

```
int anz;
```

```
if ( (anz >= 91) || (anz <= 100) )
    printf("sehr gut");    ← Kosten 1
else
    if ( (anz >= 71) || (anz <= 80) )
        printf("gut");    ← Kosten 2
    else
        ...
```

mit (allen) "else":
1 bis 5 Bedingungen
werden überprüft

ohne (alle) "else":
alle 5 Bedingungen
werden überprüft

```
if (Fall 1)
    ...
else
    if (Fall 2)
        ...
    else
        if (Fall 3)
            ...
        ...
        Fall 5
```

1. Ann.: alle Fälle sind gleich
wahrscheinlich
 $p(i) = \frac{1}{5}$

→ Mittlere Anzahl an aus-
gewerteten Bedingungen *
ihren Kosten:

$$1 * \frac{1}{5} + 2 * \frac{1}{5} + 3 * \frac{1}{5} + \dots$$

$$= \frac{1}{5} * \sum_{i=1}^5 i$$

$$= \frac{1}{5} * \frac{5 * (5+1)}{2}$$

$$= 3$$

Allgemein: $\frac{n+1}{2}$

Exponential Verteilung

$$p(i) = 2^{-i} \quad (1 - \frac{1}{2}, 2 - \frac{1}{4}, \dots)$$

Ansatz: sortiere Fälle nach ihrer
Wahrscheinlichkeit

if (Fall 1)	Kosten
...	
else	1 * $\frac{1}{2}$
if (Fall 2)	+
...	
else	2 * $\frac{1}{4}$
...	
...	...

also

$$\begin{array}{r} \dots \\ \text{if (Fall 3)} \\ \vdots \\ \end{array} \quad \begin{array}{r} \text{---} \quad 2 * \frac{1}{4} \\ + \\ \text{---} \quad 3 * \frac{1}{8} \\ \vdots \\ \end{array}$$

≤ 2

Switch: `switch (sel.expr) {` ↓ Typ unsigned int

<code>case 0: ...</code>	<code>2:</code>
<code>case 1: ...</code>	<code>1:</code>
<code>case 2: ...</code>	<code>3:</code>
<code>case 3: ...</code>	<code>0:</code>
<code>;</code>	

`}`

if (sel.expr > 3)
dann default
sonst
gehe zu
case i
(i := sel.expr)

Konstante Laufzeit

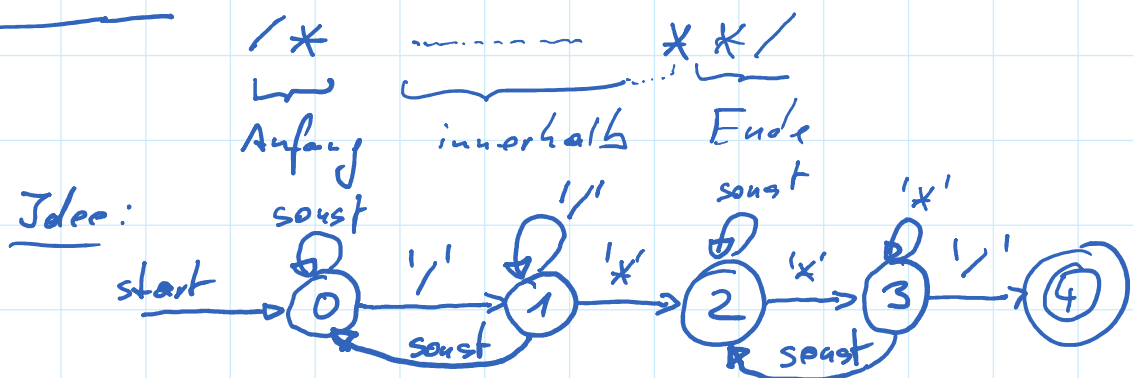
lineare Laufzeit

if else
if else...

Zusatz: sel.expr hat Typ unsigned int
⇒ Konstanten bei Cases auch.

case 1:
case 'A':

Beispiel: Kommentar in C



zustand = 0;
wiederhole lies ein Zeichen c und tue jeweils:
switch (zustand) {

```
case 0: if (c == '/')
        zustand = 1;
        else zustand = 0;
        break;
case 1: if (c == 'x')
        zustand = 2;
        else if (c == '/')
            zustand = 1;
        else zustand = 0;
        break;
```