



UNIVERSITETET I AGDER

FAKULTET FOR ØKONOMI OG SAMFUNNSVITENSKAP

## E K S A M E N

**Emnekode:** IS-207  
**Emnenavn:** Algoritmer og datastrukturer

**Dato:** 20. mai 2014  
**Varighet:** 0900-1300

**Antall sider inkl. forside:** 2

**Målform:** Norsk

**Tillatte hjelpemidler:** Alle trykte og skrevne

---



## Innledning

Dette avsnittet inneholder bakgrunnsinformasjon som skal gjøre det lettere å forstå oppgavene under. Følgende begreper blir brukt i oppgaveteksten:

*C*: Programmeringsspråk

*Funksjon*: C har ikke klasser, bare funksjoner som er det samme som metoder i java.

*C-fil*: Kildekodefil som kan inneholde en eller flere funksjoner

*Objektfil*: Resultatet av å kompilere en C-fil. Objektfilen inneholder binærkoden for de funksjonene som er definert i C-filen.

*Linker*: Program som setter sammen objektfilen(e) til et program, med objektfiler som hentes fra et eller flere biblioteker. Resultatet av å kjøre linkerer er et ferdig, kjørbart program.

*Linking*: Prosessen som linkerer utfører: Finne objektfiler som inneholder funksjoner som er brukt i programmet, men ikke definert der. Disse objektfilene hentes fra et eller flere biblioteker, og lagres i den kjørbare filen sammen med objektfilene til programmet.

*Uløste referanser*: En funksjon som er brukt i programmet, men som linkerer ikke har funnet i bibliotekene (ennå).

*Bibliotek*: En samling av objektfiler (omtrent som en zip-fil). Biblioteket har en index som sier hvilke funksjoner som er definert i biblioteket, og hvilken objektfil de finnes i.

## Oppgave 1 - Oppvarming

Vi gjør noen forenkende antagelser i denne oppgaven: Selve programmet består av en fil. Funksjonene i programmet kan kalle en eller flere biblioteksfunksjoner. Videre antar vi at biblioteksfunksjonene er definert i hver sin fil, og at de ikke bruker andre biblioteksfunksjoner. Du skal gjøre ”ferdig” klassen *Linker* (vedlegg 1)

- Gjør rede for hvilke datastrukturer du mener du trenger, og definer disse som felt i klassen eller lokale variable i *link()*. Begrunn valgene.
- Skriv ferdig metoden *findLibraryObjects()* i *Linker*.

## Oppgave 2 – Realistisk linking

Nå dropper vi antagelsene fra oppgave 1. Det får noen konsekvenser: Biblioteksfunksjonene kan kalle andre biblioteksfunksjoner.

Det betyr at når linkerer inkluderer en biblioteksfunksjon, vil vi kanskje få nye uløste referanser som vi må finne i et eller annet bibliotek.

Videre tillater vi at en objektfil kan inneholde flere funksjoner. Dermed kan linkerer komme til å løse flere referanser samtidig. Det kan også skje at linkerer finner en referanse til en ubrukt funksjon i en av biblioteksfilene den allerede har inkludert. Det må den fikse uten å inkludere filen en gang til (det ville gi dobbeltdefinisjoner av funksjonene i filen).

Oppgavene tar for seg *linking*, prosessen som knytter et program sammen med de kodebibliotekene det trenger. I java skjer dette automatisk, men det er ikke slik i alle språk. I denne oppgaven skal vi se på statisk linking av C-programmer. Statisk linking betyr at koden fra bibliotekene blir fysisk kopiert inn i programmet.

Programmeringsspråket C har ikke klasser, bare *funksjoner* (som er det metoder blir kalt i C verdenen). Kildekoden til et C-program består av en eller flere filer, som hver kan inneholde en eller flere funksjoner. Hver kildefil blir kompilert til en objektfil som inneholder den kjørbare binærkoden til funksjonene.