

SI-pass 4 OOP

Handledare: Oscar Söderlund

Mail: soscar@student.chalmers.se

1. Skriv ett program som demonstrerar att, oavsett hur många objekt man skapar av en viss klass, så finns det bara en enda instans av en given statisk instansvariabel i den klassen.
2. Gamle De Morgans lagar känner vi alla till:

$$\neg(P \vee Q) = \neg P \wedge \neg Q$$

$$\neg(P \wedge Q) = \neg P \vee \neg Q$$

Generellt sett vill man skriva sina if-satser utan massa jobbiga negationer (!-tecken), använd därför De-Morgans lagar för att förenkla följande if-satser:

(a) `if (!(x < 5) && !(y >= 7)) { }`

(b) `if (!(a == b) || !(g != 5)) { }`

(c) `if (!((x <= 8) && (y > 4))) { }`

(d) `if (!((i > 4) || (j <= 6) || !(a == b))) { }`

Målet är att så få !-operatorer som möjligt skall finnas kvar.

!= operatorn är naturligtvis en helt annan femma, den gillar vi.

3. Ett vampyrtal har ett jämnt antal siffror, och kan bildas av att dela in talet i två sifferpar och multiplicera paren med varandra.

Ex:

$$1260 = 21 * 60$$

$$1827 = 21 * 87$$

$$2187 = 27 * 81$$

Koda nu ett program som hittar och skriver ut alla fyrsiffriga vampyrtal! :F