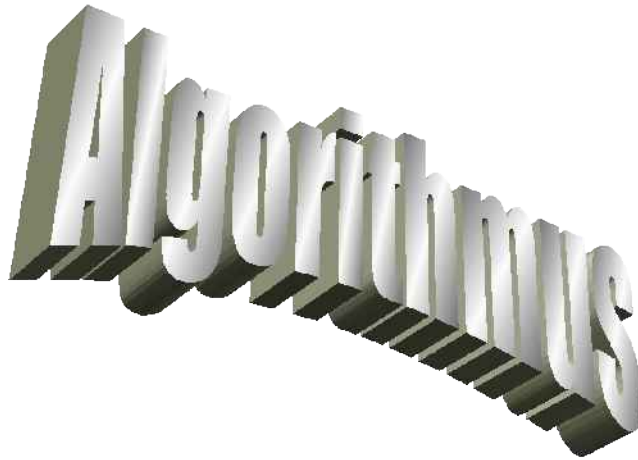


Algorithmus

Schreibe ein ...

1. A
2. l
3. g
4. o
5. r
6. i
7. t
8. h
9. m
10. u
11. s



... um das Wort „Algorithmus“ zu schreiben. Was könnte ein Algorithmus sein?

Ein **Algorithmus** ist eine endliche Folge von eindeutig bestimmten Elementaranweisungen, die den Lösungsweg eines Problems exakt und vollständig beschreiben. Umgangssprachlich könnte man sagen, dass **Algorithmen** Handlungsanweisungen sind.

Algorithmen beschränken sich nicht auf das Rechnen mit Zahlen, es gibt auch zahlreiche Beispiele aus dem Alltag. Im Folgenden werden nur einige Beispiele erwähnt:

- Die in einem Kochrezept auf einander folgenden Handlungsanweisungen zur Zubereitung einer Mahlzeit bilden in ihrer Gesamtheit einen Algorithmus.
- Reparatur- und Bedienungsanleitungen stellen Algorithmen dar.
- Hilfen zum Ausfüllen von Formularen sind ebenfalls Algorithmen.
- Bei der Softwareentwicklung schreiben die Programmierer Algorithmen, um dem Computer Handlungsanweisungen zu geben.
- Die Beschreibung eines exakten Bewegungsablaufes beim Turnen stellt einen Algorithmus dar.

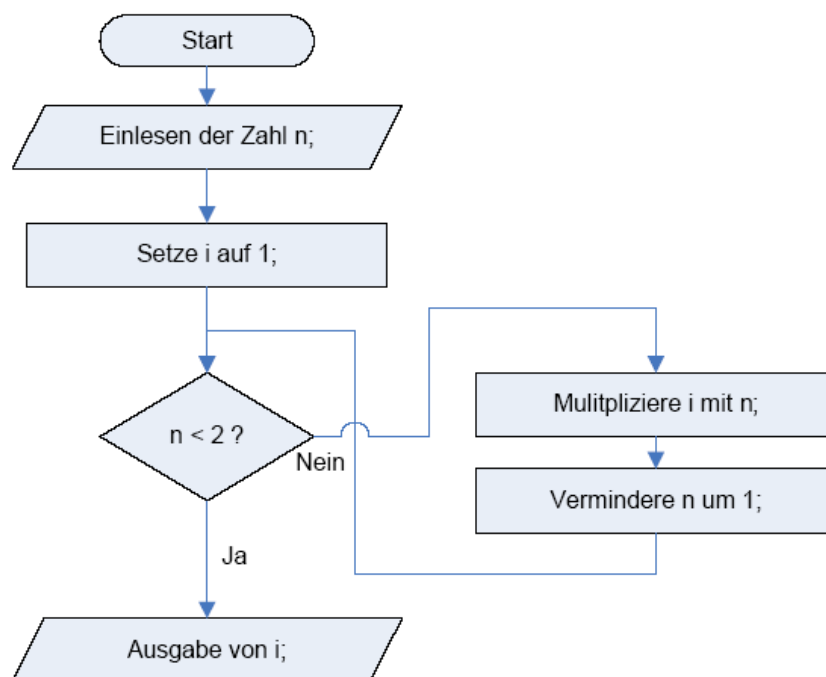
Nun sollt ihr noch die wichtigsten Eigenschaften von Algorithmen kennen lernen: Algorithmen gibt es in mehr oder weniger präziser Form seit es menschliche Sprache gibt. Umso unpräziser die Handlungsanweisungen sind, umso komplexer und länger wird der Algorithmus. Dies hat zur Folge, dass du oder der Computer

wesentlich mehr „Rechenzeit“ und „Speicherplatz“ benötigt, bis die Lösung erreicht wird (Komplexität von Algorithmen). Eine weitere Eigenschaft ist die Endlichkeit von Algorithmen, d.h. das Ergebnis wird nach der Ausführung von endlich vielen Handlungsanweisungen erreicht.

Algorithmen können in unterschiedlichster Art und Weise dargestellt werden. Bei der Softwareentwicklung verwendet man normierte Darstellungsmittel, wie zum Beispiel einen Programmablaufplan, welchen ihr hier als Grafiken auf dem Arbeitsblatt angedeutet seht (aber auch Pseudocodes und Syntaxdiagramme). Für uns genügt aber vorerst die Textbeschreibung als Mittel zur Darstellung von Algorithmen, da dies schon eine komplexe Aufgabe darstellen kann. Vielleicht könnt ihr aber trotzdem erahnen, was der abgebildete Algorithmus in der Grafik „bewirkt“ (siehe Aufgabe 2).

Beim Umgang mit Algorithmen stößt man auf folgende Aufgabenfelder:

- **Abarbeitung** von fertigen, vorgegebenen Algorithmen → oft langweilig → längere Aufgaben löst der PC schneller und zuverlässiger als der Mensch (s. Aufgabe 1)
- **Verstehen, Nachvollziehen** und **Analysieren** vorgegebener & insbesondere klassischer Algorithmen (siehe Aufgabe 2),
- **Entwurf & Konstruktion** von (neuen) Algorithmen → höchst kreative Aktivität! (siehe Aufgabe 3)



Algorithmen in der Mathematik → Aufgaben

1. Arbeite den folgenden Algorithmus zur Ermittlung der Primzahlen bis 100 ab:
(Sieb des Eratosthenes)

- Streiche zuerst die 1 weg, da es sich bei 1 um keine Primzahl handelt.
- Es folgt die 2. 2 wurde bis jetzt nicht weg gestrichen und ist deshalb Primzahl. Wir markieren 2 als Primzahl.
- Wir streichen nun alle durch 2 teilbaren Zahlen, weil diese nicht Primzahlen sein können. (Sie hätten jeweils die Teiler 1,2 und sich selbst)
- Die 3 ist nun die nächste ungestrichene Zahl! Wir markieren 3 als Primzahl.
- Wir streichen nun alle durch 3 teilbaren Zahlen, weil diese ebenfalls keine Primzahlen mehr sein können.
- Nun wiederholen wir Schritt d) und e) bis alle Zahlen entweder als Primzahlen markiert, bzw. als Nichtprimzahlen durchgestrichen sind. **Achtung:** Diese Schritte müssen nur bis zur Zahl durchgeführt werden, die größer oder gleich der Wurzel des zu überprüfenden Bereich ist. z.B. 10 bei 100, 32 bei 1000, da $10 * 10 = 100$ bzw. $32 * 32 = 1024 > 1000$

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

Aufgabe 2:

Welche Funktion hat der abgebildete Programmablaufplan?

Aufgabe 3:

Erstelle einen eigenen Algorithmus zu einem beliebigen Problem der Mathematik in Zusammenarbeit mit deinem Nachbarn. Achte hierbei auf die Komplexität von Algorithmen und versuche kurze und einfache Sätze zu bilden!

Beispiele: schriftliches Addieren, Multiplizieren, Dividieren, ...

Literatur:

- <http://www.wikipedia.de> (Stand: 16.05.2006)
- <http://www.primzahlen.de/files/theorie/sieb.htm> (Stand: 16.05.2006)
- Uwe Schneider, Dieter Werner: Taschenbuch der Informatik, München, Wien 2004, Auflage 5.