

Opracował: Włodzimierz Kasprzak

Sylabus przedmiotu „JiPP, sem. 4”

1. Nazwa jednostki prowadzącej kierunek

Warszawska Wyższa Szkoła Informatyki

2. Nazwa kierunku (podanie nazw ew. specjalności)

Informatyka

3. Nazwa przedmiotu

Języki i paradygmaty programowania – *drugi semestr przedmiotu (semestr 4 studiów)*

4. Określenie przedmiotów wprowadzających wraz z wymaganiami wstępnymi

Przedmioty poprzedzające:

- „Wstęp do programowania”,
- „Podstawy programowania”,
- „Języki i paradygmaty programowania, sem. 3”.

Wymagania wstępne:

- ogólna wiedza o arytmetyce komputerów, strukturach danych i zasadach programowania komputerów,
- podstawowe umiejętności programowania strukturalnego w języku C,
- podstawowe umiejętności programowania obiektowego w języku C#.

5. Liczba godzin zajęć dydaktycznych, w tym wykładów, ćwiczeń, laboratoriów, seminariów itp., realizowanych w systemie studiów stacjonarnych i niestacjonarnych

Wykłady: 28 (dzienne), 16 (zaoczne)

Laboratoria: 28 (dzienne), 16 (zaoczne)

6. Liczba punktów ECTS

6 pkt. ECTS

7. Założenia i cele przedmiotu

Przedmiot wprowadza zasady i pojęcia stylów programowania: zdarzeniowego, wizualnego, wielowątkowego i rozproszonego oraz pokazuje ich implementację w języku programowania C# i w środowisku programisty MS.NET. Omawiane są takie zagadnienia, jak: wskaźniki do funkcji w C++ a delegaty metod w C#, wzorce projektowe w inżynierii oprogramowania **Observer** i **Composite**, mechanizm rozgłaszania i obsługi zdarzeń, podstawowe zdarzenia dla klawiatury i myszy, aplikacje okienkowe **Forms**, klasa **Control**, podstawowe zdarzenia dla kontrolki, kontrolki w Visual Studio, okna dialogowe, obszar **Drawing** i grafika 2-D, model asynchronicznego wywołania metod, wątek obsługi zdarzeń formatki, obszar **Threading** - tworzenie i zarządzanie wątkami w

programie, współpraca i synchronizacja pracy wątków, komunikacja procesów i obiektów w warstwie aplikacji modelu sieciowego – gniazdka, RPC, COM, współpraca obiektów .NET z COM – obszar InteropServices, klasa obiektów zdalnych MarshalByRefObject, partycje procesu AppDomain, mechanizm komunikacji ze zdalnym obiektem w modelu klient-serwer na podstawie obszaru Remoting.

Celem przedmiotu jest przekazanie studentom wiedzy o zasadach tworzenia wielowątkowych aplikacji, sterowanych zdarzeniami, wykorzystujących graficzny interfejs użytkownika i aplikacji rozproszonych korzystających z modelu komunikacji klient-serwer oraz nabycie przez studentów praktycznej umiejętności tworzenia takich programów w środowisku programisty, na przykładzie języka C# i środowiska programisty MS.NET.

8. Metody dydaktyczne

Przedmiot obejmuje wykład i laboratorium. Wykład realizowany jest z użyciem komputera i sprzętu audio-wizualnego. Materiały wykładowe są w postaci elektronicznej. Laboratorium wyposażone jest w sieć lokalną komputerów klasy PC z zainstalowanym środowiskiem programisty MS.NET i edytorem Visual Studio.NET w najnowszej wersji (obecnie MS.NET 3.5). Każdy student ma do dyspozycji własną stację roboczą. Obecności i aktywność podczas ćwiczeń są oceniane. Podczas zajęć student realizuje samodzielnie dwa oceniane projekty programowe.

9. Forma i warunki zaliczenia przedmiotu

Forma: zaliczenie. Wymagane jest niezależnie pozytywne zaliczenie obu części przedmiotu: wykładu i laboratorium.

Warunki zaliczenia:

- Zaliczenie laboratorium odbywa się na podstawie 2 projektów programowych.
- Zaliczenie części wykładowej odbywa się w sesji egzaminacyjnej na podstawie sprawdzianu pisemnego i ewentualnej dodatkowej odpowiedzi ustnej.

10. Treści programowe

| Temat | Treść |
|-------------------------------------|--|
| 1. Wskaźniki do funkcji w C++ | Typ funkcji. Definiowanie wskaźnika do funkcji. Wskaźnik do składowej w klasie. |
| 2. Delegaty i zdarzenia | Klasa Delegate i słowo kluczowe delegate . Wzorzec projektowy Observer . Definiowanie typu zdarzenia i samej reprezentacji zdarzenia - słowo kluczowe event . Rejestrowanie metod obsługi zdarzenia. Typy EventHandler i EventArgs . Wywołanie i rozgłaszanie zdarzenia. Generyczne delegaty. |
| 3. Aplikacje okienkowe w MS-Windows | Wzorzec projektowy Composite . Obszar nazw System.Windows.Forms . Klasa Control – własności Parent , Controls i klasa ControlCollection . |

| | |
|---|--|
| | <p>Klasa Form – metoda Main, atrybut STAThread, ważniejsze własności, własności Owner i OwnedForms. Klasa Application i jej główne metody. Cykl życia formatki – metody i zdarzenia formatki (Show, ShowDialog, Hide, Close, Load, Activated, GotFocus, itd.) Formatki typu SDI i MDI.</p> |
| 4. Podstawowe zdarzenia w .NET | <p>Konwencja nazw zdarzeń, typów i metod obsługi w klasach bibliotecznych .NET. Zdarzenia Click i DoubleClick dla kontroltek. Zdarzenia generowane dla akcji myszy i klawiatury. Zdarzenia generowane dla akcji myszy Mousexxx i klawiatury Keyxxx. Operacja Drag-and-Drop.</p> |
| 5. Kontrolki | <p>Przegląd standardowych kontroltek w VisualStudio. Okna dialogowe dla wydruku: PrintPreviewDialog, PageSetupDialog, PrintDialog. Edytor graficzny – dodawanie kontroltek do aplikacji. Przykłady kodów inicjalizacji GUI aplikacji.</p> |
| 6. Obszar System.Drawing – grafika | <p>Biblioteka graficzna GDI+. Klasa Graphics i jej metody kreślenia grafiki. Klasy Pen, Brush i Font. Współrzędne ekranu i klienta. Metody klasy Control: PointToClient i PointToScreen. Podobszar Printing - klasa PrintDocument. Zdarzenie PrintPage i formatowanie strony wydruku. Wypełnianie wielokątów – rodzaje pędzla w Drawing2D. Przetwarzanie obrazów - podobszar Imaging</p> |
| 7. Asynchroniczne operacje | <p>Wątki a asynchroniczne wykonanie metod. Asynchroniczne metody klas bibliotecznych – BeginOperation, EndOperation. Typ delegata AsyncCallback. Interfejs IAsyncResult. Metody Invoke, BeginInvoke i EndInvoke w klasie Control. Asynchroniczne delegaty – metody BeginInvoke, EndInvoke.</p> |
| 8. Wielowątkowy program | <p>Model pracy wielowątkowej w Win32. Klasa Thread i typy delegatowe ThreadStart, ParameterizedThreadStart. Własność IsBackground. Stany wątku – typ ThreadState. Synchronizacja pracy wątków - sekcja krytyczna lock, metody Enter i Exit klasy Monitor. Metody Puls, PulsAll i Wait klasy Monitor. Podstawowe klasy dla synchronizacji pracy wątków: WaitHandle, EventWaitHandle, Mutex, Semaphore.</p> |
| 9. Współpraca z obiektami COM | <p>Podstawowe elementy systemu komunikacji w warstwie aplikacji – gniazdka, RPC. Model COM tworzenia aplikacji rozproszonych. Klient w .NET a serwer w COM. Odwzorowanie definicji co-klasy serwera i interfejsów z języku IDL na definicje w C#.</p> |

| | |
|---|---|
| 10. Zdalne obiekty - obszar System.Runtime. Remoting | Klasa MarshalByRefObject . Partycje procesu – klasa AppDomain . Podstawowe klasy w Remoting: RemotingConfiguration, RemotingServices, ObjRef . Abstrakcje kanałów transportowych – Channels . Przykład aplikacji klient-serwer z wykorzystaniem mechanizmu zdalnych obiektów. |
|---|---|

11. Wykaz literatury podstawowej i uzupełniającej

Literatura podstawowa:

1. W. Kasprzak: Materiały elektroniczne do wykładu „*Języki i paradygmaty programowania, sem. 4*”, WWSI, Warszawa, 2008 - 2009.

Literatura uzupełniająca:

1. J. Templemann, D. Vitter: Visual Studio .NET Framework. Czarna księga. Wyd. RM, Warszawa, 2003.
2. ---: MSDN Library. .NET Development.
<http://msdn.microsoft.com/en-us/library/aa139615.aspx>
3. ---: MSDN Library. Development Tools and Languages. Visual C#.
<http://msdn.microsoft.com/en-us/library/kx37x362.aspx>
4. Materiały kursu treningowego firmy *Microsoft* o numerze 2555AC (*Developing Microsoft .NET Applications for Windows – Visual C#*).