

4 Bingo

Uppgiften handlar om ett lotteri eller *bingo* vore kanske ett bättre ord.

I denna uppgift kommer ni få arbeta med vektorer och slumpal.

Vidare kommer ni få visa att ni kan *loopa* (iterera) genom en vektor på ett lämpligt sätt, och för vektorer används då ofta *for*- eller *foreach*.

Grund-funktionalitet (Betyg E):

I uppgiften ska besökaren få skriva in 10 stycken tal mellan exempelvis ett(1) och tjugofem(25) som ska lagras i en heltals-vektor.

När användaren skrivit in 10 stycken tal börjar spelet.

Nu ska ett tal slumpas fram.

Koden ska nu gå genom hela vektorn (med en loop) och testa tal för tal mot eventuella matchningar. Får man en matchning har man vunnit, d.v.s. att talet i vektorn är *lika med* talet som slumpats fram.

För denna nivå behöver bara *en* vektor användas – själva "bingo-brickan" där användaren skrivit in tal.



Ytterligare funktionalitet (Betyg C)

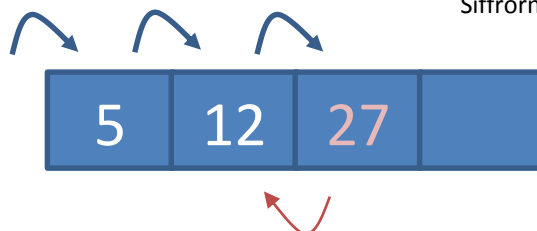
För betyget C ska flera tal slumpas av, exempelvis sju (7).

Detta kan man då hantera via en loop. För varje tal som slumpas fram ska man testa om man får "bingo". Här ska man då alltså kunna få mer än ett (1) rätt och när programmet är över ska antal rätt skrivas ut.

Ni ska också se till att talen som användaren skriver in ska vara mellan 1 och 25 vilket innebär att om användaren skriver in 27 exempelvis så ska inte det talet lagras.

Detta löser ni genom att på något sätt kontrollera loopen och hur den går framåt i vektorn.

Loopar genom vektorn
med exempelvis en *for*-
loop.



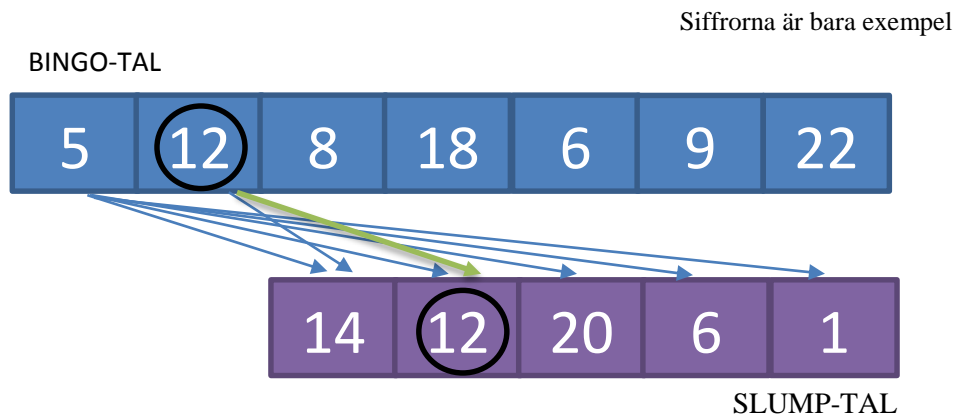
Siffrorna är bara exempel

Alternativet är att ni sparar det inmatade talet i en temporär variabel och lagrar det i vektorn om det uppnår kraven (det vill säga mellan ett och tjugofem).

Ni ska också använda undantagshantering vid inmatning och det ställs högre krav på kodstruktur och hur ni hanterar in- och utmatningar med formatering.

Massor av funktionalitet (Betyg A)

Slumptalen som tas fram ska nu lagras i en vektor *innan* dessa slumpal testat mot de tal som användaren skrivit in.



Alla tal ska alltså testat mot alla som ni kan se i figuren. I ovanstående exempelvis så kommer det slutligen bli två rätt (12 och 6).

Utöver detta gäller kraven för betyget C och överlag att det ställs högre krav på säker kodning, användarinteraktion etc.

Extra svårt

Vill ni glänsa ordentligt så kan ni se till att alla siffror är *unika* – och detta med egen kod. Detta kan gälla då både för vektorn med slumpal och vektorn med inmatade tal ("bingo-brickan")

Koden blir rätt omfattande här, det går inte att lösa på något jätte-enkelt sätt.