

Laboration 2

MINA FÖRSTA OBJEKT

OBJEKTBASERAD PROGRAMMERING I C++ Ver 11

Mål: Lära sig grunderna i objektorienterad programmering.
Lära sig använda begreppen klass, objekt, inkapsling, klass i en annan klass.
Skriva ett klientprogram som använder en klass.

Redovisning: Koden skall vara lättläst där indragningar, kommentarer och variabelnamn är logiska och konsekventa.

Skriv en laborationsbeskrivning enligt anvisning i Moodle

Skicka in källkod och laborationsbeskrivning via Moodle.
Packa alla filer i en zip-fil (eller motsvarande).

Regler för inlämning:

Genom att du lämnar in detta arbete försäkrar du att alla svar är skapade av dig själv. Du är även ansvarig att se till att det inte finns någon plagierad text i dokumentet. När du refererar och citerar andra verk måste korrekta källhänvisningar finnas och i fallet citering ska den citerade texten vara tydligt markerad. <http://www.bib.miun.se/student/skriva/referenser>

Om plagierad text finns i dokumentet riskerar du att stängas av från studier.

Om samarbete sker utan att detta har stöd i instruktionen för examinationen utgör det normalt en disciplinförseelse och du som student riskerar att stängas av från dina studier.

LABORATION 2 Mina första objekt

Inledning

I denna laboration ska du skapa tre klasser. En klass för en persons namn, en klass för en persons adress och en klass för en persons namn, adress och personnummer. För varje klass ska du skriva ett testprogram (klientprogram).

Uppgift 1 En namnklass

Skapa en klass som hanterar en persons förnamn och efternamn.

Klassspecifikation:

Namn:	Name
Datamedlemmar:	firstName (datatyp: string) lastName (datatyp: string)
Medlemsfunktioner:	Förvald konstruktor. Konstruktor för initiering av datamedlemmarna. setFirstName setLastName getFirstName getLastName

- Du får själv ange medlemsfunktionernas parameterlistor och returdatatyp.
- const-deklarera de medlemsfunktioner som inte ändrar data.
- Placera klassdefinitionen i en headerfil och implementationerna av medlemsfunktionerna i en separat cpp-fil.
- Skriv ett testprogram som placeras i en egen fil. **Testprogrammet ska testa alla de operationer klassen kan utföra, d.v.s. alla konstruktorer, setfunktioner, get-funktioner och eventuella andra medlemsfunktioner.**

Uppgift 2 En adressklass

Skapa en klass som hanterar en persons adress.

Klassspecifikation:

Namn:	Address
-------	---------

I klassen ska det ingå datamedlemmar som hanterar gatuadress, postnummer och stad.

Skapa medlemsfunktioner med namnklassen som mall.

- Placera klassen i en header- och en implementationsfil.
- Skriv ett testprogram för adressklassen. **Testprogrammet ska testa alla de operationer klassen kan utföra, d.v.s. alla konstruktorer, setfunktioner, get-funktioner och eventuella andra medlemsfunktioner.**
- Låt alla datamedlemmar vara av datatypen string.

Uppgift 3 En personklass

Skapa en klass som hanterar namn, adress, personnummer och skonummer för en person.

Klassspecifikation:

Namn:	Person
Datamedlemmar:	Name name Address address string persNr int skoNr
Medlemsfunktioner:	Skapa medlemsfunktioner (inkl konstruktorer) med vilka man kan sätta värden på och läsa värden från datamedlemmarna.

Placera klassen i en header- och en implementationsfil.

Skriv ett program som använder personklassen enligt kravspecifikationen nedan.

Kravspecifikation:

- Skapa en vector i vilken man kan lägga in personer av klassen (datatypen) Person
- Gör en meny med alternativen
 - Mata in data för en person
 - Visa data för alla personer
 - Sluta
- Skriv en funktion som läser in data för en person och lägger in posten i vektorn.
- Skriv en funktion som skriver ut data för alla personer som finns i vektorn.
- Inmatning från tangentbord och utskrift till skärmen får inte göras från main().
- Globala variabler är inte tillåtna.
- Initiera data för 3 personer i vektorn innan menyn startas.