

Lektion	Kapitel	Uppgift	Lösning med programmering
4 – For	1 Talteori, Geometrisk talföljd och summor	1226 Beräkna den geometriska summan $2\,000 + 2\,000 \cdot 1,025 + 2\,000 \cdot 1,025^2 + \dots + 2\,000 \cdot 1,025^7$	Lös uppgiften med ett program, t.ex. <pre>summa = 0 for n in range (0, 8): summa = summa + 2000 * 1.025**n print(summa)</pre>
6 – If, and och or	1 Talteori, Delbarhetsregler och division med rest	1117 Vad blir kvoten och resten när följande tal delas med 4? a) 6 b) 19 c) 3 d) 40	Vill man låta eleverna träna på att använda restoperatören i Python kan man låta eleverna lösa uppgiften med programmering. <pre>print("Kvoten är", 6//4, "och resten är", 6 % 4)</pre> Operatören // returnerar heltalsdelen av kvoten.
	1 Talteori, Delbarhetsregler och division med rest	1123 Bestäm alla primtal i intervallet mellan 110 och 120.	Utvidga uppgift 1123 till att låta eleverna skriva ett program som avgör om ett inmatat tal större än 2 är ett primtal. <pre>n = int(input("Det här programmet kollar om talet n är ett primtal. Ange n.")) for x in range (2,n): if n % x == 0: print(n, "är inget primtal.") break else: print(n, "är ett primtal")</pre> Diskutera hur man kan ändra i programmet så att det blir mer effektivt, till exempel utifrån resonemangen i uppgift 1126.

Lektion	Kapitel	Uppgift	Lösning med programmering
9 – Listor	1 Talteori, Diskret matematik	1109, 1110 För ett tal a gäller att talet kallas fattigt om summan av alla delare, förutom talet självt, är mindre än talet. Vilka tal i intervallet är $25 \leq a \leq 30$ är a) fattiga tal b) rika tal c) perfekta tal	Låt eleverna lösa problemet med ett program: <pre> perfekta_tal = [] rika_tal = [] fattiga_tal = [] for x in range(25, 31): delare = [] for a in range(1, x): if x % a == 0: delare.append(a) b = sum(delare) if b == x: perfekta_tal.append(x) elif b > x: rika_tal.append(x) else: fattiga_tal.append(x) print("Antalet perfekta tal mellan 25 och 30 är:", len(perfekta_tal)) print("Antalet rika tal mellan 25 och 30 är:", len(rika_tal)) print("Antalet fattiga tal mellan 25 och 30 är:", len(fattiga_tal)) </pre>
	1 Talteori, Aritmetiska talföljder och summor	1220 Figuren visar ett mönster som bildas av prickar. c) Hur många prickar innehåller figur 1 till och med 100 sammanlagt?	Lös uppgiften med ett program och jämför med en algebraisk lösning: <pre> lista = [] for n in range(1, 101): lista.append(3 + 4 * (n - 1)) print(sum(lista)) </pre>

Lektion	Kapitel	Uppgift	Lösning med programmering
11 – Turtle	2 Mängder, kombinatorik och grafer, Permutationer	Permutationer (s. 64 i Origo 5)	I teoritexten på sidan 64 i elevboken definieras begreppet fakultet. Låt eleverna definiera en funktion som beräknar $n!$ för ett givet n. <pre>def fakultet(n): a = 1 for x in range (1, n + 1): a = a * x return(a) n = int(input("Det här programmet beräknar n!. Ange n:")) print(fakultet(n))</pre> Låt dem sedan använda funktionen för att skriva program som beräknar $P(n, k)$ och $C(n, k)$ när man matar in n och k.

Lektion	Kapitel	Uppgift	Lösning med programmering
	3 Differential-ekvationer, Eulers stegmetod	3124 Beräkna ett närmevärde till $y' = y/x$ som uppfyller villkoret $y(1) = 2$. Använd Eulers stegmetod och steglängden 1.	Lös med ett program, t.ex. <pre> # xa: utgångsvärde för x # ya: utgångsvärde för y # xb: slutligt värde för x # n : antal steg # Programmet använder Eulers stegmetod och returnerar approximation av y vid xb import math def f(x, y): return y/x def Euler(xa, ya, xb, n, f): h = (xb - xa) / float(n) x = xa y = ya for i in range(n): y = y + h * f(x, y) x = x + h return y print(Euler(1,2,4,3,f)) </pre> I programmet kan man också lätt variera steglängden.
12 – Nästlade satser	2 Mängder kombinatorik och grafer, Multiplikationsprincipen	2201 Pax väljer mellan tre par byxor och fyra skjortor när han ska på dejt. På hur många olika sätt kan han välja sin klädsel?	Med nästlade satser kan man skriva ett program som skriver ut alla möjliga klädval: <pre> byxor = ["byxa_1", "byxa_2", "byxa_3"] tröjor = ["tröja_1", "tröja_2", "tröja_3", "tröja_4"] for byxa in byxor: for tröja in tröjor: print(byxa, tröja) </pre>