

Sáld Eratospenesar

Til að athuga hvort N sé prímtala

- Einfaldur kóði tekur $N^{1/2}$ aðgerðir
- Prufum allar tölur frá 2 ... $N^{1/2}$
- Hægt að gera betur $\log(N)^{7.5}$

Til að finna allar prímtölur $\leq N$

- Prufum allar tölur 2 ... N
- Kostar $N * N^{1/2} = N^{3/2}$

Hægt að gera betur

Sáld Eratospenesar

Búum til boolean fylki `isPrime` til að halda utan um hvort `i` sé prímtala, `isPrime[i]`, satt í upphafi

Fyrir `i` frá 2 ... \sqrt{N}

1. Ef `i` er ekki merkt sem prímtala þá þurfum við ekki að gera neitt
2. Ef `i` er merkt sem prímtala þá er `i` prímtala merkjum öll margfeldi af `i` sem ekki prímtölur

Sáld Eratospenesar

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

        // initially assume all integers are prime
        boolean[] isPrime = new boolean[N+1];
        for (int i = 2; i <= N; i++)
            isPrime[i] = true;

        // mark non-primes <= N using Sieve of Eratosthenes
        for (int i = 2; i*i <= N; i++) {
            if (isPrime[i]) {
                for (int j = i; i*j <= N; j++)
                    isPrime[i*j] = false;
            }
        }

        // count primes
        int primes = 0;
        for (int i = 2; i <= N; i++)
            if (isPrime[i]) primes++;
        StdOut.println("The number of primes <= " + N + " is " + primes);
    }
}
```

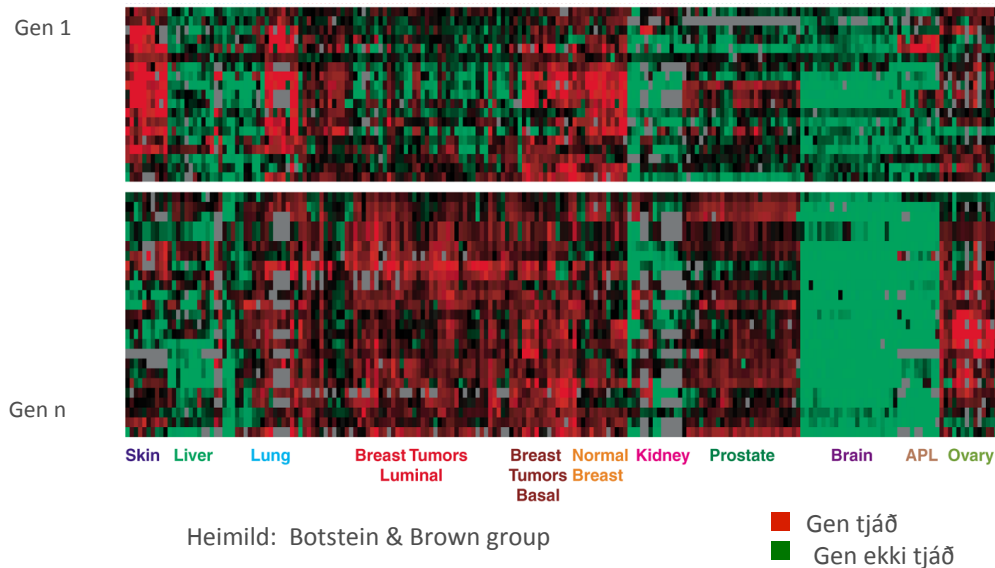
Margvíð fylki

Margvíð fylki eru notuð fyrir

- Töflur af gögnum, margar tilraunir á mörgum efnum
- Töflur af einkunum, hver nemandi gerir mörg verkefni
- Töflur af litum, gráskali fyrir hvern pixel í mynd

Stærðfræðihugtak: Fylki (kassar af tölum)

Java hugtak: Tvívítt fylki (fylki af fylkjum)



Tvívíð fylki í Java

Gildi í fylkjum: notum `a[i][j]` til að vísa í röð `i` og dálk `j`. Byrjum að telja í 0

```
int M = 10;
int N = 3;
double[][] a = new double[M][N];
for (int i = 0; i < M; i++) {
    for (int j = 0; j < N; j++) {
        a[i][j] = 0.0;
    }
}
```

`a.length` gefur fjölda raða
`a[0].length` gefur fjölda dálka

The diagram shows a 10x3 array of double values. The rows are labeled from a[0] to a[9]. The columns are labeled from a[i][0] to a[i][2]. The row a[5] is highlighted in grey. Blue arrows point from the labels 'a[][]' and 'a[5]→' to the top-left and the highlighted row, respectively.

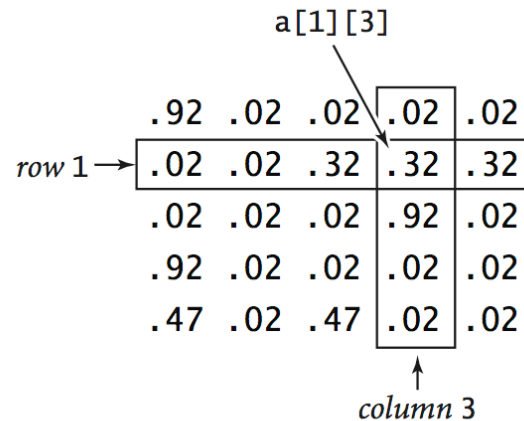
a[0][0]	a[0][1]	a[0][2]
a[1][0]	a[1][1]	a[1][2]
a[2][0]	a[2][1]	a[2][2]
a[3][0]	a[3][1]	a[3][2]
a[4][0]	a[4][1]	a[4][2]
a[5][0]	a[5][1]	a[5][2]
a[6][0]	a[6][1]	a[6][2]
a[7][0]	a[7][1]	a[7][2]
a[8][0]	a[8][1]	a[8][2]
a[9][0]	a[9][1]	a[9][2]

A 10-by-3 array

Tvívíð fylki í Java

Getum líka upphafsstillt fylki í kóða

```
double[][] p = {  
    { .02, .92, .02, .02, .02 },  
    { .02, .02, .32, .32, .32 },  
    { .02, .02, .02, .92, .02 },  
    { .92, .02, .02, .02, .02 },  
    { .47, .02, .47, .02, .02 },  
};
```



Fylkjasamlagning

Ef a og b eru $N \times N$ þá skilgreinum við $c = a + b$, þar sem c er $N \times N$ fylki og

$$c[i][j] = a[i][j] + b[i][j]$$

```
double[][] c = new double[N][N];
for (int i = 0; i < N; i++)
    for (int j = 0; j < N; j++)
        c[i][j] = a[i][j] + b[i][j];
```

$a[][]$

.70	.20	.10
.30	.60	.10
.50	.10	.40

$a[1][2]$

$b[][]$

.80	.30	.50
.10	.40	.10
.10	.30	.40

$b[1][2]$

$c[][]$

1.5	.50	.60
.40	1.0	.20
.60	.40	.80

$c[1][2]$

Fylkjamargföldun

Ef a, b eru $N \times N$ fylki þá skilgreinum við $c = a * b$, þar sem c er $N \times N$ fylki og $c[i][j]$ er innfeldi i -tu raðar a og j -ta dálks b

```
double[][] c = new double[N][N];
for (int i = 0; i < N; i++)
    for (int j = 0; j < N; j++)
        for (int k = 0; k < N; k++)
            c[i][j] += a[i][k] * b[k][j];
```

$a[][]$

.70	.20	.10
.30	.60	.10
.50	.10	.40

← row 1

$b[][]$

column 2
↓

.80	.30	.50
.10	.40	.10
.10	.30	.40

$$\begin{aligned} c[1][2] &= .3 * .5 \\ &+ .6 * .1 \\ &+ .1 * .4 \\ &= .25 \end{aligned}$$

$c[][]$

.59	.32	.41
.31	.36	.25
.45	.31	.42

Fylkjamargföldun

Hversu margar aðgerðir keyrum við?

A. N

B. N^2

C. N^3

D. N^4

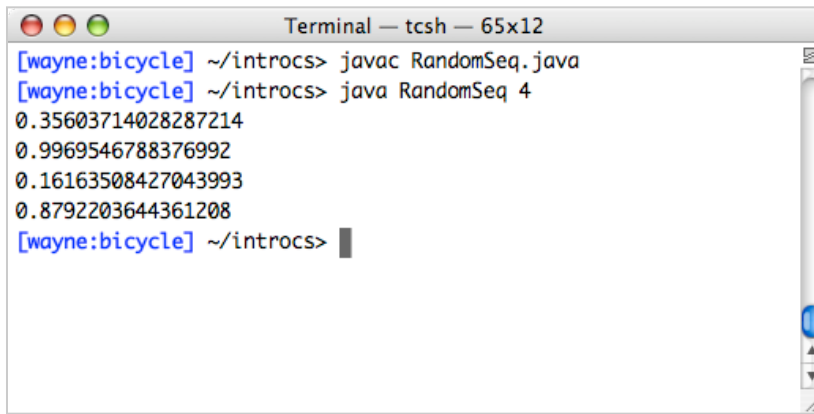
```
double[][] c = new double[N][N];
for (int i = 0; i < N; i++)
    for (int j = 0; j < N; j++)
        for (int k = 0; k < N; k++)
            c[i][j] += a[i][k] * b[k][j];
```

Inntak/úttak

Við höfum notað skel (terminal) til að þýða og kalla á Java forrit.

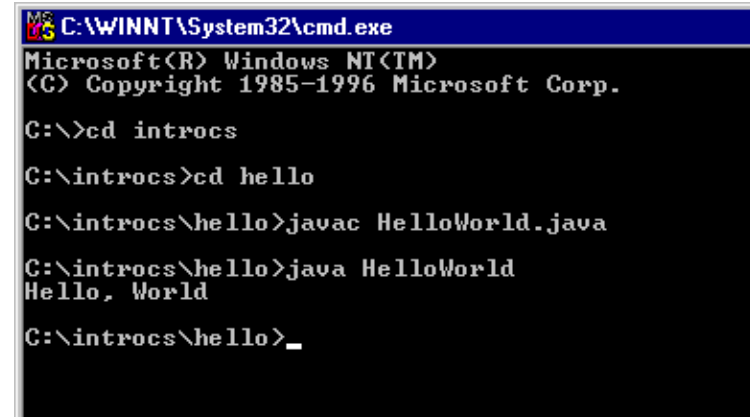
Lesið viðföng (arguments) af skelinni (args[0], args[1]).

Hentar fyrir nokkrar breytur, ekki mikið af gögnum



```
Terminal — tcsh — 65x12
[wayne:bicycle] ~/introscs> javac RandomSeq.java
[wayne:bicycle] ~/introscs> java RandomSeq 4
0.35603714028287214
0.9969546788376992
0.16163508427043993
0.8792203644361208
[wayne:bicycle] ~/introscs> |
```

Mac OS X



```
C:\WINNT\System32\cmd.exe
Microsoft(R) Windows NT(TM)
(C) Copyright 1985-1996 Microsoft Corp.

C:\>cd introscs
C:\introscs>cd hello
C:\introscs\hello>javac HelloWorld.java
C:\introscs\hello>java HelloWorld
Hello, World
C:\introscs\hello>_
```

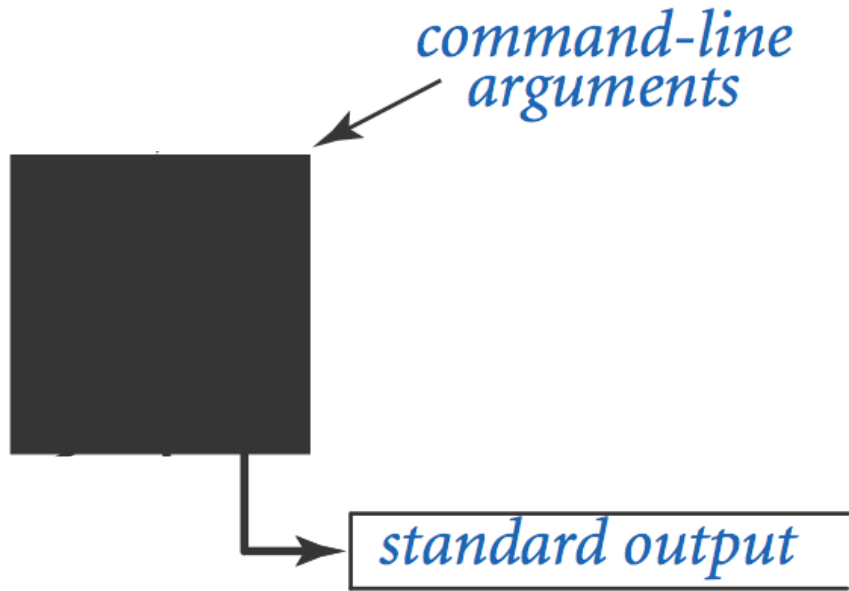
Microsoft Windows

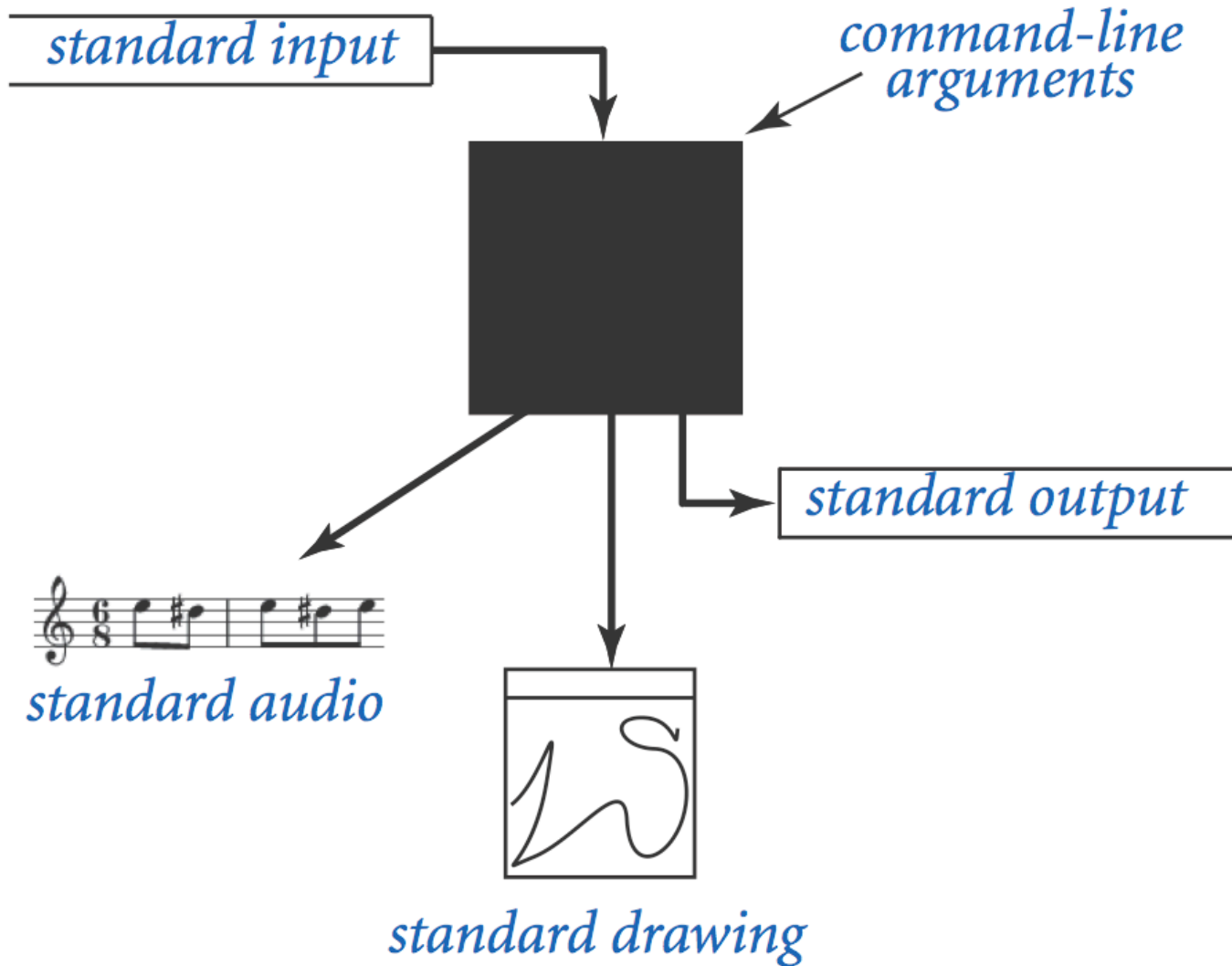
Staðal inntak/úttak (standard input, standard output) einnig kallað stdin og stdout

- Uppruni: Unix 1970
- Einföld og þægileg leið til að lesa inn og skrifa út gögn
- Í Java skrifar `System.out.println()` á stdout
- Skrifar út á skjá nema annað sé tekið fram

```
public class RandomSeq {  
    public static void main(String[] args) {  
        int N = Integer.parseInt(args[0]);  
        for (int i = 0; i < N; i++) {  
            System.out.println(Math.random());  
        }  
    }  
}
```

```
% java RandomSeq 4  
0.9320744627218469  
0.4279508713950715  
0.08994615071160994  
0.6579792663546435
```





Staðalinntak

Viðföng af skipanalínu eru lesin inn áður en kallað er á `main(String[] args)`

Staðalinntak (`stdin`) er lesið á meðan keyrslu stendur og hægt að lesa hvenær sem er. Við vitum ekki fyrirfram hversu mikið kemur inn.

`StdIn` er klasi sem fylgir bókinn, einfaldar lestur af `stdin`

StdIn

Til að nota þarf að ná í `StdIn.java` og setja í sömu möppu og forrit.

```
public class Add {  
    public static void main(String[] args) {  
        StdOut.print("Type the first integer: ");  
        int x = StdIn.readInt();  
        StdOut.print("Type the second integer: ");  
        int y = StdIn.readInt();  
        int sum = x + y;  
        StdOut.println("Their sum is " + sum);  
    }  
}
```

```
% java Add  
Type the first integer: 1  
Type the second integer: 2  
Their sum is 3
```

Meðaltal talna

Lesum inn straum af tölum og skrifum út meðaltal

```
public class Average {  
    public static void main(String[] args) {  
        double sum = 0.0; // cumulative total  
        int n = 0; // number of values  
  
        while (!StdIn.isEmpty()) {  
            double x = StdIn.readDouble();  
            sum = sum + x;  
            n++;  
        }  
  
        StdOut.println(sum / n);  
    }  
}
```

```
% java Average  
10.0 5.0 6.0  
3.0 7.0 32.0  
<Ctrl-d>  
10.5
```

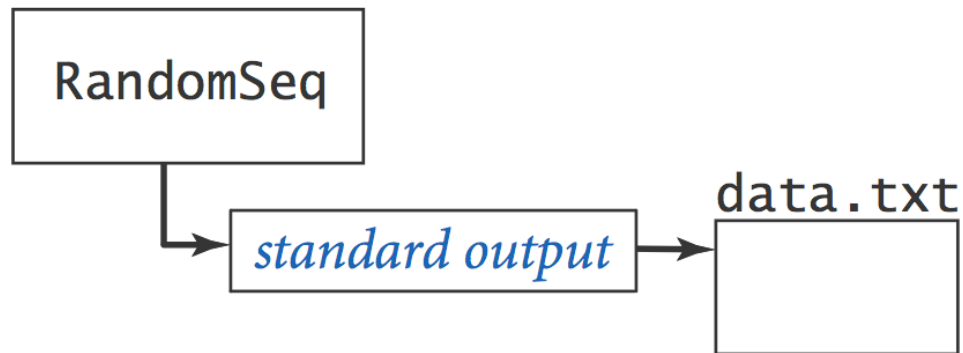
<Ctrl-d> fyrir mac/linux

<Ctrl-z> fyrir Windows

Ótakmarkaður fjöldi talna

Tenging forrita

Getum stýrt úttaki með pípum

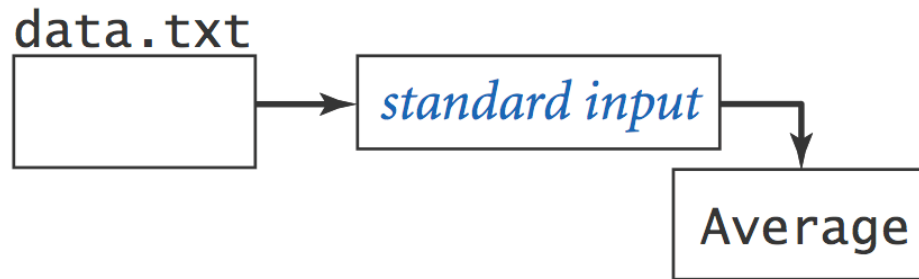


```
% java RandomSeq 1000 > data.txt
```

pípum stdout yfir í
data.txt

Tenging forrita

Getum stýrt staðalinntaki inn í forrit



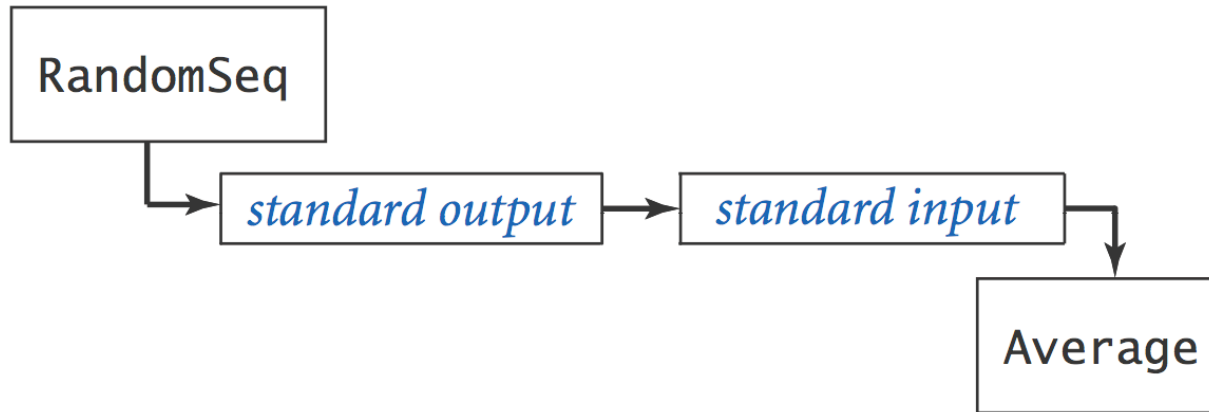
```
% less < data.txt  
0.5475375782884312  
0.4971087292684019  
0.23123808041753813  
...
```

pípum staðalinntaki inn í forrit

```
% java Average < data.txt  
0.4947655567740991
```

Tenging forrita

Getum tengt úttak eins forrits við inntak annars



Pípum úttaki `RandomSeq` inn í inntak `Average`

```
% java RandomSeq 1000000 | java Average  
0.4997970473016028  
  
% java RandomSeq 1000000 | java Average  
0.5002071875644842
```