

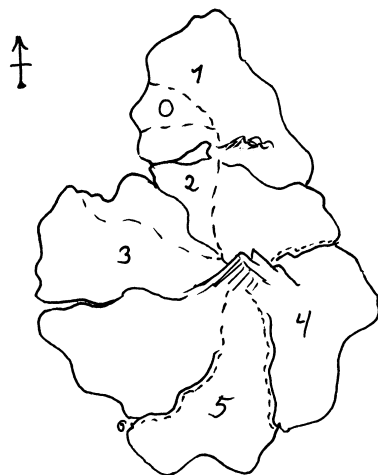
Fargelegg Ruritania

Harald Askestad
haraldas@ifi.uio.no

Fargelegg Ruritania

Ruritania's kartverk har nylig tatt i bruk datagrafikk. Kartet over en av Ruritania's største øyer skal fargelegges. Ett av problemene er at grensene ikke er helt klarlagte og det kan derfor være nødvendig å bruke flere enn fire farger på kartet. All programvaren baserer seg på at det kun finnes matematisk korrekte kart. Kartverket har derfor blitt nødt til å lage programvaren selv. Nå skal funksjonen som regner ut fargene testes.

RURITANIA



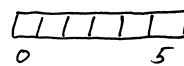
FYLKER M/FELLES GRENSE
REPRESENTERES I KART-OBJEKTET

$n = 6$

boolean felles:

	0				5
0	T	T			
	T	T	T	T	T
	T	T	T		
		T	T	T	T
5	T	T	T		

int farge:



(Her kommer løsningen)

Figur 1: Kart og representasjon av en av Ruritania's øyer. Fritt etter Athanasius Kircher (1644).

På figuren ser vi en gammel karttegning av øya, og til høyre ser vi hvordan informasjonen skal representeres i datamaskinen. De illustrerte variablene

ligger i et objekt av klassen *Kart*. Variabelen *n* angir antall land, *felles* er en matrise for hvilke fylker som har felles grenser, og i arrayen *farge* skal det komme en fargekode for hvert land.

Under ser vi *Kart*-klassen og et lite testprogram:

```
// Et testprogram for fargelegg
class FargeleggRuritania {
    public static void main (String args[]) {
        Kart ruritania = new Kart ();
        ruritania.fargelegg (0);
    }
}

class Kart {
    // Informasjon om kartet
    int n=6;
    boolean[][] felles = {{F,T,T,F,F,F},
                          {T,F,T,T,T,T},
                          {T,T,F,T,F,F},
                          {F,T,T,F,T,T},
                          {F,T,F,T,F,T},
                          {F,T,F,T,T,F}};

    static boolean T=true, F=false;

    // Løsningen
    int[] farge = new int[n];

    // Metoden som gjør jobben
    void fargelegg(int land) {
        if (land < n) {
            boolean[] brukt = new boolean[4]; // Farger brukt av naboland

            // Sjekker farger allerede brukt av naboland
            for (int nabo = 0; nabo < land; nabo++) {
                if (felles [land] [nabo]) {
                    brukt[farge[nabo]] = true;
                }
            }

            // Prøver de ledige fargene
            for (int f = 0; f <= 3; f++) {
                if (! brukt[f]) {
                    System.out.println("Prøver farge " + f + " på land " + land);
                    farge[land] = f;
                    fargelegg(land + 1);
                }
            }
        } // Slutt på første if

        else {
            System.out.println("Løsning:");
            for (int i = 0; i < n; i++) {
                System.out.print(farge[i]);
            }
            System.out.println();
            System.exit(0); // Avslutt etter første løsning
        }
    } // slutt på fargelegg
}
}
```

Programmet ga til slutt en løsnig:

Løsning:
012023