Grundbegriffe der Informatik

Tutorium 1 - 4. Sitzung

Dennis Felsing

dennis.felsing@student.kit.edu
http://www.stud.uni-karlsruhe.de/~ubcqr/2010w/tut_gbi/

2010-11-15

Überblick

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: a **div** $b = x$, $x \in$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a <$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a < (x+1) \cdot b$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a < (x+1) \cdot b$

mod

Der Operator **mod** liefert den Rest der Ganzzahldivision.
Es gilt: $0 \leq (x \,\mathbf{mod}\, y) <$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a < (x+1) \cdot b$

mod

Der Operator **mod** liefert den Rest der Ganzzahldivision.
Es gilt: $0 \leq (x \,\mathbf{mod}\, y) < y$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a < (x+1) \cdot b$

mod

Der Operator **mod** liefert den Rest der Ganzzahldivision.
Es gilt: $0 \leq (x \,\mathbf{mod}\, y) < y$
Und: $x = y \cdot (x \,\mathbf{div}\, y) +$

div und mod

div

Der Operator **div** stellt die Ganzzahldivision dar.
Es gilt: $a \,\mathbf{div}\, b = x,\ x \in \mathbb{Z},\ x \cdot b \leq a < (x+1) \cdot b$

mod

Der Operator **mod** liefert den Rest der Ganzzahldivision.
Es gilt: $0 \leq (x \,\mathbf{mod}\, y) < y$
Und: $x = y \cdot (x \,\mathbf{div}\, y) + (x \,\mathbf{mod}\, y)$

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4												

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0											

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0										

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0									

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0								

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1							

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1						

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1					

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1				

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2			

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2		

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$												

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0											

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0										

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0									

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0								

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4							

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4						

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4					

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4				

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8			

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8		

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4												

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0											

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1										

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2									

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3								

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0							

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1						

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2					

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3				

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0			

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1		

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y							

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2						

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5					

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25				

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0			

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2		

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y							

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1						

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0					

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6				

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6	999			

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6	999	0		

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6	999	0	2	

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6	999	0	2	6

Was sagt uns x **mod** 2 über x?

Aufgaben

x	0	1	2	3	4	5	6	7	8	9	10	11
x **div** 4	0	0	0	0	1	1	1	1	2	2	2	2
$4(x$ **div** $4)$	0	0	0	0	4	4	4	4	8	8	8	8
x **mod** 4	0	1	2	3	0	1	2	3	0	1	2	3

x	7	20	256	999	6	12	13
y	3	4	10	1000	3	5	7
x **div** y	2	5	25	0	2	2	1
x **mod** y	1	0	6	999	0	2	6

Was sagt uns x **mod** 2 über x? Wenn x **mod** $2 = 0$, dann ist x gerade. Wenn x **mod** $2 = 1$, dann ist x ungerade.

Informeller Algorithmusbegriff

Algorithmus als Beschreibung einer Folge von Anweisungen mit diesen Merkmalen:

Endliche Beschreibung: Kein unendlicher Text als Beschreibung

Elementare Anweisungen: Einfach ausführbar

Determinismus: Eindeutig welche Anweisung als nächstes

Endliche Eingabe: Arbeitet auf endlichem Wort

Endliche Ausgabe: Gibt endliches Wort aus

Terminierung: Endet nach endlich vielen Schritten

Beliebig große Eingaben: Keine Längenbeschränkung

Nachvollziehbarkeit: Fachleute verstehen Algorithmus

Beispiel

//Eingaben : $a, b \in \mathbb{N}_0$

$S_0 \leftarrow a$

$Y_0 \leftarrow b$

for $i \leftarrow 0$ **to** $b - 1$ **do**

$S_{i+1} \leftarrow S_i + 1$

$Y_{i+1} \leftarrow Y_i - 1$

od

Beispiel

//Eingaben : $a, b \in \mathbb{N}_0$
$S_0 \leftarrow a$
$Y_0 \leftarrow b$
for $i \leftarrow 0$ **to** $b - 1$ **do**
    $S_{i+1} \leftarrow S_i + 1$
    $Y_{i+1} \leftarrow Y_i - 1$
od

bzw.

//Eingaben : $a, b \in \mathbb{N}_0$
$S \leftarrow a$
$Y \leftarrow b$
for $i \leftarrow 0$ **to** $b - 1$ **do**
    $S \leftarrow S + 1$
    $Y \leftarrow Y - 1$
od

Schleifeninvariante

Definition

Eine Schleifeninvariante ist eine Aussage, die vor und nach jeder Schleifenausführung gilt.

Nützlich für Beweise. Dazu zeigt man: Die Schleifeninvariante gilt

- vor dem ersten Schleifendurchlauf
- nach jedem Schleifendurchlauf

$\Rightarrow$ Induktion!

Beispiel

//Eingaben : $a, b \in \mathbb{N}_0$

$S \leftarrow a$

$Y \leftarrow b$

for $i \leftarrow 0$ **to** $b - 1$ **do**

$S \leftarrow S + 1$

$Y \leftarrow Y - 1$

od

Was macht der Algorithmus?
Welche Schleifeninvariante hat der Algorithmus?

Beweisidee

Vermutung:

Schleifeninvariante

$S + Y = a + b$

- Wie widerlegt man sowas?

Beweisidee

Vermutung:

Schleifeninvariante

$S + Y = a + b$

- Wie widerlegt man sowas?
  Indem man zeigt, dass es für einen bestimmten Fall nicht gilt.

Beweisidee

Vermutung:

Schleifeninvariante

$S + Y = a + b$

- Wie widerlegt man sowas?
  Indem man zeigt, dass es für einen bestimmten Fall nicht gilt.
- Wie beweist man sowas?

Beweisidee

Vermutung:

Schleifeninvariante

$S + Y = a + b$

- Wie widerlegt man sowas?
  Indem man zeigt, dass es für einen bestimmten Fall nicht gilt.
- Wie beweist man sowas?
  Induktion!

Korrektheitsbeweis

Induktionsanfang: Vor der ersten Schleifenausführung:

$$S_0 = a, Y_0 = b \Rightarrow S_0 + Y_0 = a + b \checkmark$$

Induktionsvorraussetzung: Für ein beliebiges aber festes $i \in \mathbb{N}_0$ gelte: $S_i + Y_i = a + b$

Induktionsschluss: Wir zeigen, dass dann auch $S_{i+1} + Y_{i+1} = a + b$ gelten muss.

$$S_{i+1} + Y_{i+1} = S_i + 1 + Y_i - 1 = S_i + Y_i + 1 - 1 = S_i + Y_i$$

Nach IV: $= a + b$ $\square$

Korrektheitsbeweis

Induktionsanfang: Vor der ersten Schleifenausführung:

$$S_0 = a, Y_0 = b \Rightarrow S_0 + Y_0 = a + b \checkmark$$

Induktionsvorraussetzung: Für ein beliebiges aber festes $i \in \mathbb{N}_0$ gelte: $S_i + Y_i = a + b$

Induktionsschluss: Wir zeigen, dass dann auch $S_{i+1} + Y_{i+1} = a + b$ gelten muss.

$$S_{i+1} + Y_{i+1} = S_i + 1 + Y_i - 1 = S_i + Y_i + 1 - 1 = S_i + Y_i$$

Nach IV: $= a + b$ $\square$

Überlegung: Schleife wird b mal ausgeführt, Y_i wird jedes mal um eins verringert. Am Ende gilt aber immer noch $S_i + Y_i = a + b$, also ist $S_i = a + b$.

Aufgabe 3.2 von 2008

//Eingaben : $a, b \in \mathbb{N}_+$

$X_0 \leftarrow a$

$Y_0 \leftarrow b$

$P_0 \leftarrow 1$

$x_0 \leftarrow X_0$ **mod** 2

$n \leftarrow 1 + \lceil \log_2 a \rceil$

for $i \leftarrow 0$ **to** $n - 1$ **do**

$P_{i+1} \leftarrow P_i \cdot Y_i^{x_i}$

$X_{i+1} \leftarrow X_i$ **div** 2

$Y_{i+1} \leftarrow Y_i^2$

$x_{i+1} \leftarrow X_{i+1}$ **mod** 2

od

- Welchen Wert hat am Ende P_n?
- Gib eine sinnvolle Schleifeninvariante an.
- Beweise beides.

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0				

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1			

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4		

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1				

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1			

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2		

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2				

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1			

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1		

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3				

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16			

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0		

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4				

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4	16			

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4	16	0		

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4	16	0	256^2	

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4	16	0	256^2	0

Vermutung Schleifeninvariante

Aufgabe 3.2 von 2008

Was macht der Algorithmus mit Eingabe $a = 4, b = 2$?

i	P_i	X_i	Y_i	x_i
0	1	4	2	0
1	1	2	4	0
2	1	1	16	1
3	16	0	256	0
4	16	0	256^2	0

Vermutung Schleifeninvariante

$P \cdot Y^X = b^a$

Beweis Schleifeninvariante

Induktionsanfang: Vor der ersten Schleifenausführung:

$$P_0 = 1, Y_0 = b, X_0 = a \Rightarrow P_0 \cdot Y_0^{X_0} = b^a \checkmark$$

Induktionsvorraussetzung: Für ein beliebiges aber festes $i \in \mathbb{N}_0$ gelte: $P_i \cdot Y_i^{X_i} = b^a$

Induktionsschluss: Wir zeigen, dass dann auch $P_{i+1} \cdot Y_{i+1}^{X_{i+1}} = b^a$ gelten muss.

$$P_{i+1} \cdot Y_{i+1}^{X_{i+1}} = (P_i \cdot Y_i^{x_i}) \cdot Y_{i+1}^{X_{i+1}} =$$
$$(P_i \cdot Y_i^{x_i}) \cdot Y_i^{2(X_i \textbf{ div } 2)} = P_i \cdot Y_i^{x_i + 2(X_i \textbf{ div } 2)} =$$
$$P_i \cdot Y_i^{(X_i \textbf{ mod } 2) + 2(X_i \textbf{ div } 2)} = P_i \cdot Y_i^{X_i} = b^a \square$$

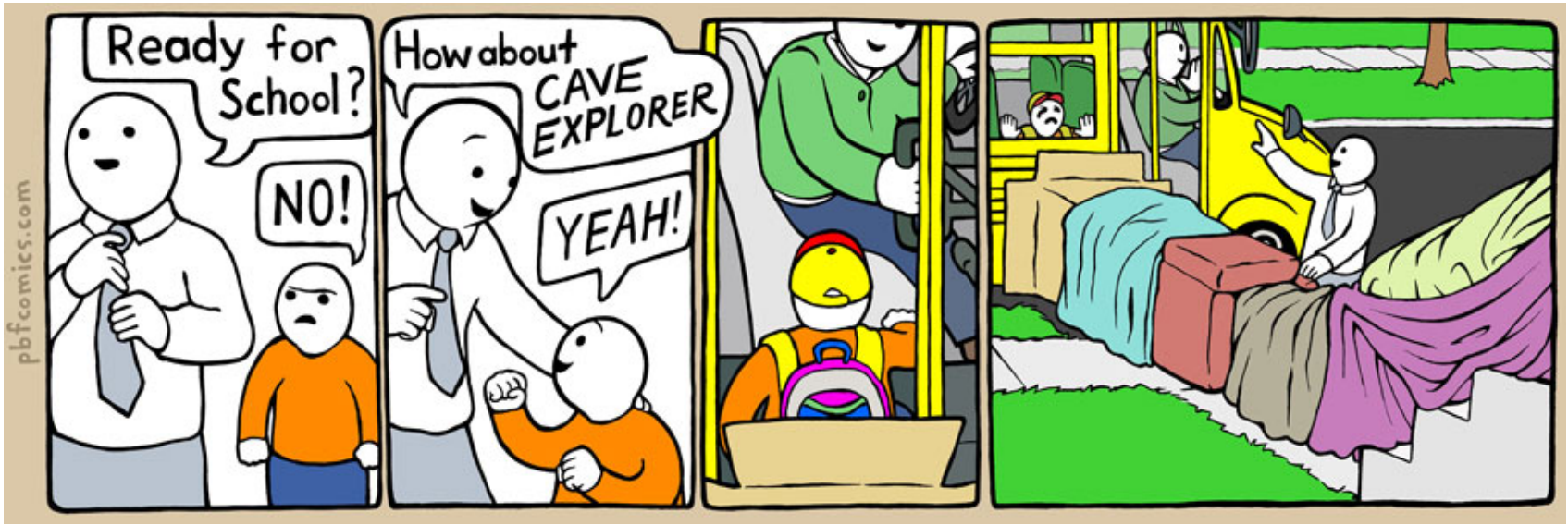
Wert von P_n

Jeder Schritt halbiert X_i. Daher gilt $\forall i \in \mathbb{N}_0 : X_i \leq a \,\mathbf{div}\, 2^i$.
Es gilt $2^{\lceil log_2 a \rceil} \geq a$. Also auch $X_{\lceil log_2 a \rceil} \leq a \,\mathbf{div}\, 2^{\lceil log_2 a \rceil} < 1$ und somit $X_{n-1} = X_{\lceil log_2 a \rceil} = 0$.
Da $0 \,\mathbf{div}\, 2 = 0$, gilt $X_n = X_{n-1}$.
Schleifeninvariante gilt auch nach n-tem Durchlauf.
Also ist $P_n \cdot Y_n^{X_n} = P_n \cdot Y_n^0 = P_n = b^a$ $\square$

Aufgabe 3.1 von 2008

Siehe `http://gbi08.ira.uka.de/uebung`

Überblick


Ready for School?
NO!
How about CAVE EXPLORER
YEAH!
pbfcomics.com