

1. Pohdi Occamin partaveitsen ilmentymiä arkielämässä. Periaatteen mukaan kaikkien muiden seikkojen ollessa yhteneviä, tulee suosia yksinkertaisinta mallia / selitystä. Sama maksimi voitaneen laajentaa myös muille elämänaloille. Tapahtuuko näin? Millaisia (vasta)esimerkkejä löydät?
2. Päätöspuussa ei koskaan testata samaa diskreettiä muuttujaa kahdesti yhdellä juuri-lehti-polulla. Miksi?
3. Mikä on attribuuttien *Asiakkaita* ja *Tyyppi* informaation lisäys oheisella opetusaineistolla?

Esim.	Attribuutit										Odot
	VE	B	VL	N	Asiakkaita	H	S	V	Tyyppi	Arv.	
x_1	On	Ei	Ei	On	Joitain	\$\$\$	Ei	On	Ranskal.	0–10	Kyllä
x_2	On	Ei	Ei	On	Täysi	\$	Ei	Ei	Thai	30–60	Ei
x_3	Ei	On	Ei	Ei	Joitain	\$	Ei	Ei	Hampur.	0–10	Kyllä
x_4	On	Ei	On	On	Täysi	\$	On	Ei	Thai	10–30	Kyllä
x_5	On	Ei	On	Ei	Täysi	\$\$\$	Ei	On	Ranskal.	>60	Ei
x_6	Ei	On	Ei	On	Joitain	\$\$	On	On	Ital.	0–10	Kyllä
x_7	Ei	On	Ei	Ei	Tyhjä	\$	On	Ei	Hampur.	0–10	Ei
x_8	Ei	Ei	Ei	On	Joitain	\$\$	On	On	Thai	0–10	Kyllä
x_9	Ei	On	On	Ei	Täysi	\$	On	Ei	Hampur.	>60	Ei
x_{10}	On	On	On	On	Täysi	\$\$\$	Ei	On	Ital.	10–30	Ei
x_{11}	Ei	Ei	Ei	Ei	Tyhjä	\$	Ei	Ei	Thai	0–10	Ei
x_{12}	On	On	On	On	Täysi	\$	Ei	Ei	Hampur.	30–60	Kyllä

4. Yksinkertainen jälkikarsinta päätöspuille on *reduced error pruning* (Rep), joka opetusjoukon lisäksi vaatii erillisen karsintajoukon esimerkkejä. Annettuna opittu päätöspuu T Rep-algoritmi ensin luokittelee kaikki karsintajoukon tapaukset T :llä pitäen matkalla yllä tarvittavia laskureita ja sen jälkeen alhaalta ylöspäin käy puun läpi karsien siitä alipuut, joiden poistaminen ei kasvata virhettä karsintajoukon luokittelussa. Kirjoita Rep-algoritmi.