# ITKPU: Læringsmål

* Redegøre for principperne for komponentbaseret programudvikling

designe, implementere og anvende DDL-filer og plugins i C++.

* Redegøre for COM's arkitektur og terminologi
* Redegøre for .NETs komponentmodel samt kunne designe, implementere og anvende .Net komponenter ved brug af C#.
* Redegøre for anvendelsen af framework til dependency injection.
* Redegøre for begrebet "Interoperability" og kunne designe for og anvende interoperability mellem unmanaged og managed kode.
* Redegøre for Cross Platform Development på .Net platformen, og kunne designe for og anvende Cross Platform Development.
* Designe og implementere Windows RT Compoments
* Designe og implementere Windowsservices ved brug af .Net og C#.

ITKPU Eksamensdispositioner:

## Spørgsmål 1: Komponentbaseret programudvikling, designprincipper

Redegør for ideerne bag komponentbaseret programudvikling:

Redegør for tilhørende designprincipper:

## Spørgsmål 2: Komponentbaseret programudvikling, DLL i C++, vis eksempel plugins C++

Redegør for ideerne bag komponentbaseret programudvikling:

Redegør for udvikling og brug af DLL-filer i C++:

Vis og forklar hvorledes man kan udvikle og anvende plugins i C++:

## Spørgsmål 3: .Net komponentmodel, Lifecycle Management, Components C#, Eksempel

Redegør for .Nets komponentmodel og Lifecycle Management:

Vis og forklar hvorledes man kan udvikle og anvende komponenter i C#:

## Spørgsmål 4: Dependency Injection, IoC-containere, typer DI +/ Extensibility Frameworks, MEF. Eksempel MEF/Unity

Redegør for begrebet dependency injection:

Redegør for brugen af IoC containere:

Giv et overblik over Microsofts forskellige dependency injectons og/eller Extensibility Frameworks:

Redegør for de grundlæggende begreber i MEF:

Vis et eksempel på brug af MEF eller Unity:

## Spørgsmål 5: Prism, Pros & Cons, Prism Eksempel

Redegør for den grundlæggende arkitektur og begreber i Prism:

Hvilke fordele og ulemper er der ved at bruge Prism:

Vis et eksempel på brug af Prism:

## Spørgsmål 6: Interoperability, PInvoke, Interoperability COM og .Net, interoperability eksempel

Redegør for begrebet ”Interoperability” generelt:

Redegør for brugen af PInvoke samt interoperability mellem COM og .Net

Vis eksempler på interoperability:

## Spørgsmål 7: Cross Platform .Net, Cross platform component eksempel, Windows RT components.

Redegør for problemer og muligheder for cross-platform developement på .Net platformen:

Vis et eksempel på en cross-platform component:

Redegør for hvorledes man designer og implementerer Windows RT components:

## Spørgsmål 8: Cross platform Android + iOS, Xamarin + platformer, Pros & Cons Xamarin, Xamarin eksempel.

Giv et overblik over de forskellige muligheder for cross-platform udvikling for Android og iOS enheder:

Redegør for Xamarins arkitektur og virkemåde for de forskellige understøttede platforme:

Redegør for både fordele og udfordringer ved brug af Xamarin til App udvikling:

Vis et eksempel på brug af Xamarin:

## Spørgsmål 9: Services, Windowsservice .Net C#, Windowsservice eksempel

Redegør for begrebet ”Services”:

Redegør for hvorledes man designer og implementerer en WindowsService ved brug af .Net og C#:

Vis et eksempel på en WindowsService:

## Spørgsmål 10: Components in Clientside web, UI-components JS, Web Component eksempel (u. framework, evt. LitElement)

Redegør for brugen af komponenter ved client side webudvikling:

Hvad skal browseren understøtte for at man kan lave UI-komponenter, som er uafhængige af et JS framework:

Vis et eksempel på en web komponent, som er udviklet uden brug af framework eller evt. ved brug af LitElement: