**Oppgave 3**

**1. Finn ut hva objekt filer heter for de mest brukte platformene (Unix/Linux, MS Windows, Mac OS X)! Hvorfor, etter din mening, har disse plattformene så forskjellige objektfil-formater?**

Objektfiler deltar i programkobling (bygge et program) og programgjøre (kjører et program) For enkelhets skyld og effektivitet gir objektfilformatet parallelle visninger av innholdet i en fil, noe som gjenspeiler de ulike behovene til disse aktivitetene.

**Unix / Linux: ELF** (Executable og Linking Format) er et vanlig standard filformat for kjørbare filer, objektkode, delte biblioteker og kjernefelt.

**MS Windows: PE** (Portable Executable) format er et filformat for kjørbare, objektkode, DLLer, FON Font-filer og andre. Brukes i 32-biters og 64-biters versjoner av Windows-operativsystemer.

**Mac OS X: Mach-O** (Mach-objektfilformat) er et filformat for kjørbare filer, objektkoden delte biblioteker, dynamisk lastet kode og core dump. Mach-O brukes av de fleste systemer basert på Mach kernel. NeXTSTEP, macOS og iOS er eksempler på systemer som har brukt dette formatet for grunnlegande kjøring, biblioteker og objektkoden.

Jeg vet ikke for mye om objektfiler eller operativsystemer, men jeg tror at disse operativsystemene har forskjellige objektfiler fordi de ganske enkelt er skrevet forskjellig og har forskjellige behov.

**2. Hvilke forskjeller ser du i forhold til programmeringsspråket Java?**

**GO:**   
- Hvert program består av pakker

- Starter i pakken "main"

- Kortere kodelinjer

- Et navn blir eksportert hvis det begynner med en stor bokstav

- Golang er kanskje ikke objektorientert

- Go er kompilert til maskinkode og kjøres direkte.

- Har fjernet noen ting

- Kanskje mer effektiv

- Lettere å lese