

Learning to Program with F#  
Exercises  
Department of Computer Science  
University of Copenhagen

Jon Sparring, Martin Elsmann, Torben Mogensen, Christina Lioma

September 28, 2021

## 0.1 Simple Jack

### 0.1.1 Opgave(r)

**0.1.1:** Design og implementér et program som kan simulere Simple Jack ved brug af klasser. Start med grundigt at overveje hvilke aspekter af spillet som giver mening at opdele i klasser. Spillet skal implementere således, at en spiller enten kan være en bruger af Simple Jack programmet, som foretager sine valg og ser kortene på bordet via terminalen, eller en spiller kan være en AI som skal følge en af følgende strategier:

- (a) Vælg altid "Hit", medmindre summen af egne kort kan være 17 eller over, ellers vælg "Stand"
- (b) Vælg tilfældigt mellem "Hit" og "Stand". Hvis "Hit" vælges trækkes et kort og der vælges igen tilfældigt mellem "Hit" og "Stand" osv.

Dealer skal følge strategi nummer 1. Der skal også laves:

- En rapport (maks 2 sider)
- Unit-tests
- Implementation skal kommenteres jævnfør kommentarstandarden for F#

Hint: Man kan generere tilfældige tal indenfor et interval (f.eks. fra og med 1 til og med 100) ved brug af følgende kode:

```
let gen = System.Random()  
let ran_int = gen.Next(1, 101)
```