Kunnskaper – kandidaten skal

•  vite hva arv og polymorfi er

* Arv lar definisjonen av klasse med å forlenge av en annen klasse.
* Polymofi er at man kan holde objekter av en deklarert type, eller av subtyper av en deklarert type.

•  vite om likheter/forskjeller mellom abstrakte klasser og interfaces

•  vite hva unntakshåndtering innebærer

* Det handler om defensiv programmering.

•  vite hva et design pattern er, og kjenne til noen eksempler på slike

Design patterns help clarify relationships, and promote reuse.

Ferdigheter – kandidaten skal

•  definere spesialiseringer av klasser gjennom arv

Subklasse og superklasse

•  beherske bruk av abstrakte klasser og interface (’kontrakter’)

abstrakt klasse: En klasse som det **ikke** er tiltenkt å lage instanser av. Den skal fungerer som superklasse for andre klasser. Den inneholder gjerne abstrakte metoder.

Interface - grensesnitt (sier gjerne interface på norsk): Spesifikasjon av en type. Har som regel ikke noe implementasjon.

•  anvende arv og interfaces gjennom kode med polymorfisk oppbygning

•  programmere enkle grafiske grensesnitt

* JavaFX tegning

Generell kompetanse – kandidaten skal

•  forklare begrepet «teknisk gjeld» sett i lys av design og implementasjon av kode

Technical debt is a concept in programming that reflects the extra development work that arises when code that is easy to implement in the short run is used instead of applying the best overall solution.

•  kunne forklare behovet for analyse og design ifm utvikling av kode