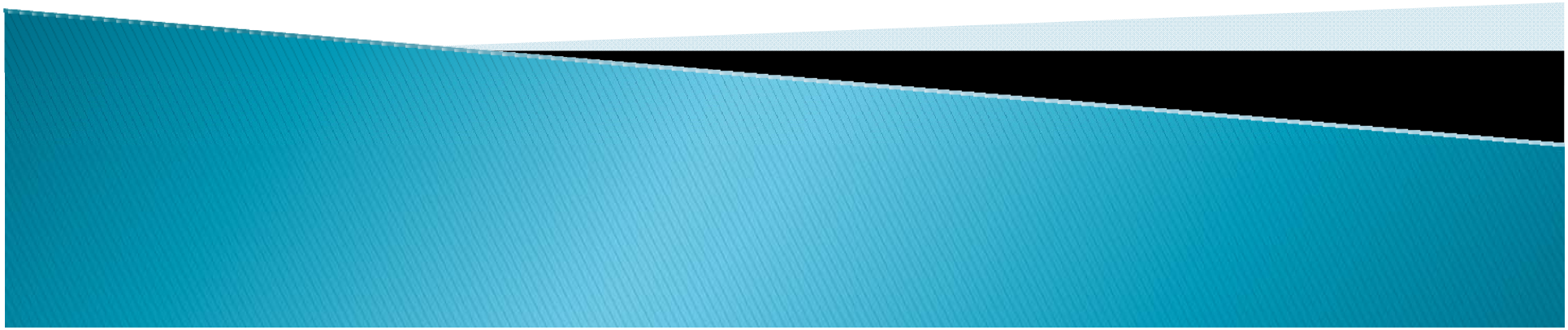
HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

Angewandte Mathematik

Marcel Sachse, Fabian Seidl

SS09 IFB2C



Gelöste Probleme

- Teiler, Teilbarkeit, Primzahlen
  - #382 Perfection
  - #412 Pi
- Zahlen, Zahlen, Zahlen
  - #136 Ugly Numbers
  - #443 Humble Numbers
  - #944 Happy Numbers
  - #10042 Smith Numbers
  - #10579 Fibonacci Numbers
- Große Zahlen, Modulo
  - #374 Big Mod
- Verschiedene Formeln
  - #579 Clock Hands
  - #993 Product of Digits
  - #10162 Last Digit
  - #11152 Colourful Flowers
  - #11526 H(n)
- Basisumwandlung
  - #575 Sew Binary
  - #10469 To Carry or not to Carry
  - #10473 Simple Base Conversion
- Zeichenketten
  - #10082 WERTYU
  - #10222 Decode the Mad man
  - #10679 I Love Strings!
  - #11223 O: dah, dah, dah!

Schönste Probleme

I. The Return of the Roman Empire #759

II. Colourful Flowers #11152

III. Big Chocolate #10970

Schönste Programme

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

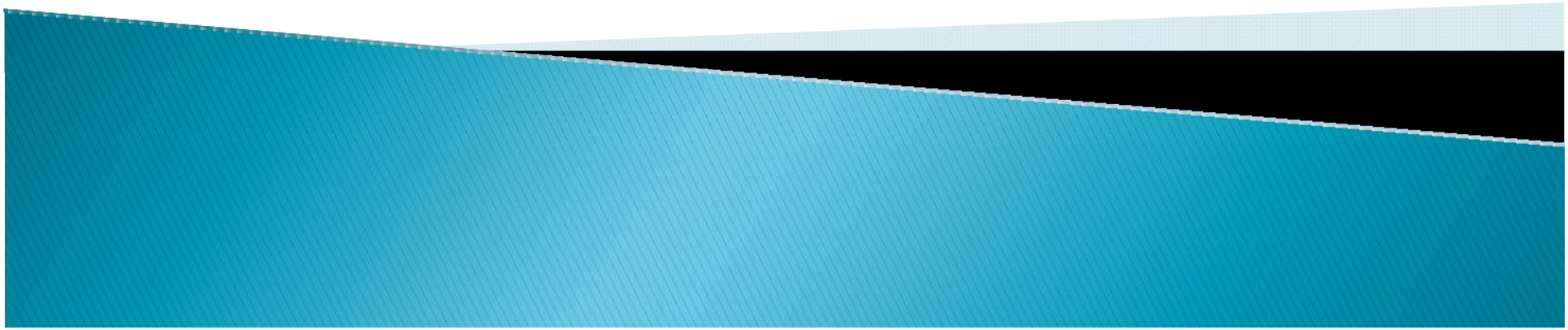
I. Simple Base Conversion #10473
   Christian Posselt, Jonathan Schubert
II. Largest Prime Devisor #11466
   Christian Posselt, Jonathan Schubert
III. Odd Sum #10783
   Christian Mitterreiter, Rolf Luigs

Ausgewählte Probleme

- Happy Numbers #944
- Product of Digits #993
- Ugly Numbers #136
- Humble Numbers #443
- Last Digit #10162

HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

#944 Happy Numbers



944 Happy Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Was ist eine Happy Number?

- Quadrate der Stellen bilden
- Quadrate aufsummieren
- von Summe die Stellenquadrate bilden
- Summe 1
  -> Happy
- Zyklus (4, 16, 37, 58, 89, 145, 42, 20, 4)
  -> unhappy

944 Happy Numbers

- Beispiel Zahl 7:

- 7^2 -> 49
- 4^2+9^2 -> 97
- 9^2+7^2 -> 130
- 1^2+3^2+0^2 -> 10
- 1^2+0^2 -> 1
- 1^2 -> 1

- 7 ist eine Happy Number da 1

944 Happy Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Problem:

- Intervall gegeben
- alle Happy Numbers finden
- in wie viel Schritten ist Quadratsumme 1
- alle Happy Numbers Ausgeben
- Schritte für Quadratsumme Ausgeben

944 Happy Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Input/Output

Input

- 2 Zahlen pro Zeile L und H (1 <= 99999)
- L <= H
- Ende: EOF (Ende des Files)

Output

- gefundene Happy Number (HN)
- Anzahl der Quadratsummen bis 1 (QS)
- neue Zeile für HN QS
- zwischen 2 Intervallen eine Leerzeile

944 Happy Numbers

Beispiel:

Input:

5 28
233 250

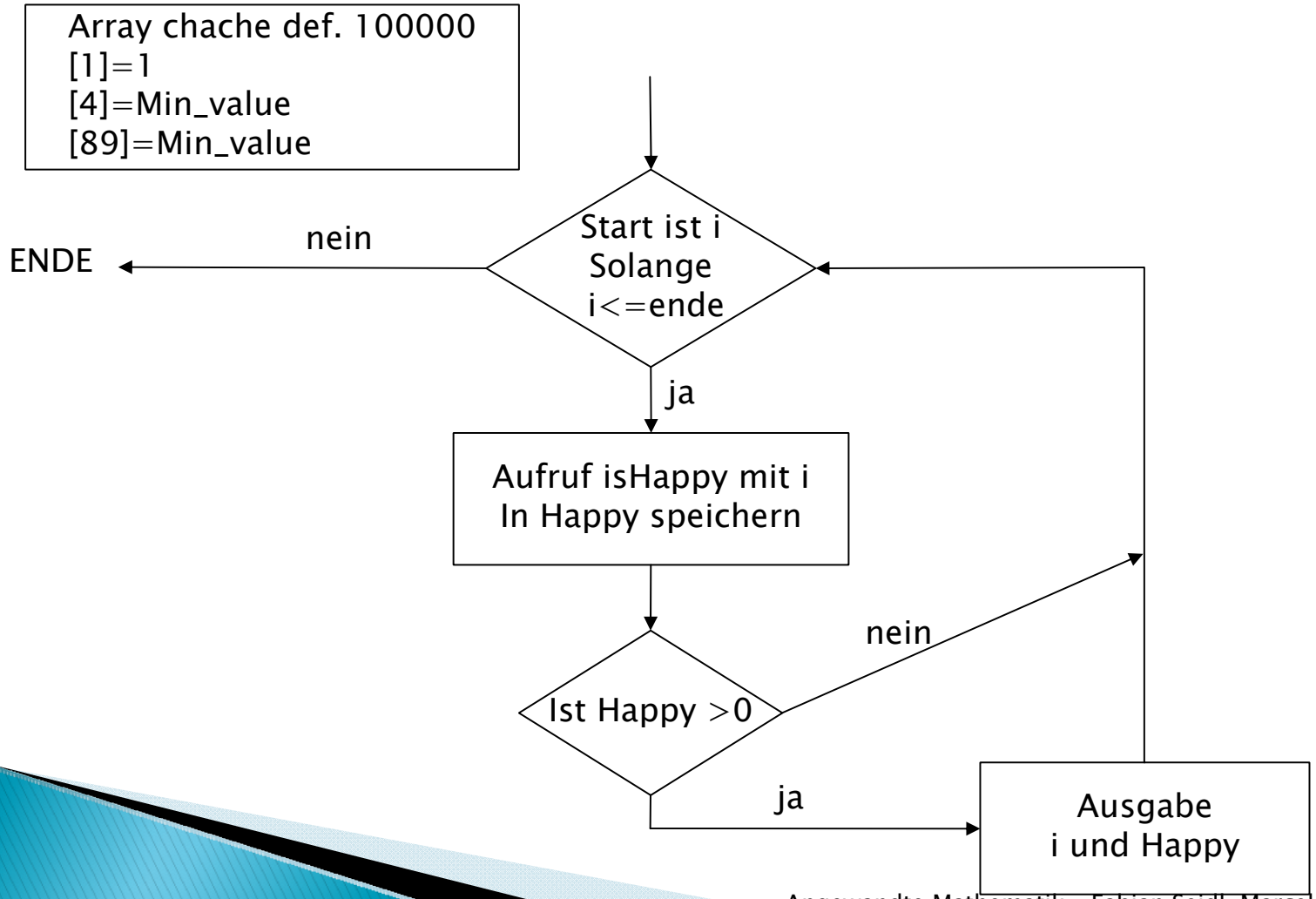
Output:

7 6
10 2
13 3
19 5
23 4
28 4

236 6
239 6

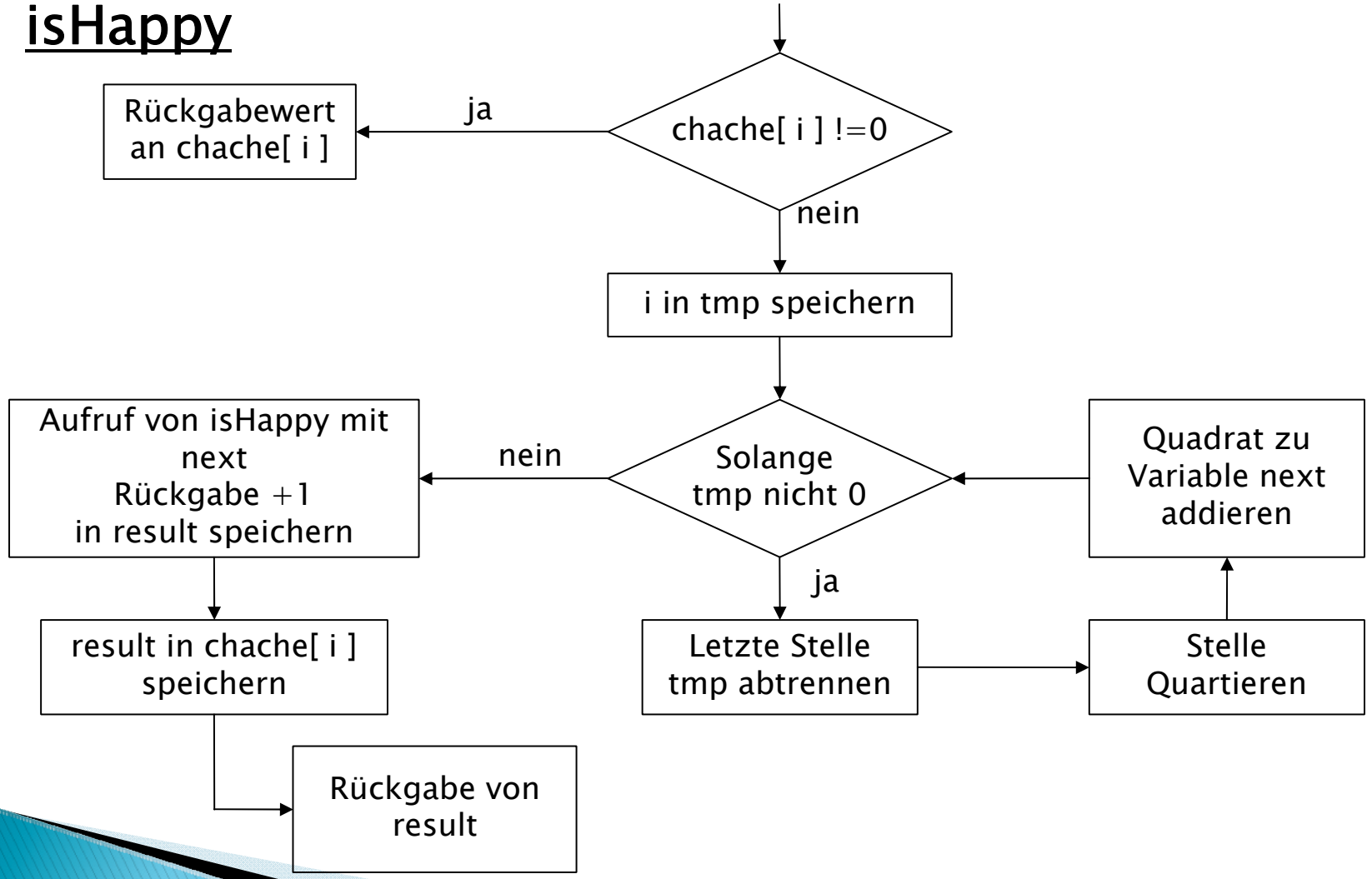
944 Happy Numbers

Hauptprogramm



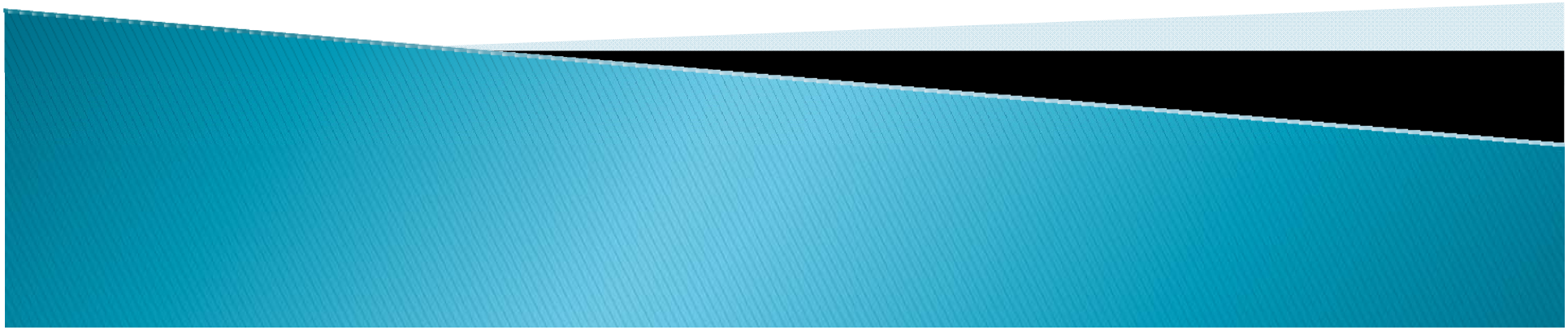
944 Happy Numbers

isHappy



HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

#993 Product of Digits



993 Product of Digits

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Problem:

- Zahl N ist gegeben
- Zahl Q ist gesucht
- Produkt aller der Stellen von Q ergibt N
- Q muss Minimal sein

993 Product of Digits

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Input

- 1.Zeile Anzahl Testfälle
- Ein N pro Zeile
- N ist ($0<=N<=10^9$)

Output

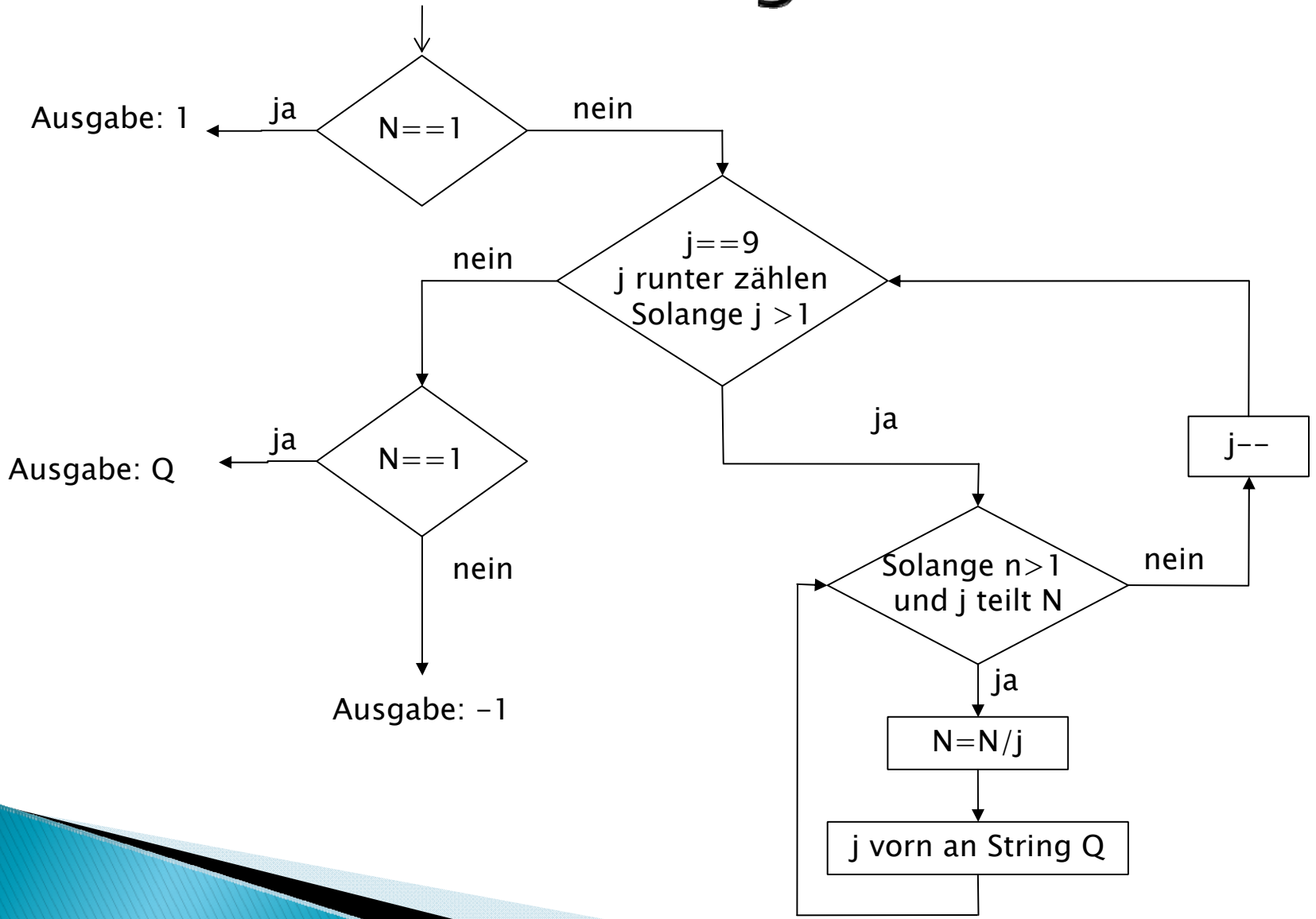
- Für jedes N ein Q pro Zeile
- Wenn kein Q existiert -1

993 Product of Digits

Beispiel:

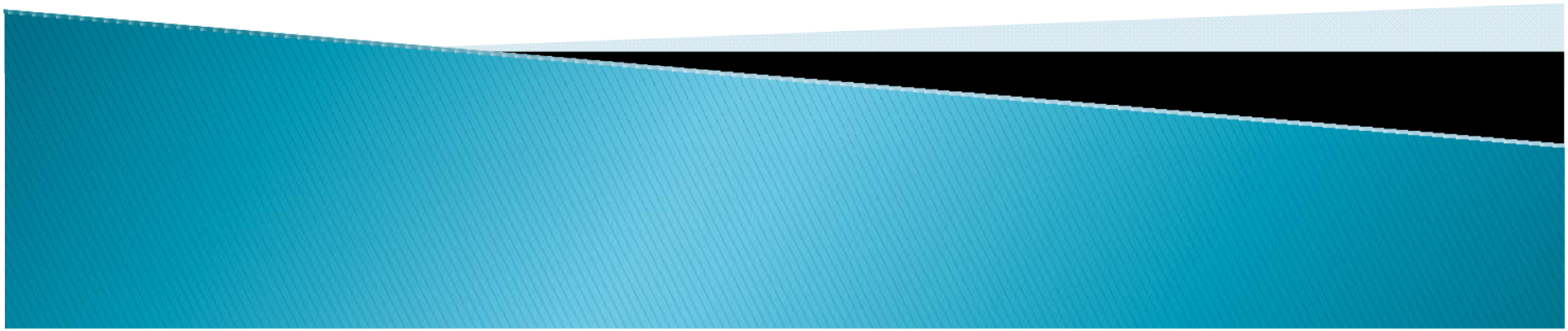
Input:	Output:
3	
1	1
10	25
123456789	-1

993 Product of Digits



HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

#136 Ugly Numbers



#136 Ugly Numbers

- Zahlen, die nur die Primfaktoren 2,3 und 5 enthalten
- 1, 2, 3, 4, 5, 6, 8, 9, 10, 12, 15, ...
- Es ist die 1500te Zahl auszugeben (kein Input)

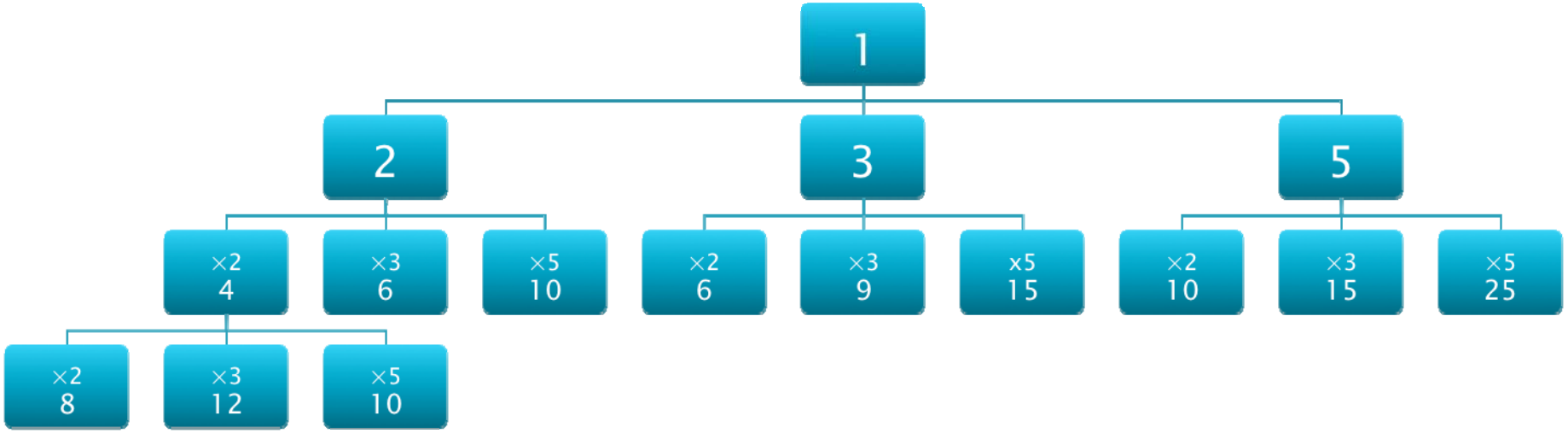
#136 Ugly Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

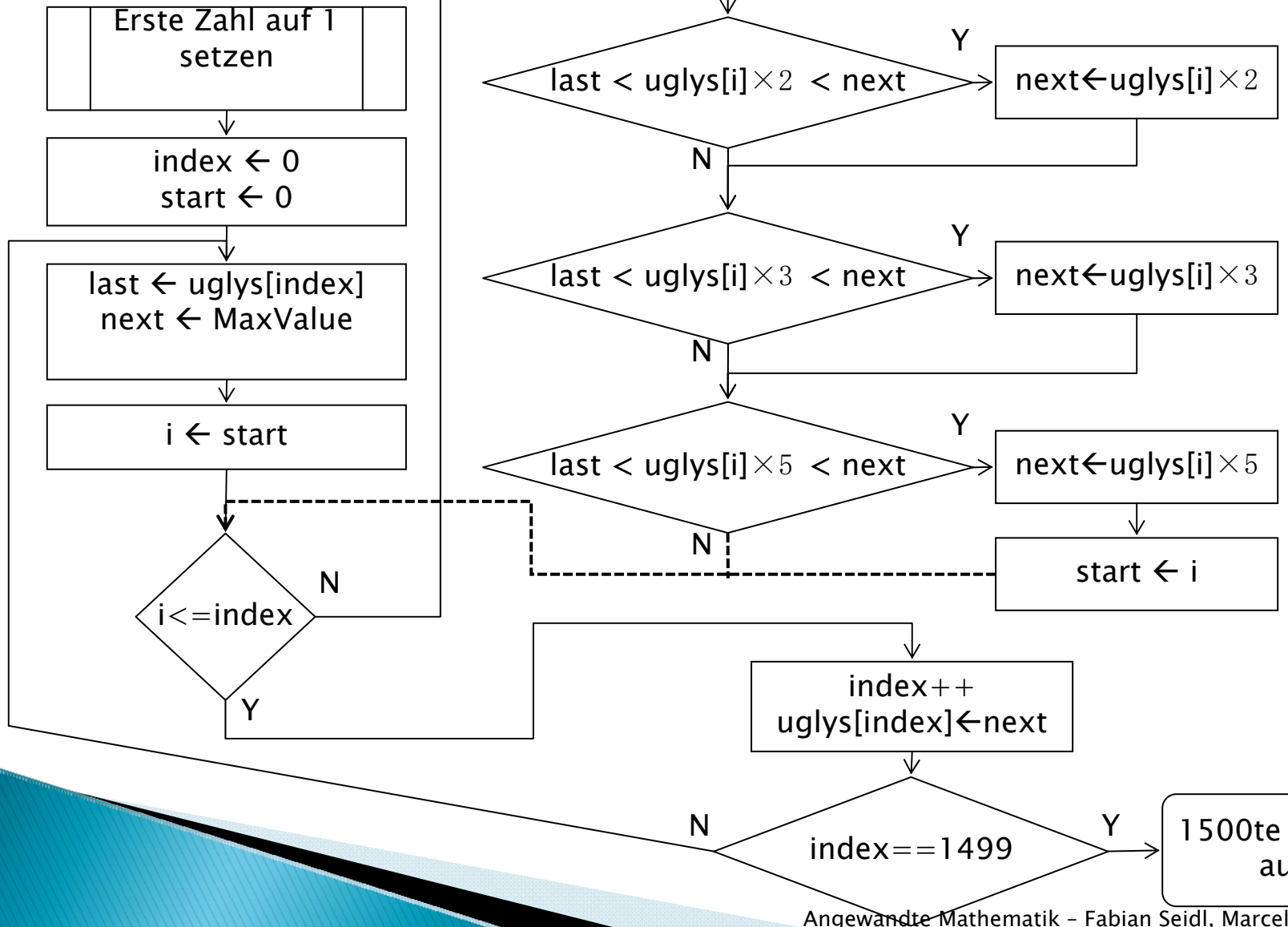
Lösungsansätze:

- Für alle natürlichen zahlen Prüfen, ob sie nur 2,3,5 als Primteiler haben
- Besser: jeder Zahl aus den vorangegangenen Zahlen durch Multiplikation von 2,3,5 erzeugen
- bei dieser Aufgabe: Precalculation 1500ten Ugly Number

#136 Ugly Numbers

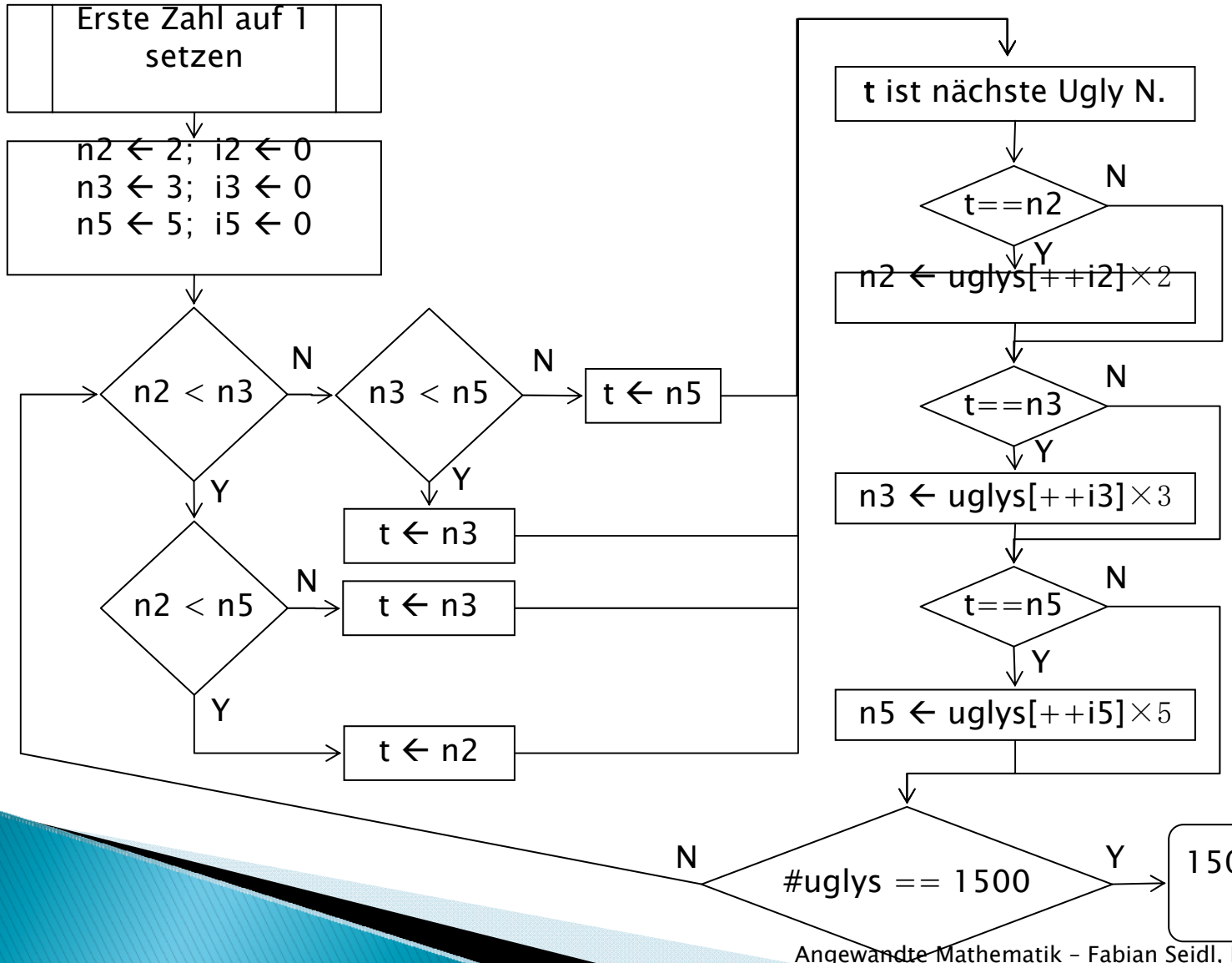


#136 Ugly Numbers



uglys	
0	1
1	2
2	3
3	4
4	5
5	6
6	8
7	9
8	10
9	12
10	15
11	16
12	18

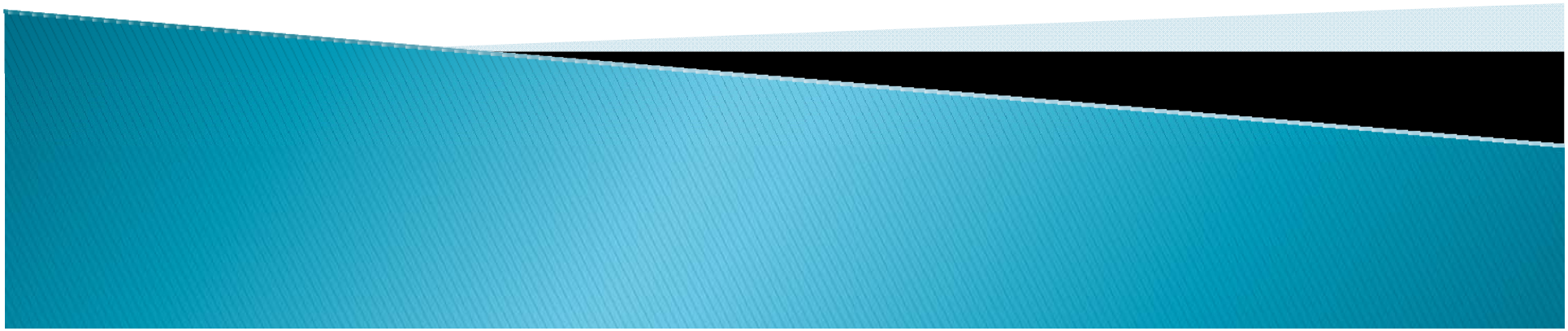
#136 Ugly Numbers



uglys	
0	1
1	2
2	3
3	4
4	5
5	6
6	8
7	9
8	10
9	12
10	15
11	16
12	18

HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

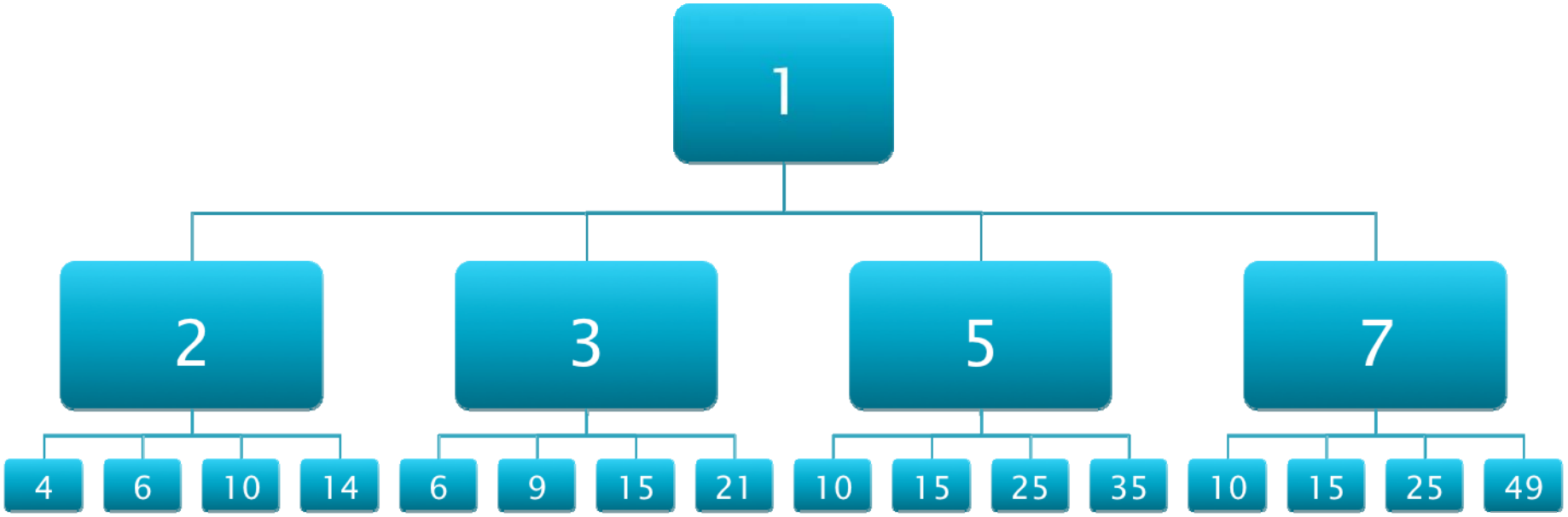
#443 Humble Numbers



#443 Humble Numbers

- Humble Numbers sind Zahlen die nur die Primfaktoren 2,3,5,7 besitzen
- 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 24, 25, 27, ...
- es ist die n-te Humble Number auszugeben

#443 Humble Numbers



#443 Humble Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

- Input
  - ein oder mehrere Testfälle mit $1 \leq n \leq 5842$
  - beendet mit 0

- Output
  - Für jeden Testfall eine Zeile:
    „The <n>th humble number is <number>.“
  - abhängig von n richtiger suffix
    (st, nd, rd, th)

#443 Humble Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

- Sample Input:

  1
  2
  3
  4
  11
  12
  13
  21
  22
  23
  100
  1000
  5842
  0

- Sample Output:

  The 1st humble number is 1.
  The 2nd humble number is 2.
  The 3rd humble number is 3.
  The 4th humble number is 4.
  The 11th humble number is 12.
  The 12th humble number is 14.
  The 13th humble number is 15.
  The 21st humble number is 28.
  The 22nd humble number is 30.
  The 23rd humble number is 32.
  The 100th humble number is 450.
  The 1000th humble number is 385875.
  The 5842nd humble number is 2000000000.

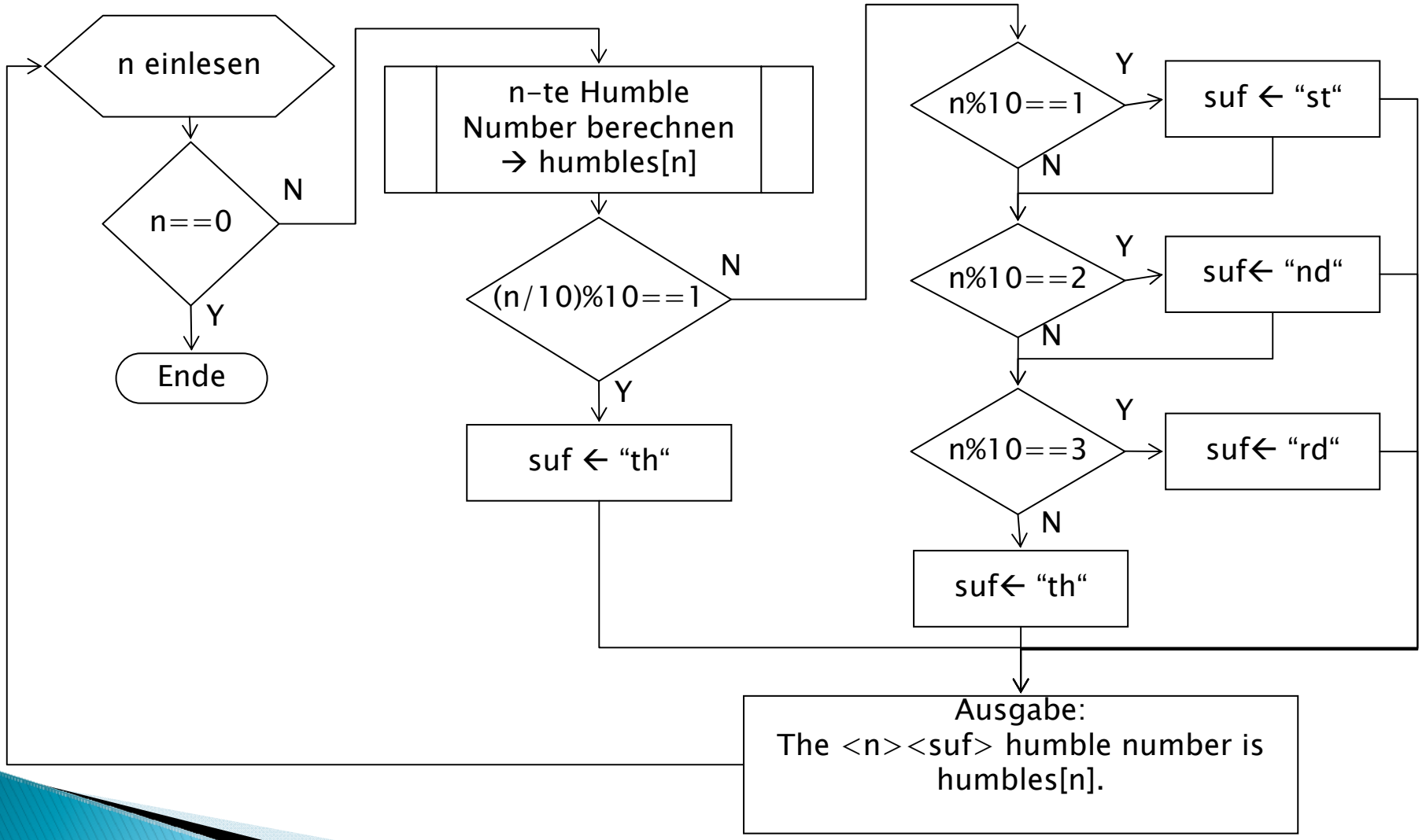
#443 Humble Numbers

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Lösungsansatz:

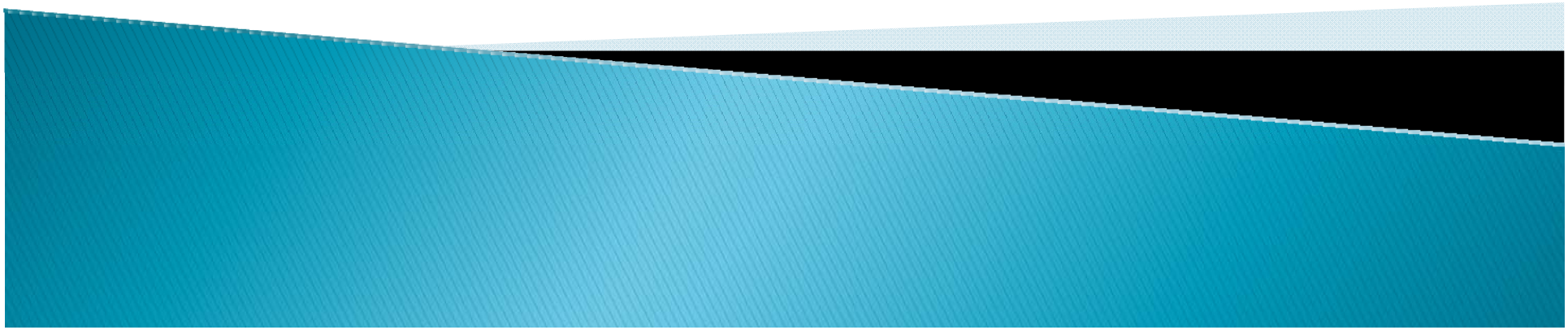
- Zahlen Analog zu den Ugly Numbers berechnen und speichern
- Suffix
  - 10–19: th
  - Ansonsten:
    ...1: st
    ...2: nd
    ...3: rd
    rest: th

#443 Humble Numbers



HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

#10162 Last Digit



#10162 Last Digit

- Geg.: Integer N, $1 \leq N \leq 2*10^{100}$
- zu berechnen ist die letzte Ziffer von

$$S = 1^1 + 2^2 + 3^3 + \ldots + N^N$$

- Input: mehrere N, abgeschlossen mit 0
- Output: die letzte Ziffer von S

#10162 Last Digit

- Sample Input:

  1
  2
  3
  101
  102
  103
  201
  202
  203
  78799801
  78799802
  78799803
  0

- Sample Output:

  1
  5
  2
  1
  5
  2
  1
  5
  2
  1
  5
  2

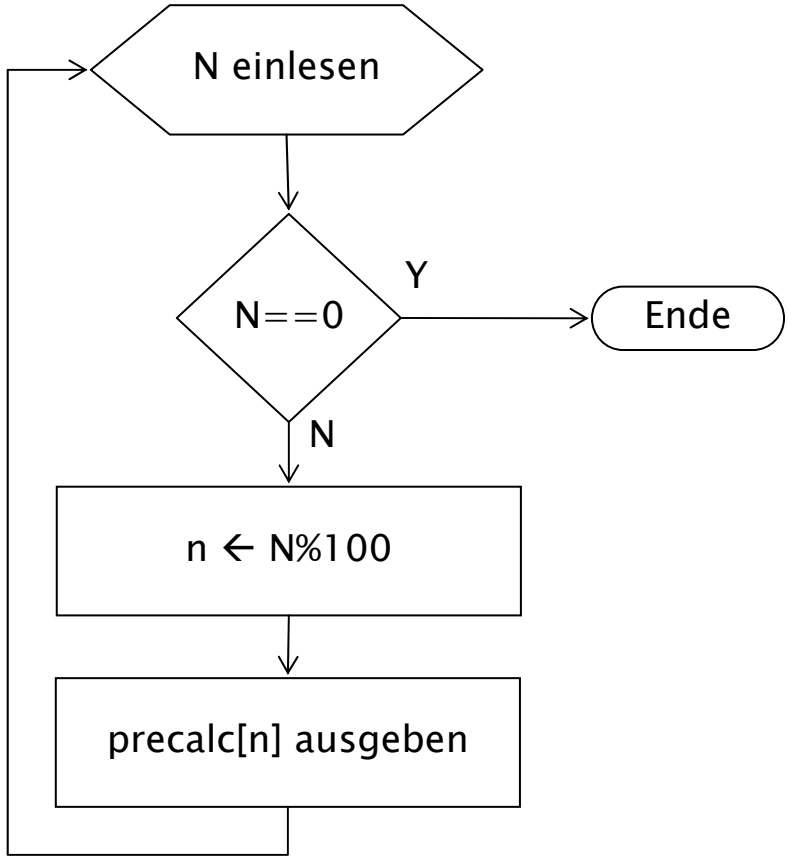
#10162 Last Digit

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN · FH MÜNCHEN

Lösungsansatz:

- Berechnung durch N Aufrufe von BigInteger.ModPow
- Besser:
  - $S_{100}=0$; $S_{200}=0$; $S_{300}=0$ ....
  - $S_{1xx}=S_{100}+S_{xx}$ ➔ $S_{1xx}=S_{xx}$
  - Es müssen nur die letzten zwei Ziffern beachtet werden (0-99)
  - zusätzlich Precalculation für 0-99

#10162 Last Digit




```
int[] precalc=
{0,1,5,2,8,3,9,2,8,7,7,8,4,7,3,8,4
,1,5,4,4,5,9,6,2,7,3,6,2,1,1,2,8,1
,7,2,8,5,9,8,8,9,3,0,6,1,7,0,6,5,5
,6,2,5,1,6,2,9,3,2,2,3,7,4,0,5,1,4
,0,9,9,0,6,9,5,0,6,3,7,6,6,7,1,8,4
,9,5,8,4,3,3,4,0,3,9,4,0,7,1,0};
```

HOCHSCHULE
FÜR ANGEWANDTE
WISSENSCHAFTEN · FH
MÜNCHEN

Fragen?

Danke für Eure Aufmerksamkeit