

## Sýnisláusnir á heimadæmum 5

1. Búa til fylki og leggja saman hliðstæð stök þess:

```
public class SumTwo {  
    public static void main(String[] args) {  
  
        int[] a = new int[100];  
        for (int i=0; i<100; i++)  
            a[i] = (int) (Math.random()*1000);  
  
        int[] b = new int[50];  
        for (int j=0; j<50; j++)  
            b[j] = a[2*j] + a[2*j+1];  
  
    }  
}
```

2. Athuga hvort öll gildi í inntaki séu ólík:

```
public class Unique {  
    public static void main(String[] args) {  
  
        int N = args.length;  
        int[] a = new int[N];  
        for (int i=0; i<N; i++)  
            a[i] = Integer.parseInt(args[i]);  
  
        boolean duplicates = false;  
        for (int i=0; i<N; i++) {  
            for (int j=i+1; j<N; j++) {  
                if (a[i] == a[j]) {  
                    System.out.println("Tvö eintök af " + a[i]);  
                    duplicates = true;  
                    break;  
                }  
            }  
        }  
        if (!duplicates)  
            System.out.println("Allar mismunandi");  
  
    }  
}
```

## 3. Sýna framá slæma dreifingu stokkunar:

```

public class BadShuffle {
    public static void main(String[] args) {
        int M = Integer.parseInt(args[0]);
        int N = Integer.parseInt(args[1]);

        int[] a = new int[N];
        int[][] b = new int[N][N];

        for (int k=0; k<M; k++) {
            // ein ítrun af hermun

            // upphafsstillla stokkunarfylki
            for (int i=0; i<N; i++)
                a[i] = i;

            // gera slæmu stokkun
            for (int i = 0; i < N; i++) {
                int r = (int) (Math.random()*N);
                int t = a[r];
                a[r] = a[i];
                a[i] = t;
            }

            // skrá niðurstöðuna
            for (int i = 0; i < N; i++) {
                b[a[i]][i]++;
            }
        }

        // prenta út niðurstöðu
        for (int i=0; i<N; i++) {
            for (int j=0; j<N; j++)
                System.out.print(b[i][j] + " ");
            System.out.println();
        }
    }
}

```

## Útkoma:

```

% java BadShuffle 200000 8
25136 25145 24993 24948 25032 24855 24853 25038
31305 23139 23435 23805 24032 24315 24930 25039
29197 30404 21885 22687 22982 23570 24088 25187
26652 27971 29424 21713 22300 23076 24055 24809
24381 26059 27616 29166 21424 22824 23440 25090
22750 23805 25827 27582 29611 21981 23525 24919
20993 22359 23964 25775 28092 30459 23400 24958
19586 21118 22856 24324 26527 28920 31709 24960

```

Sést að það er mikill munur á milli staka rétt fyrir neðan hornalínuna (rúmlega 30 þús.) og staka neðst til vinstri (rúmlega 20 þús.).

## 4. Dæmi 1.4.17 í kennslubók. Margfalda saman tvö fylki af mism. víddum:

```
public class MatrixMul {  
    public static void main(String[] args) {  
        int M = Integer.parseInt(args[0]);  
        int N = Integer.parseInt(args[1]);  
        int P = Integer.parseInt(args[2]);  
  
        double[][] a = new double[M][N];  
        double[][] b = new double[N][P];  
  
        for (int i=0; i<M; i++)  
            for (int j=0; j<N; j++)  
                a[i][j] = Math.random();  
  
        for (int i=0; i<N; i++)  
            for (int j=0; j<P; j++)  
                b[i][j] = Math.random();  
  
        double[][] c;  
  
        if (a[0].length != b.length)  
            System.out.println("Villa, víddir passa ekki");  
        else {  
            c = new double[a.length][b[0].length];  
            for (int i=0; i<a.length; i++)  
                for (int j=0; j<b[0].length; j++)  
                    for (int k=0; k<a[0].length; k++)  
                        c[i][j] += a[i][k]*b[k][j];  
        }  
    }  
}
```

Það þurfti aðeins að skila inn neðsta hlutanum, þ.e. þar sem fylkin eru margfölduð.

## 5. Athuga hvort lögleg lausn á Sudoku:

```
public class Sudoku {
    public static void main(String[] args) {
        // Fylki sem er lögleg lausn á Sudoku þraut
        int[][] s = {
            { 5, 3, 4, 6, 7, 8, 9, 1, 2 },
            { 6, 7, 2, 1, 9, 5, 3, 4, 8 },
            { 1, 9, 8, 3, 4, 2, 5, 6, 7 },
            { 8, 5, 9, 7, 6, 1, 4, 2, 3 },
            { 4, 2, 6, 8, 5, 3, 7, 9, 1 },
            { 7, 1, 3, 9, 2, 4, 8, 5, 6 },
            { 9, 6, 1, 5, 3, 7, 2, 8, 4 },
            { 2, 8, 7, 4, 1, 9, 6, 3, 5 },
            { 3, 4, 5, 2, 8, 6, 1, 7, 9 }
        };

        // Fylki sem er mjög líklega ekki lögleg lausn
        int[][] t = new int[9][9];
        for (int i=0; i<9; i++)
            for (int j=0; j<9; j++)
                t[i][j] = (int)(Math.random()*9)+1;

        boolean alltILagi = true;

        // athuga fyrst hvort línur í lagi
        for (int i=0; i<9; i++) {

            // notum ekki ath[0], bara ath[1] til ath[10]
            boolean[] ath = new boolean[10];
            for (int j=0; j<9; j++)
                ath[t[i][j]] = true;

            for (int k=1; k<10; k++) {
                if (!ath[k]) {
                    System.out.println("Vandamál í línu " + i);
                    alltILagi = false;
                    break;
                }
            }
        }

        // athuga svo hvort dálkar í lagi
        for (int j=0; j<9; j++) {

            // notum ekki ath[0], bara ath[1] til ath[10]
            boolean[] ath = new boolean[10];
            for (int i=0; i<9; i++)
                ath[t[i][j]] = true;

            for (int k=1; k<10; k++) {
                if (!ath[k]) {
                    System.out.println("Vandamál í dálki " + j);
                    alltILagi = false;
                    break;
                }
            }
        }
    }
}
```

```
        }  
    }  
}  
  
    if (alltILagi)  
        System.out.println("Lögleg lausn");  
}  
}
```