstalować CMS,</li> <li>- korzystać z panelu administratora,</li> <li>- administrować systemem zarządzania treścią,</li> <li>- wykorzystywać gotowe szablony,</li> <li>- uzupełniać portal treścią np. dodawać artykuły,</li> </ul>	<ul style="list-style-type: none"> <li>- aktualizować system CMS,</li> <li>- projektować strony internetowe przy wykorzystaniu systemów CMS,</li> </ul>
<b>V Podstawowe algorytmy</b>	1. Algorytmy sortowania, wyszukiwania, rekurencyjne.	<ul style="list-style-type: none"> <li>- podać podstawy teoretyczne działania algorytmów sortowania,</li> <li>- charakteryzować typy sortowania i ich złożoność obliczeniową,</li> <li>- stosować różne typy sortowania, np. bąbelkowe, zachłanne, przez wstawienie, szybkie, metodą dzieli i zwyciężaj,</li> <li>- charakteryzować podstawy teoretyczne działania algorytmów wyszukiwania,</li> <li>- stosować algorytmy wyszukiwania dla tablic, list, kolejek, stosów,</li> <li>- zaimplementować program z rekurencją,</li> <li>- wskazać algorytmy, w których można zaimplementować rekurencję,</li> </ul>	<ul style="list-style-type: none"> <li>- określić złożoność obliczeniową algorytmów,</li> <li>- przepisać program i zaimplementować w nim rekurencję</li> </ul>

<b>Zakończenie praktyki</b>	Podsumowanie praktyki. Ocena efektów kształcenia i zaliczenie praktyki zawodowej.		
---------------------------------	--	--	--

Osiągnięcia ucznia oceniać na bieżąco będą opiekunowie praktyk. Na zakończenie praktyki uczeń musi przedłożyć opiekunowi dziennik praktyki.