Fliesskommazahlen für Schüler der Gymnasialstufe

amaximov

Juni 2020

Inhaltsverzeichnis

1 Einführung

Wie können Computer so viele unterschiedliche Dinge mit nur Nullen und Einser darstellen? Auf dem Bildschirm sehen wir Texte, Bilder, Videos, die wir noch dazu verändern können.

Vorletzte Woche haben wir gesehen, wie Computer ganze Zahlen speichern und manipulieren. Sie stellen die Zahlen in Basis 2 dar und speichern eine feste Anzahl Bits.

Aufgabe 1.1. *Schreibe folgende Zahlen in der 32-Bit Zweierkomplement Darstellung:*

(a) 5

(b) -1

(c) 25

(d) -25

Letzte Woche haben wir angefangen, uns mit der Darstellung der reellen Zahlen zu beschäftigen, mit den **Fliesskommazahlen**. Heute machen wir damit weiter.

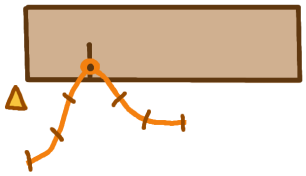
Die Fliesskommazahlen, wie wir gesehen haben, sind im Grunde genommen nichts anderes, als die **Exponentialschreibweise**, die wir schon aus Chemie und Physik kennen: Die **Mantisse** wird mit der Basis hoch einem **Exponenten** multipliziert.

Zum Beispiel, die Avogadro-Konstante schreiben die Chemiker als $6.02214076 \cdot 10^{23}$, anstatt alle 24 Stellen anzugeben. In dieser Schreibweise lassen sich Zahlen der unterschiedlichsten Grössenordnungen kompakt und übersichtlich darstellen:

- Masse von einem Proton: $1.673 \cdot 10^{-27}$ Kilogramm
- Grösse eines Coronavirus: $1,4 \cdot 10^{-7}$ Meter
- Lichtgeschwindigkeit: $2.998 \cdot 10^{8}$ Meter pro Sekunde
- Durchmesser von der Andromedagalaxie: $1.32 \cdot 10^{21}$ Meter

Im Unterschied zu Menschen, arbeitet der Computer meistens in der Basis 2 statt 10 und **schränkt** die Anzahl der möglichen signifikanten Stellen und Exponenten **ein**.

Um diese Einschränkungen sichtbar und anfassbar zu machen, hatten wir das **"Kasten-Seil"-Modell** eingeführt.

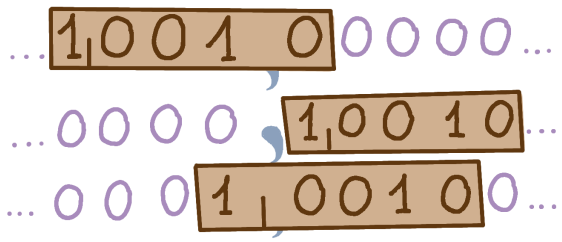


Das Modell besteht aus einem **Kasten**, in welchem wir eine fixe Anzahl Ziffern speichern können, und einem **Seil**, welches die Position vom Kasten bezüglich dem Komma speichert. Die Mitte vom Seil ist genau nach der ersten Stelle im Kasten befestigt und zwei Enden hängen lose: Das **negative Ende** und das **positive Ende**. Auf dem Seil wird **markiert**, wo sich das Komma befindet.

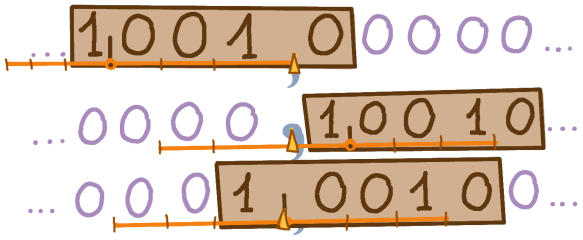
Was machen wir, wenn wir folgende reelle Zahlen, binär in einer Fixkommadarstellung dargestellt, im "Kasten-und-Seil"-Modell umschreiben wollen?

$$\ldots 1001{,}00000\ldots$$
$$\ldots 0000{,}1010\ldots$$
$$\ldots 0001{,}00100\ldots$$

Erstens, müssen wir bestimmen, wo wir den Kasten hinstellen, d.h. welche Ziffern gespeichert werden. Alle Ziffern, die sich ausserhalb vom Kasten befinden, gehen verloren. Es macht Sinn, die Ziffern mit dem grössten Wert zu nehmen, also mit der linkesten Eins anzufangen. Die Bits im Kasten sind die **Mantisse**.



Nun haben wir 3 Kasten mit genau dem gleichen Inhalt: `10010`. Um die ursprünglichen Zahlen rekonstruieren zu können, müssen wir uns auch die Position des Kastens bezüglich dem Komma merken. Das machen wir im zweiten Schritt mit dem Seil. Die Mitte vom Seil wird gleich nach der linkesten Ziffer im Kasten befestigt, und die Position vom Komma wird am Seil markiert. Die Markierung auf dem Seil modelliert den **Exponenten**.



Jetzt haben wir alle Informationen gespeichert, die wir brauchen, um die ursprünglichen Zahlen wiederherzustellen. Damit diese Darstellung eindeutig ist, verlangen wir, dass das erste Bit der Mantisse eine Eins ist.

Folgende Elemente charakterisieren ein Fliesskommazahlensystem: Die Grösse vom Kasten, d.h. die **Mantissenlänge**, und die Länge vom Seil, d.h. der **Exponentenbereich**. Die Bits im Kasten und die Markierung am Seil stellen eine Zahl dar.

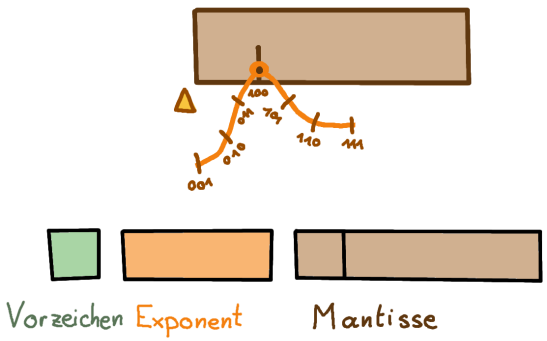
Der Computer hat intern keine Kasten und keine Seile. Er arbeitet mit Bitmuster. Jede Zahl hat eine fixe Anzahl Bits, typischerweise 32 (float) oder 64 (double), und diese Bits werden in 3 Bereiche aufgeteilt:

- Vorzeichen (grün auf dem Bild),
- Exponent (Orange auf dem Bild),
- Mantisse (braun auf dem Bild).

Im Vorzeichenteil wird das **Vorzeichen** kodiert: `0` für positive Zahlen und `1` für negative.

Im Exponententeil wird die Markierung am Seil kodiert. Damit positive und negative Exponente in der fixen Anzahl Bits kodiert werden können, ein **Biaswert** wird zum Exponenten addiert, so dass die 0 mit `1000...0` kodiert wird.

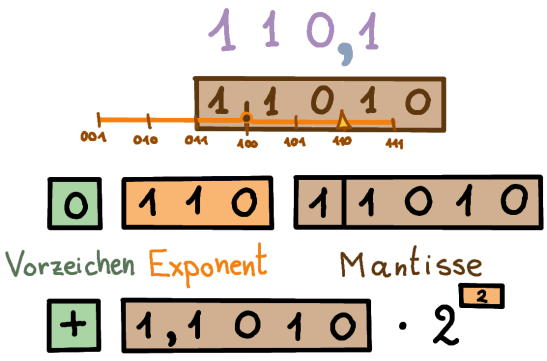
Im Mantissenteil werden die Bits aus dem Kasten gespeichert. Da das erste Bit immer eine Eins ist, wird es im Computer nicht gespeichert (implizites oder verstecktes Bit). Hier wird deshalb die führende Eins immer in Klammern angeführt.



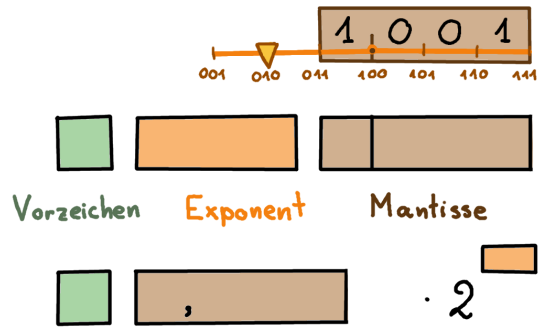
Beispiel 1.1. *Wir werden zusammen die Zahl* 6.5 *im Fliesskommazahlensystem mit Mantissenlänge* 5 *und Exponenten von* -3 *bis* 3 *darstellen.*

Die reelle Zahl in Basis 2 in einer Fixkommadarstellung ist 110.1.

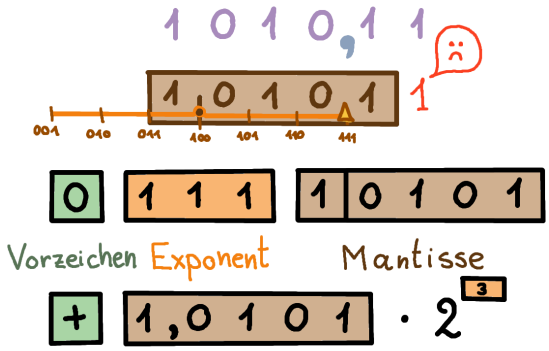
Der Kasten hat 5 Plätze. Alle signifikanten Stellen haben dort Platz. Dann verbinden wir das Seil mit dem Komma und setzen eine Markierung. Das lässt sich direkt ins Bitmuster übersetzen: Das Vorzeichen ist positiv, die Kodierung vom Exponenten lässt sich am Seil ablesen, die Mantisse speichert man direkt.



Aufgabe 1.2. *Welche Zahl ist unten dargestellt? Vervollständige das Bitmuster und die Exponentialschreibweise und schreibe den Dezimalwert auf.*



Beispiel 1.2. *Nicht alle Zahlen lassen sich im Fliesskommazahlensystem genau darstellen. Manche müssen gerundet werden. Zum Beispiel, die Zahl* 10.75 *sieht in Binär in einer Fixkommadarstellung so aus:* 1010.11*. Sie hat* 6 *signifikante Stellen, aber nur* 5 *haben in der Mantisse Platz. Die letzte Eins kann nicht gespeichert werden und geht verloren.*



2 Fliesskommazahlen

2.1 Kleinste und grösste positive Zahlen

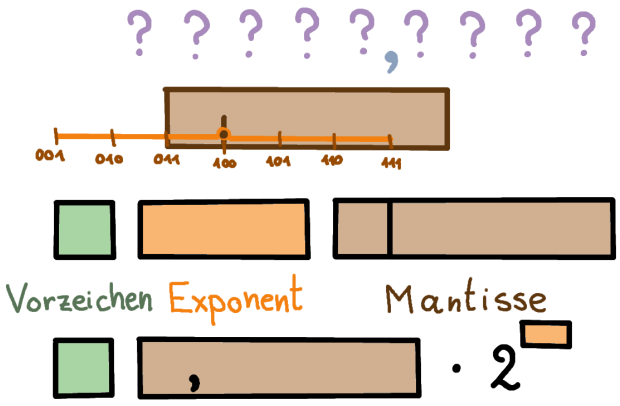
Es gibt unendlich viele reelle Zahlen. Wir können aber nur endlich viele davon in einem Fliesskommazahlensystem darstellen. Im Kasten, welcher uns die Mantisse veranschaulicht, finden nur endlich viele Bits Platz. Das Seil, welches den Kasten an den Komma bindet und welches uns den Exponenten veranschaulicht, hat ebenfalls eine endliche Länge.

Weil es endlich viele darstellbare Zahlen gibt, muss es eine kleinste und eine grösste Zahl geben. In diesem Abschnitt werden wir die kleinste und die grösste positive darstellbare Zahlen in Abhängigkeit vom Exponentenbereich und Mantissenlänge finden.

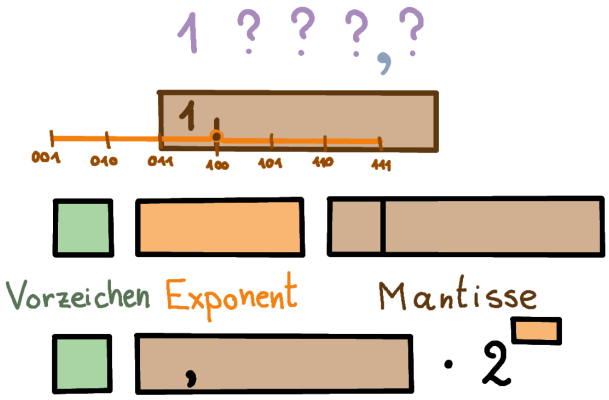
Beispiel 2.1. *Wir konstruieren die grösste positive Zahl im Fliesskommazahlensystem mit Mantissenlänge* 5 *und Exponenten von* -3 *bis* 3*.*

In violett wird die reelle Zahl in Basis 2 aufgeschrieben, in der zweiten Zeile kommt die "Kasten-und-Seil"-Darstellung aus der Einführung, in der dritten Zeile das Bitmuster und als letztes die binäre Exponentialschreibweise. Darstellungsübergreifend ist die Mantisse in braun markiert, der Exponent in Orange und das Vorzeichen in grün.

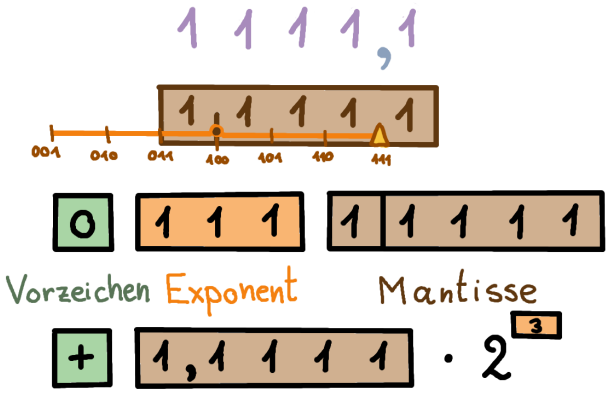
Als erstes platzieren wir den Kasten. Damit die Zahl so gross wie möglich wird, muss der Kasten links vom Komma stehen, und zwar so weit entfernt wie möglich. Wir haben aber eine Einschränkung: Das Seil muss immer mit dem Komma verbunden bleiben.



Der Exponent muss also möglichst gross sein.
Was ist mit der Mantisse? Sicher muss eine Eins an der ersten Stelle stehen.



Damit die Mantisse möglichst gross wird, muss sie aus lauter Einser bestehen.



Die grösste positive darstellbare Zahl in diesem Fliesskommazahlensystem ist also 15.5.

Aufgabe 2.1. *Konstruiere die kleinste positive Zahl im Fliesskommazahlensystem mit Mantissenlänge* 5 *und Exponenten von* -3 *bis* 3. *Schreibe die Zahl im "Kasten-und-Seil"-Modell, als Bitmuster und in der binären Exponentialschreibweise auf und berechne ihren Dezimalwert.*

Wir haben gesehen, dass die positive Zahlen, welche sich exakt in einem Fliesskommazahlensystem mit Mantissenlänge 5 und 3 Bits für den Exponenten darstellen lassen, zwischen 1/8 und 15.5 liegen müssen. Wie stark verändern sich diese Werte, wenn wir ein Bit weniger für die Mantisse zur Verfügung stellen?

Aufgabe 2.2. *Betrachte das Fliesskommazahlensystem mit Mantissenlänge* 4 *und Exponenten von* -3 *bis* 3. *Was erwartest du für die grösste positive darstellbare Zahl? Ist sie kleiner oder grösser als* 15.5*? Wie stark unterscheidet sie sich davon?*

Und was erwartest du für die kleinste positive darstellbare Zahl? Ist sie grösser oder kleiner als $1/8$*? Wie stark unterscheidet sie sich davon?*

Konstruiere die grösste und die kleinste positive darstellbare Zahlen in diesem System. Schreibe die Zahlen im "Kasten-und-Seil"-Modell, als Bitmuster und in der binären Exponentialschreibweise auf und berechne den Dezimalwert.

In der vorherigen Aufgabe hast du gesehen, welchen Einfluss die Mantissenlänge auf die grösste und kleinste positive darstellbare Zahlen hat. Das Ergebnis könnte überraschend kommen. Da wir ein Bit weniger für die Mantisse und gleich viele Bits für die Exponentenkodierung genommen haben, ist unser Bitmuster um ein Bit kürzer geworden, und somit können wir höchstens halb so viele Zahlen darstellen als vorher. Trotzdem haben sich die grösste und die kleinste positive darstellbare Zahlen kaum verändert.

Was passiert, wenn wir nun ein Bit weniger für die Exponentenkodierung nehmen?

Aufgabe 2.3. *Betrachte das Fliesskommazahlensystem mit Mantissenlänge* 5 *und mit nur* 2 *Bits, um den Exponenten zu kodieren.*

(a) Welche Exponenten können wir mit 2 *Bits darstellen? Wie lang wird das Seil? Zeichne das "Kasten-und-Seil"-Modell für dieses Fliesskommazahlensystem.*

(b) Was erwartest du für die grösste und die kleinste positive darstellbaren Zahlen? Wie stark unterscheiden sie sich von 15.5 *und* $1/8$*?*

(c) Konstruiere die grösste und die kleinste positive darstellbare Zahlen in diesem System. Schreibe die Zahlen im "Kasten-und-Seil"-Modell, als Bitmuster und in der binären Exponentialschreibweise auf und berechne den Dezimalwert.

In der vorherigen Aufgabe hast du gesehen, welchen Einfluss die Länge der Exponentenkodierung auf die grösste und kleinste positive darstellbare Zahlen hat. Wie auch im Fliesskommazahlensystem mit Mantissenlänge 4 und 3 Bits für den Exponenten, können wir im Fliesskommazahlensystem aus Aufgabe 2.3 höchstens halb so viele Zahlen darstellen wie im Fliesskommazahlensystem mit Mantissenlänge 5 und 3 Bits für die Exponentenkodierung. Der Bereich der darstellbaren Zahlen hat sich dieses Mal aber extrem verändert.

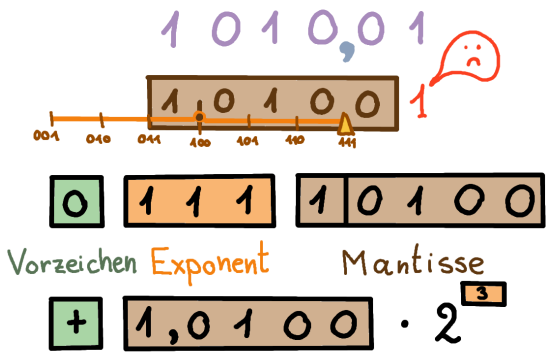
Aufgabe 2.4. *Wie sehen die kleinste und die grösste positive darstellbare Zahlen im allgemeinen Fliesskommazahlensystem aus? Nehme Mantissenlänge* m *und Exponent zwischen* e_{min} *und* e_{max} *an. Schreibe die kleinste und die grösste positive darstellbare Zahl in diesem Fliesskommazahlensystem als Bitmuster und in der binären Exponentialdarstellung auf.*

2.2 Darstellbare Zahlen

Wir wissen, dass es eine kleinste und eine grösste positive Zahl gibt, welche sich exakt in einem Fliesskommazahlensystem darstellen lassen. Dass man nicht alle unendlich viele reelle Zahlen zwischen diesen zwei Schranken darstellen kann, können wir uns denken. Die Frage ist nun, welche Zahlen sich darstellen lassen und wie sich der Abstand zwischen darstellbaren Zahlen verhält.

Hier und in den folgenden Kapiteln, falls nicht speziell vermerkt, werden wir mit Mantissenlänge 5 und Exponentenbereich von -3 bis 3 arbeiten.

Beispiel 2.2. *Nehmen wir eine Zahl zwischen 1/8 und 15.5 (die kleinste und grösste positive darstellbare Zahlen in diesem Fliesskommazahlensystem), zum Beispiel* 10.25*. Lässt sich diese Zahl darstellen?*



Diese Zahl lässt sich in gegebenem System nicht exakt darstellen. Für die letzte 1 *gibt es in der Mantisse kein Platz. Deswegen wird* 10.25 *mit* 10.0 ***approximiert****.*

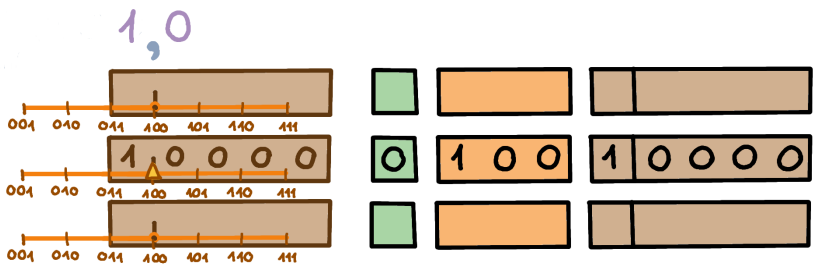
Wir haben gesehen, dass reelle Zahlen sich nur dann exakt darstellen lassen, wenn alle signifikante Stellen in der Mantisse Platz haben.

Beispiel 2.3. *Betrachten wir die Zahl* 1*.*

In einem Fliesskommazahlensystem lassen sich nur endlich viele Zahlen exakt darstellen. Einige von diesen darstellbaren Zahlen sind grösser als 1*. Da es nur endlich davon gibt, muss es darunter eine kleinste geben. Diese Zahl bezeichnen wir hier als "nächste" oder "nächstgrösste". Die "vorherige" oder "nächstkleinste" darstellbare Zahl ist entsprechend die grösste unter den darstellbaren Zahlen, die kleiner als* 1 *sind.*

Wie sieht die nächstgrösste Nachbarzahl von 1 *aus?*

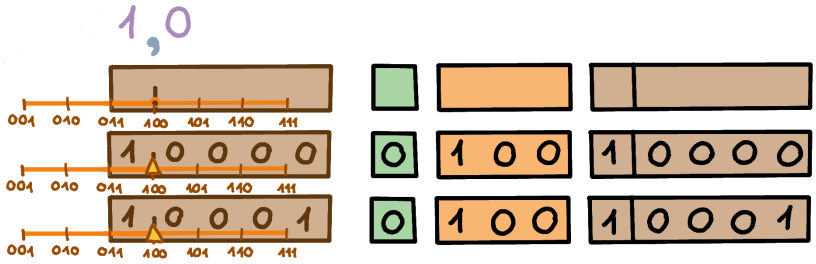
Im ersten Schritt stellen wir die Zahl 1 *im gegebenem Fliesskommazahlensystem dar.*



Wir suchen die Zahl, die minimal grösser als 1 *ist.*

Den Exponenten können wir in diesem Fall nicht verändern: Wenn wir den Kasten nach rechts bewegen, dann wird die Zahl kleiner; wenn wir den Kasten nach links bewegen, dann wird die Zahl wegen der obligatorischen führenden Eins zu gross.

Also müssen wir die Mantisse verändern. Da wir eine grössere Zahl suchen, müssen wir eine der Nullen zu einer Eins machen. Welche? Diejenige mit dem kleinsten Wert, d.h. die letzte Eins rechts.



Die nächste darstellbare Zahl ist also $1 + 1/16 = 17/16$.

Aufgabe 2.5. *Betrachte die Zahl* 1. *Finde die nächstkleinste darstellbare Zahl. Ist der Abstand zwischen der nächstkleinsten Zahl und* 1 *gleich dem Abstand zwischen* 1 *und der nächstgrössten Zahl?*

Aufgabe 2.6. *Finde die nächste und die vorherige darstellbare Zahlen von folgenden Zahlen. Schreibe die Werte in der Dezimaldarstellung auf und stelle alle Zahlen als Bitmuster und in der binären Exponentialschreibweise dar. Bilde alle Zahlen, die in der Aufgabe und in deiner Lösung vorkommen, auf einem Zahlenstrahl ab. Was beobachtest du? Sind alle Nachbarn gleich entfernt?*

(a) 2

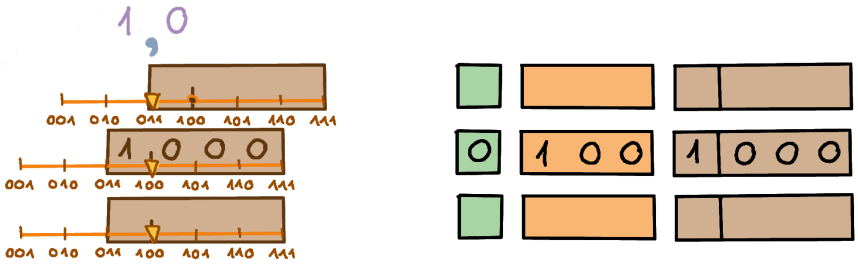
(b) 3

(c) 4

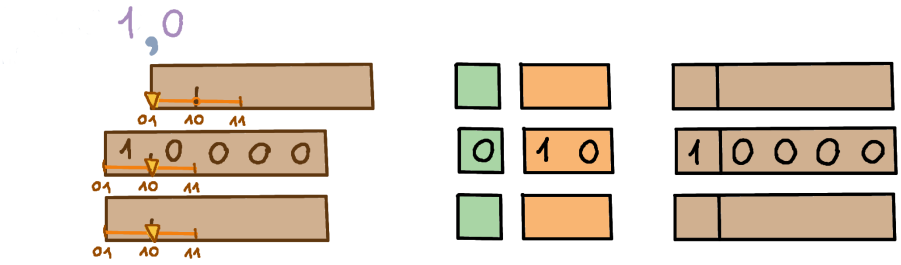
Im Abschnitt 2.1 haben wir gesehen, dass die Länge der Exponentenkodierung deutlich mehr Einfluss auf dem darstellbaren Zahlenbereich hat als die Mantissenlänge. Ist das so auch für den Abstand zwischen Nachbarzahlen?

Aufgabe 2.7. *Betrachte die Zahl* 1.

(a) Fülle die vorherige und die nächste darstellbare Zahlen im Fliesskommazahlensystem mit Mantissenlänge 4 *und Exponent zwischen* -3 *und* 3 *aus.*



(b) Fülle die vorherige und die nächste darstellbare Zahlen im Fliesskommazahlensystem mit Mantissenlänge 5 *und Exponent zwischen* -1 *und* 1 *aus.*



(c) Bei welchem Fliesskommazahlensystem ist der Abstand zwischen benachbarten Zahlen ähnlich zu dem im Fliesskommazahlensystem mit Mantissenlänge 5 *und Exponent zwischen* -3 *und* 3*?*

Teste dich selber

Aufgabe 2.8. *Beantworte folgende Fragen:*

(a) Kann man im Fliesskommazahlensystem alle reelle Zahlen darstellen? Wieso?

(b) Gibt es eine grösste Zahl im Fliesskommazahlensystem? Falls nein, warum? Falls ja, wie findet man sie?

(c) Gibt es eine kleinste Zahl im Fliesskommazahlensystem? Falls nein, warum? Falls ja, wie findet man sie?

(d) *Was beeinflusst stärker den Bereich der positiven darstellbaren Zahlen in einem Fliesskommasystem? Die Mantissenlänge oder die Länge der Exponentenkodierung?*

(e) *Gib eine Zahl zwischen* $1/2$ *und* 3.5 *an, die im Fliesskommazahlensystem mit Mantissenlänge* 3 *und Exponenten von* -1 *bis* 1 *nicht darstellbar ist.*

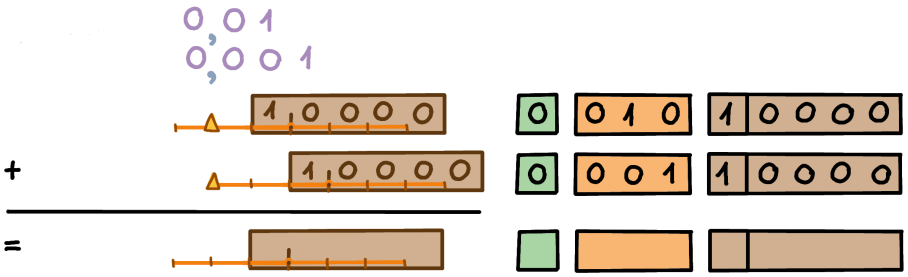
(f) *Sind alle Zahlen im Fliesskommazahlensystem gleichverteilt? Falls nicht, welche Zahlen stehen dichter beieinander, die kleineren oder die grösseren?*

(g) *Was beeinflusst stärker den Abstand zwischen den positiven darstellbaren Zahlen in einem Fliesskommazahlensystem? Die Mantissenlänge oder die Länge der Exponentenkodierung?*

3 Addition

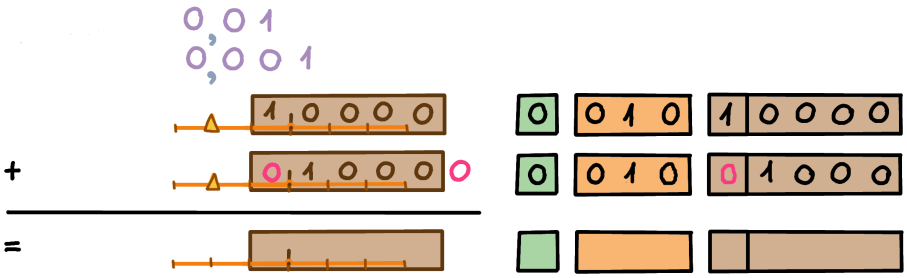
Im vorherigen Kapitel haben wir gesehen, welche Zahlen in einem Fliesskommazahlensystem dargestellt werden können, das heisst welche Zahlen exakt in einem Computer gespeichert werden können. Alle andere Zahlen werden gerundet. Computer werden aber nicht nur zum Speichern von Zahlen verwendet, sondern auch für Berechnungen. Auch bei Berechnungen verhalten sich Fliesskommazahlen nicht ganz wie reelle Zahlen. In diesem Kapitel werden wir dies am Beispiel der Addition erfahren.

Beispiel 3.1. *Wir möchten* $1/4 + 1/8$ *ausrechnen. Der erste Schritt ist beide Zahlen als Fliesskommazahlen aufzuschreiben. Wie in den vorherigen Kapiteln, sind in violett die reelle Zahlen in Basis 2 aufgeschrieben und braune "Kasten" mit orangenem "Seil" verwendet, um Mantisse und Exponent zu veranschaulichen. Rechts wird das Bitmuster in der gewöhnlichen Form angegeben.*

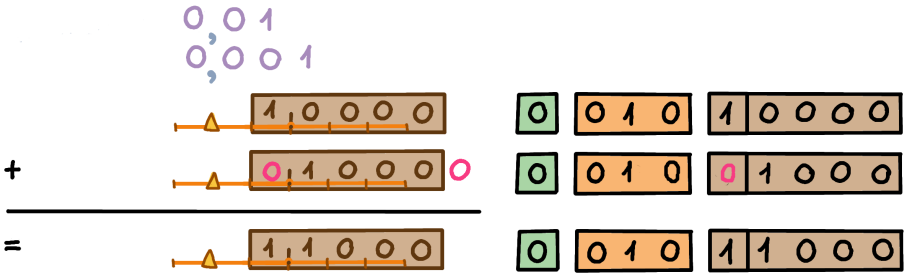


Damit wir die Bits der Mantisse stellenweise addieren können, wie wir das von den ganzen Zahlen kennen, müssen wir die zwei "Kasten" so verschieben, dass sie sich untereinander befinden. Da alles, was ausserhalb vom "Kasten" landet, verloren geht, werden wir den Kasten von der kleineren Zahl unter den Kasten von der grösseren Zahl schieben. So werden wir die Stellen mit dem niedrigsten Wert verlieren. In diesem Fall verlieren wir eine Null, der Wert der Zahl verändert sich also nicht.

Beachte, dass wenn der Kasten verschoben wird, verschiebt sich auch die Markierung bezüglich des "Seils", das heisst der Exponent verändert sich. Die Markierung am "Seil" bleibt immer unter dem Komma.

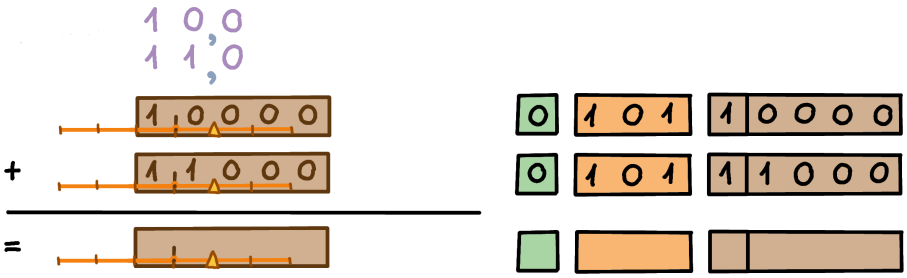


Wenn die Kasten untereinander sind, können wir die Bits in den Kasten wie gewöhnlich addieren, wie bei den Integers.

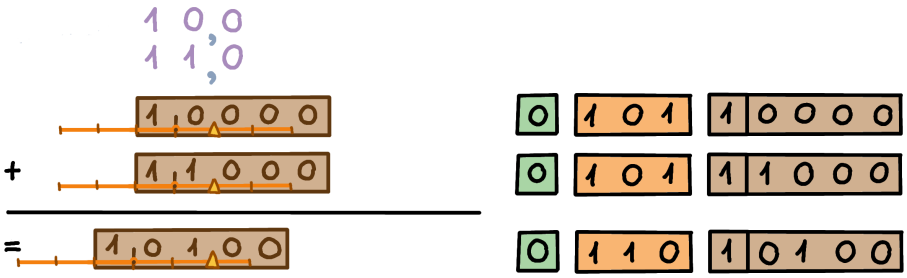


Wir haben ausgerechnet, dass $1/4 + 1/8 = 3/8$*, in der Exponentialschreibweise* $1.1000 \cdot 2^{-2}$*.*

Beispiel 3.2. *Wir möchten* $2 + 3$ *ausrechnen. Im ersten Schritt schreiben wir beide Zahlen auf.*



Die Kasten befinden sich schon untereinander. Wir müssen also nichts verschieben und können sofort losrechnen.



Beachte, dass der Kasten vom Ergebnis bezüglich den Kasten der Summanden verschoben ist, um die neue signifikante Stelle zu enthalten.

Wir haben ausgerechnet, dass $2 + 3 = 5$*, in der Exponentialschreibweise* $1.0100 \cdot 2^2$*.*

Aufgabe 3.1. *Rechne folgende Summen aus. Die Mantissenlänge beträgt* 5 *Bits, der Exponent geht von* -3 *bis* 3*. Gebe bitte das Bitmuster und die Exponentialdarstellung des Resultats an.*

(a) $5/8 + 3/4$

(b) $10 + 2.25$

(c) $17/16 + 2$

Die Addition im Fliesskommazahlensystem ist wie gewöhnlich kommutativ, weil wir immer die kleinste Zahl so verschieben, dass ihr Kasten unter dem Kasten der grösseren Zahl steht und dann die Bits in beiden Kasten stellenweise zusammen addieren.

Aufgabe 3.2. *Berechne dazu zwei Mal die gleiche Summe in einem Fliesskommazahlensystem mit Mantissenlänge* 5 *und Exponenten von* -3 *bis* 3*: Das erste Mal als* $1/8 + 2/8 + 3/8 + 4/8 + 5/8 + 6/8 + 7/8 + 8/8$ *und das zweite Mal als* $8/8 + 7/8 + 6/8 + 5/8 + 4/8 + 3/8 + 2/8 + 1/8$*.*

Welchen Resultat erwartest du? Sind die zwei Summen gleich oder unterschiedlich? Nimm dir Zeit und rechne die zwei Summen tatsächlich aus.

Die zwei Summen, die du ausgerechnet hast, liefern unterschiedliche Ergebnisse. Die erste liefert den exakten Wert 4.5, während bei der zweiten Summe kriegen wir im Fliesskommazahlensystem nur 4.25, und das obwohl der exakte Wert dargestellt werden kann. Das passiert, weil man bei den Fliesskommazahlen nur Zahlen der ähnlichen Grössenordnung exakt addieren kann. In der ersten Summe addieren wir die kleineren Summanden am Anfang, wenn die kumulative Summe noch nicht zu gross ist. In der zweiten Summe wächst die kumulative Summe sehr schnell, und irgendwann sind die Summanden zu klein bezüglich der kumulativen Summe, um einen Unterschied zu machen.

Daraus können wir folgern, dass die Addition bei den Fliesskommazahlen nicht assoziativ ist.

Aufgabe 3.3. *Betrachten wir die Summe* $1/8 + 1/8 + 1/8 + \cdots + 1/8$. *Bei den reellen Zahlen können wir mit solchen Summen auf beliebig grossen Zahlen kommen. Bei den Fliesskommazahlen kann das nicht gehen, weil, wie wir im vorherigen Kapitel gesehen haben, es eine grösste Fliesskommazahl gibt. Aber können wir diese Zahl auch tatsächlich erreichen?*

In einem Fliesskommazahlensystem mit Mantissenlänge 5 *und Exponentenbereich von* -3 *bis* 3*, was ist die grösste Zahl, die wir erreichen können, wenn wir beliebig viele* $1/8$ *zusammen rechnen? Wie viele Summanden brauchen wir, um diese Zahl zu erreichen?*

Teste dich selber

Aufgabe 3.4. *Beantworte folgende Fragen:*

(a) Warum kann man im Allgemeinen zwei Mantissen nicht stellenweise zusammen addieren?

(b) Gregory behauptet, dass der Kasten vom Ergebnis sich immer genau unter dem Kasten der grössten Zahl befindet. Hat er recht? Argumentiere.

(c) Hannah behauptet, dass die Addition bei den Fliesskommazahlen nicht kommutativ und nicht assoziativ ist. Hat sie recht? Argumentiere.

Aufgabe 3.5. *Die Ameisenkönigin möchte ausrechnen, wie viele Ameisen braucht sie, um* 10 *Reiskörnchen zu transportieren. Sie weiss, dass eine Ameise allein* $1/4$ *Reiskorn transportiert. Die Ameisenkönigin hat dazu folgendes Programm geschrieben.*

```
def nof_ameisen():
    sum = 0.0
    i = 0
    while node != 10.0:
      i += 1
       sum += 0.25
    return i
```

Listing 1: Programm von der Ameisenkönigin

Die Ameisencomputer arbeiten mit Fliesskommazahlen mit Mantissenlänge 5 *und Exponenten zwischen* -3 *und* 3*. Kann die Ameisenkönigin mit diesem Programm die gewünschte Anzahl Ameisen herausfinden? Falls ja, wie viele Ameisen braucht sie, um 10 Reiskörnchen zu transportieren laut diesem Programm? Falls nein, was ist die maximale Summe, die das Programm erreichen kann?*

4 Zusammenfassung

Wir haben gesehen, wie man im Computer reelle Zahlen durch **Fliesskommazahlen** approximieren kann. Da wir eine endliche Darstellung verwenden, gibt es eine endliche Anzahl Zahlen, die wir darstellen können. Es gibt also eine grösste und eine kleinste positive Zahl.

Die **grösste positive Zahl** kriegen wir, wenn wir die grösstmögliche Mantisse mit dem grössmöglichen Exponenten kombiniren, d.h. die Mantisse besteht aus lauter Einser und der Exponent ist maximal.

Die **kleinste positive Zahl** kriegen wir, wenn wir die kleinstmögliche Mantisse mit dem kleinstmöglichen Exponenten kombinieren. Da wir in der Darstellung verlangen, dass das erste Bit der Mantisse eine Eins ist, besteht die kleinstmögliche Mantisse aus einer führenden Eins und vielen Nullen. Der Exponent ist minimal.

Wir haben gelernt, die nächste und die vorherige darstellbare Zahl zu bestimmen. So haben wir gesehen, dass darstellbare Zahlen nicht gleichverteilt auf der Zahlengerade auftreten, sondern dass der **Abstand** zwischen benachbarten darstellbaren Zahlen wächst, wenn die Zahlen grösser werden.

Mit den Fliesskommazahlen kann man auch rechnen. Wir haben insbesondere die **Addition** kennengelernt.

Wenn man zwei Fliesskommazahlen zusammen addieren möchte, muss man sie zuerst zum gleichen Exponenten bringen, dann kann man die neuen Mantissen wie ganze Zahlen addieren. Am Schluss muss man sicherstellen, dass der Exponent und die Mantisse gültig sind: der Exponent muss zwischen dem minimalen und dem maximalen Exponenten sein, die Mantisse muss eine führende Eins haben.

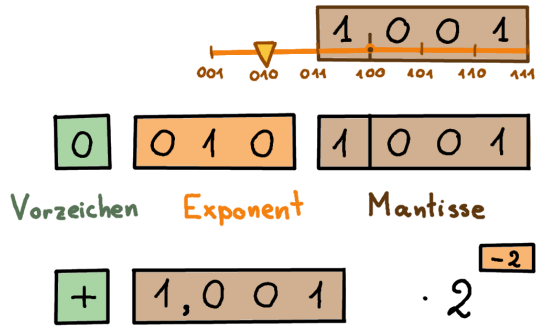
Wir haben gesehen, dass die Addition im Fliesskommazahlensystem Unterschiede zur Addition bei den reellen Zahlen aufweist. Erstens, nicht alle darstellbare Zahlen lassen sich exakt addieren. Zweitens, die Addition bei den Fliesskommazahlen ist **nicht assoziativ**. Es kann einen Unterschied machen, welche Teilsummen man zuerst berechnet.

Die Fliesskommazahlen sind eine mächtige Darstellung, die mit wenig Bits sehr unterschiedliche Zahlen speichern kann. Das hat aber auch seine Grenzen. Wir müssen in Kauf nehmen, dass die Resultate der Berechnungen nicht immer exakt sind.

5 Beispiellösungen

5.1 Einführung

Aufgabe 1.2 Das Vorzeichen ist positiv. Die Mantisse übernehmen wir aus dem Kasten. Die Kodierung vom Exponenten können wir am Seil ablesen. Den Exponenten bestimmen wir, indem wir die 100 auf dem Seil als 0 interpretieren und die Stellen zwischen der Null und der Markierung zählen. In diesem Fall sind es -2 Stellen.

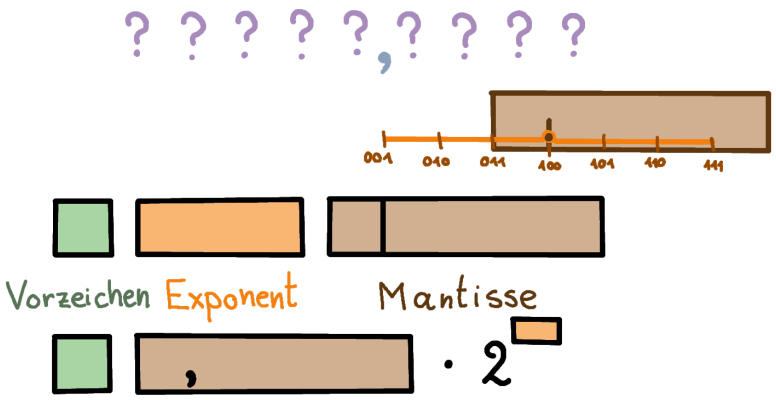


Den Dezimalwert berechenen wir, indem wir $(-1)^0 \cdot 1.001 \cdot 2^{-2} = 0.01001$ nach Dezimal konvertieren. In diesem Fall erhalten wir $1/4 + 1/32 = 9/32$.

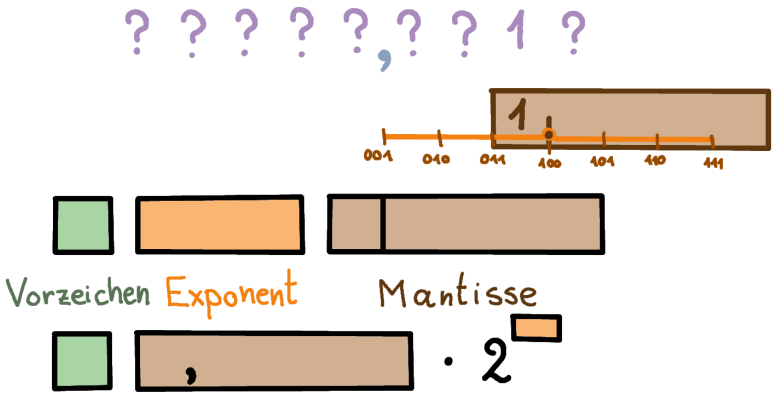
5.2 Fliesskommazahlen

Aufgabe 2.1 Wir konstruieren die kleinste positive Zahl im Fliesskommazahlensystem mit Mantissenlänge 5 und Exponenten von -3 bis 3.

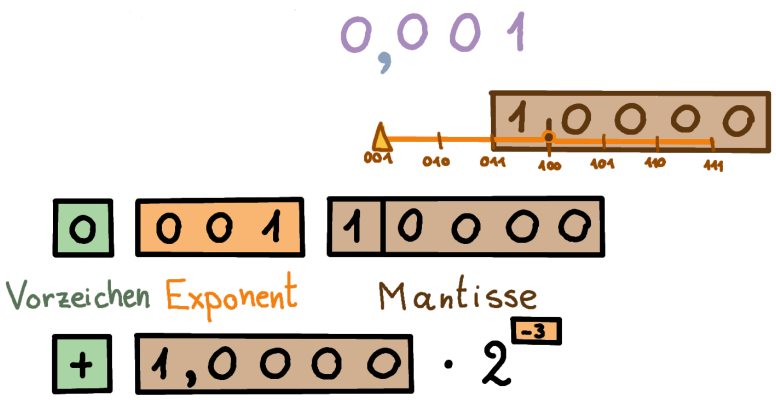
Als erstes platzieren wir den Kasten. Damit die Zahl möglichst klein wird, muss der Kasten nach rechts möglichst weit weg vom Komma stehen. Wir haben aber eine Einschränkung: Das Seil muss immer mit dem Komma verbunden bleiben.



Der Exponent muss also möglichst klein sein.
Was ist mit der Mantisse? Sicher muss eine Eins an der ersten Stelle stehen.

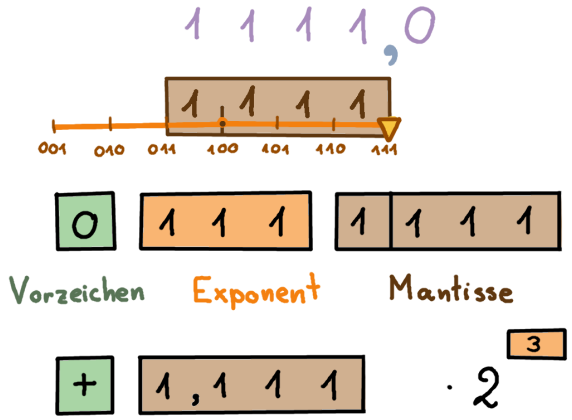


Damit die Mantisse möglichst klein wird, müssen wir so viele Stellen wie möglich auf Null setzen.

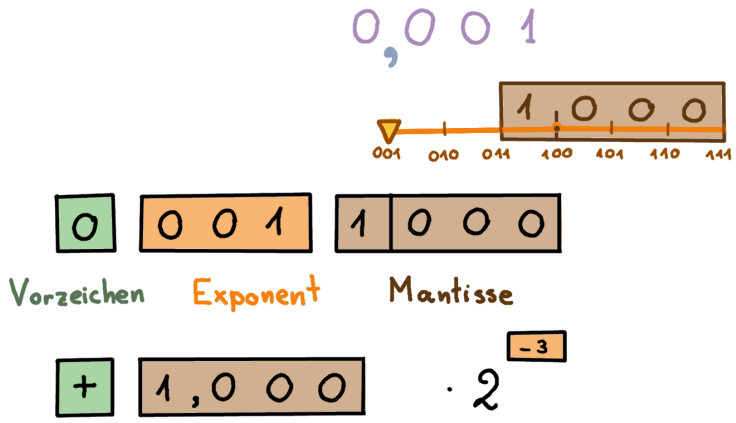


Die kleinste darstellbare Zahl in diesem Fliesskommazahlensystem ist also 1/8.

Aufgabe 2.2 Die grösste positive darstellbare Zahl in einem Fliesskommazahlensystem mit Mantissenlänge 4 und Exponent zwischen -3 und 3 ist 15. Das ist nicht viel kleiner als 15.5, die grösste positive darstellbare Zahl in einem Fliesskommazahlensystem mit einem Bit mehr für die Mantisse.



Die kleinste positive darstellbare Zahl in einem Fliesskommazahlensystem mit Mantissenlänge 4 und Exponent zwischen -3 und 3 ist auch 1/8, genau wie die kleinste positive darstellbare Zahl in einem Fliesskommazahlensystem mit einem Bit mehr für die Mantisse.



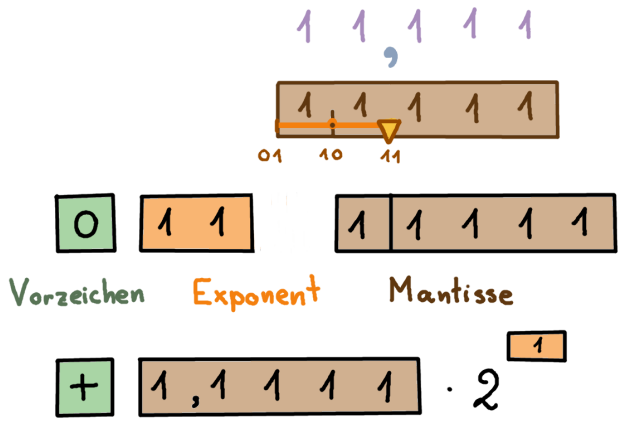
Wie wir sehen, die Länge der Mantisse scheint wenig Einfluss auf die grösste und kleinste positive darstellbare Zahlen zu haben.

Aufgabe 2.3

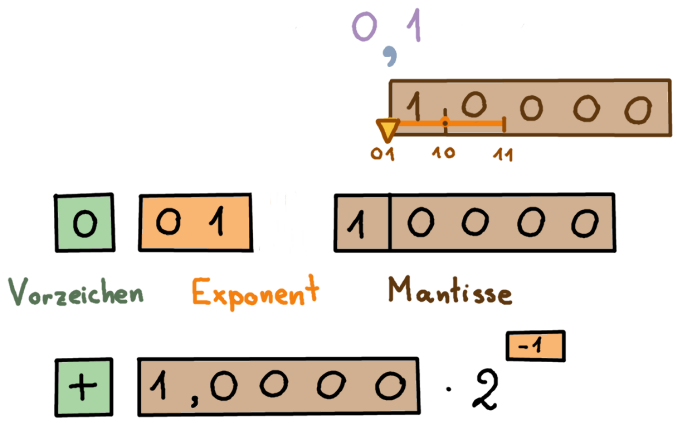
(a) Der Exponent liegt zwischen -1 und 1 und die mögliche Kodierungen sind `01, 10, 11`.

(b) Die Erwartung ist, dass die grösste positive darstellbare Zahl deutlich kleiner wird, weil das Seil viel kürzer ist, und wir den Kasten nicht mehr so weit nach links ziehen können, wie im Fliesskommazahlensystem mit 3 Bits für den Exponenten. Analog, die kleinste positive darstellbare Zahl wird deutlich grösser.

(c) Die grösste positive darstellbare Zahl in diesem System ist $3+7/8 = 31/8$. Wie erwartet, das ist viel grösser als in einem Fliesskommazahlensystem mit einem Bit mehr für den Exponenten.



Die kleinste positive darstellbare Zahl in diesem System ist 0.5. Das ist viel grösser als in einem Fliesskommazahlensystem mit einem Bit mehr für den Exponenten.



Aufgabe 2.4 Im Allgemeinen für einen Fliesskommazahlensystem mit Mantissenlänge m und Exponenten zwischen e_{min} und e_{max} findet man die grösste und kleinste positive Zahlen wie folgt.

Für die grösste positive Zahl wählt man den grösstmöglichen Exponenten e_{max} und die grösstmögliche Mantisse $1.111\ldots111$. In der Exponentialschreibweise ist die grösste Zahl also

$$1.1111111\ldots111 \cdot 2^{e_{max}}$$

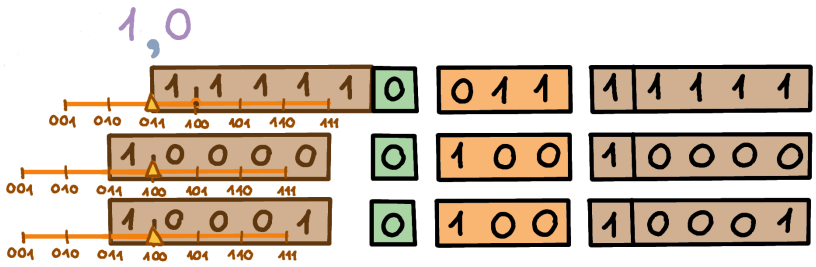
und hat das Bitmuster `0 1111...111 (1)111111...111`.

Für die kleinste positive Zahl wählt man den kleinsten möglichen Exponenten e_{min} und die kleinste mögliche Mantisse. Beachte, dass die Mantisse immer mit einer Eins starten muss. Die kleinste mögliche Mantisse ist deswegen $1.0000\ldots000$. In der Exonentialschreibweise ist die kleinste Zahl also

$$1.0000000\ldots000 \cdot 2^{e_{min}}$$

und hat das Bitmuster `0 0000...001 (1)00000000...000`.

Aufgabe 2.5 Die nächstkleinste, oder vorherige, darstellbare Zahl finden wir, indem wir die Mantisse kleiner zu machen versuchen. Da die Mantisse von 1 die kleinste mögliche Mantisse ist, müssen wir den Exponenten um Eins zurücksetzen und die grösstmögliche Mantisse wählen.

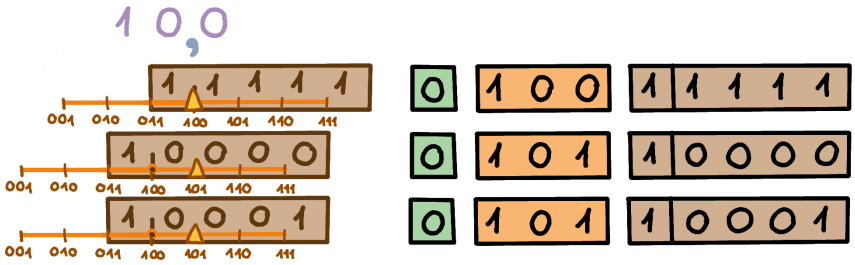


Die vorherige Zahl ist also 31/32.

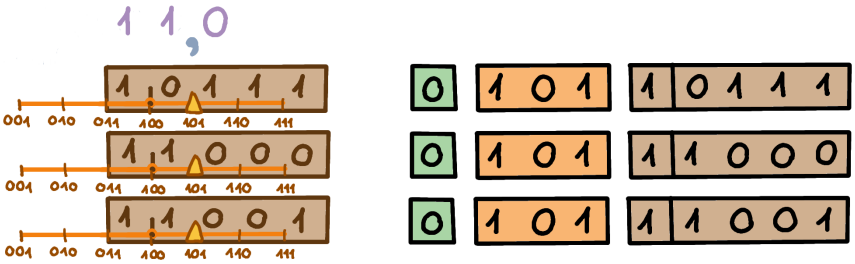
Beachte, dass der Abstand zur nächsten und vorherigen darstellbaren Zahlen in diesem Fall nicht symmetrisch ist: die nächste Zahl ist 1/16 entfernt, während die vorherige nur 1/32.

Aufgabe 2.6

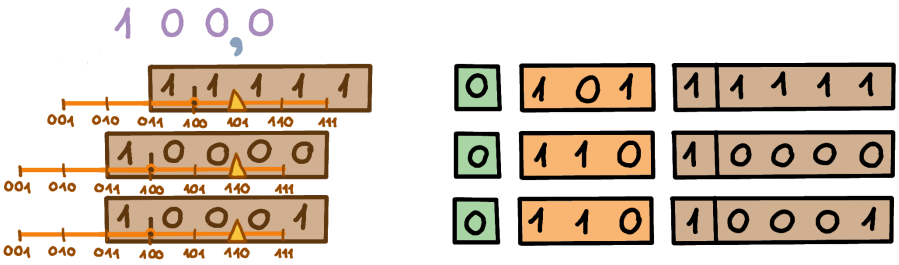
(a) Die Nachbarn von 2 sind 31/16 und 17/8.



(b) Die Nachbarn von 3 sind 23/8 und 25/8.



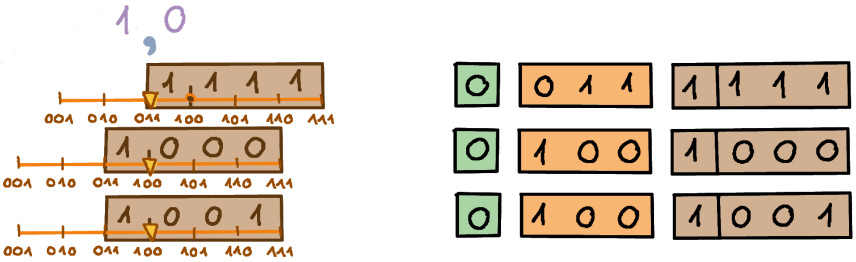
(c) Die Nachbarn von 4 sind 31/8 und 17/4.



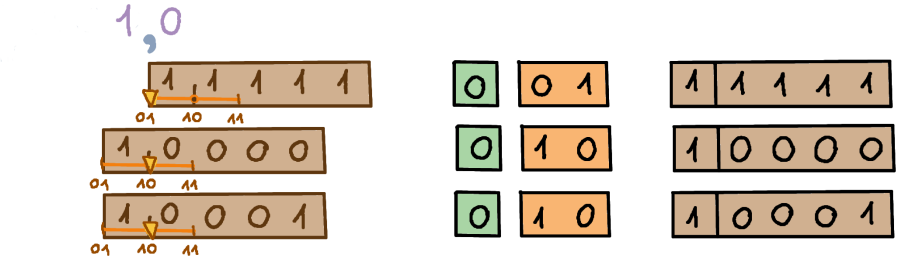
Die positive darstellbare Zahlen sind auf dem Zahlenstrahl nicht gleichverteilt.

Aufgabe 2.7

(a) Im Fliesskommazahlensystem mit Mantissenlänge 4 und Exponent zwischen -3 und 3 sind die Nachbarn von 1: 15/16 und 9/8.



(b) Im Fliesskommazahlensystem mit Mantissenlänge 5 und Exponent zwischen -1 und 1 sind die Nachbarn von 1: 31/32 und 17/16.



(c) Die Länge der Mantisse beeinflusst den Abstand zwischen darstellbaren Zahlen stärker als die Länge der Exponentenkodierung. Wenn der Kasten grösser ist, gibt es mehr Platz für signifikante Stellen und Zahlen können genauer approximiert werden. Das führt dazu, dass der Abstand zwischen darstellbaren Zahlen kleiner wird.

Aufgabe 2.8

(a) Nein, es gibt unendlich viele reelle Zahlen und endlich viele Fliesskommazahlen.

(b) Ja, die grösste Zahl ist $1.1111\ldots111 \cdot 2^{e_{max}}$.

(c) Ja, die kleinste Zahl ist $1.0000\ldots000 \cdot 2^{e_{min}}$.

(d) Die Länge der Exponentenkodierung beeinflusst den Bereich stärker als die Mantissenlänge. Wenn das Seil länger ist, kann man den Kasten weiter weg vom Komma platzieren und viel grössere oder kleinere Zahlen darstellen.

(e) Zum Beispiel, die Zahl 2.25 lässt sich in diesem System nicht exakt darstellen. In der binären Exponentialschreibweise diese Zahl ist $1.001 \cdot 2^1$. Um diese Zahl exakt zu speichern bräuchten wir 4 Bits für die Mantisse, wir haben aber nur 3.

(f) Nein, die darstellbare Fliesskommazahlen sind nicht gleichverteilt. Die kleineren stehen dichter beieinander, weil bei kleineren Zahlen die letzte Stelle der Mantisse weniger Wert ist.

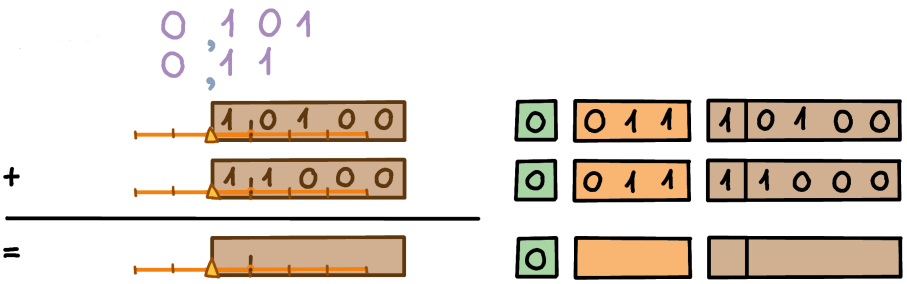
(g) Die Mantissenlänge beeinflusst stärker den Abstand zwischen positiven darstellbaren Zahlen in einem Fliesskommazahlensystem. Wenn der Kasten mehr Plätze hat, kann man mehr Stellen speichern und somit Zahlen genauer darstellen.

5.3 Addition

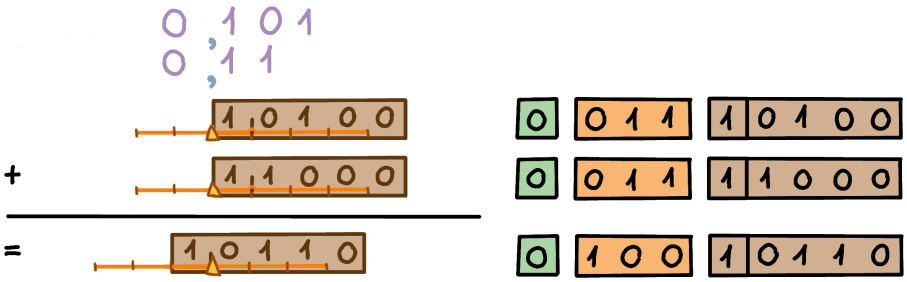
Aufgabe 3.1

(a) $5/8 + 3/4 = 11/8$, in der Exponentialschreibweise $1.0110 \cdot 2^0$

Im ersten Schritt schreiben wir die Zahlen auf.



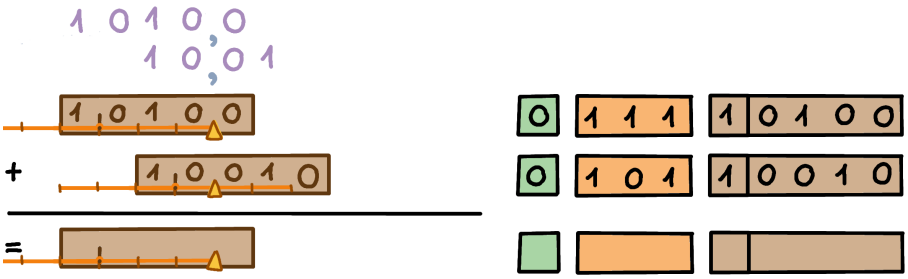
Da die zwei Kasten schon übereinander liegen, müssen wir sie nicht verschieben und können die Bits stellenweise zusammen addieren.



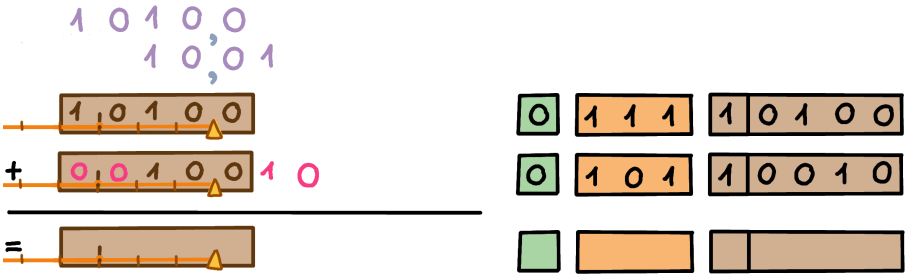
Der Kasten vom Ergebnis ist verschoben bezüglich den Kasten der Summanden.

(b) $10 + 2.25 = 12$, in der Exponentialschreibweise $1.1000 \cdot 2^3$

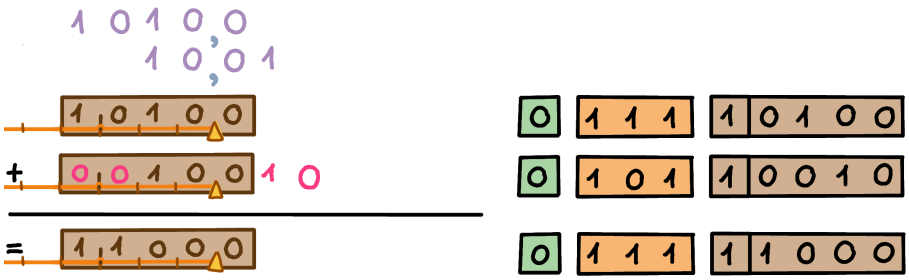
Im ersten Schritt schreiben wir die Zahlen auf.



Im zweiten Schritt schiben wir den Kasten von der kleinsten Zahl unter den Kasten der grössten Zahl. Dabei gehen zwei Stellen verloren, eine davon ist eine Eins.

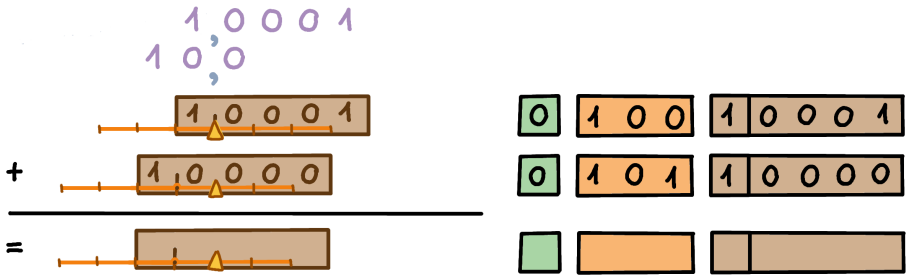


Nun können wir die Bits stellenweise zusammenrechnen.

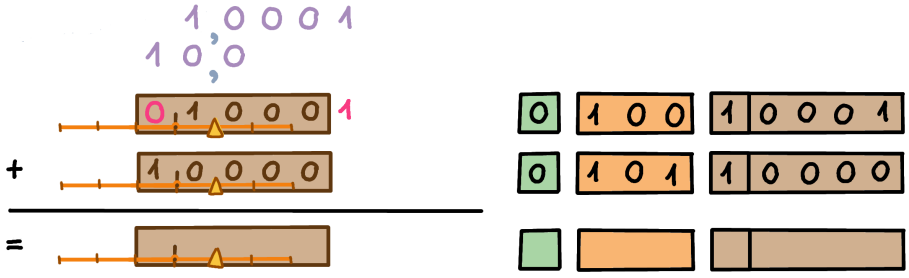


(c) $17/16 + 2 = 3$, in der Exponentialschreibweise $1.1000 \cdot 2^1$.

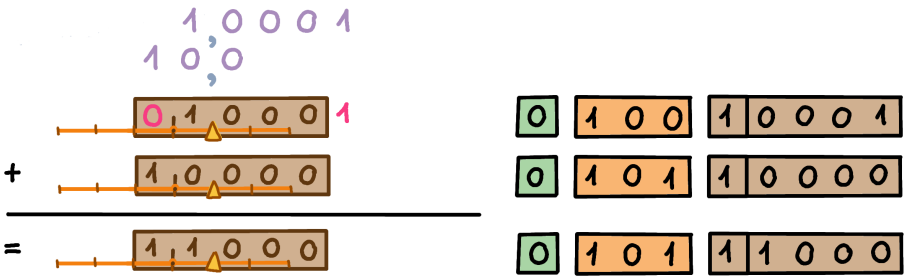
Im ersten Schritt schreiben wir die Zahlen auf.



Im zweiten Schritt schiben wir den Kasten von der kleinsten Zahl unter den Kasten der grössten Zahl. Dabei geht eine Stelle verloren.

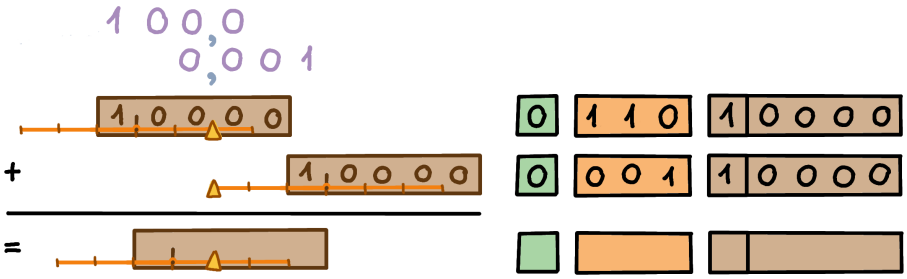


Nun können wir die Bits stellenweise zusammenrechnen.

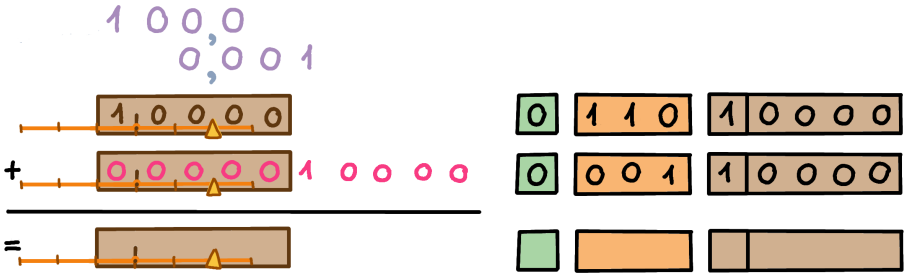


Aufgabe 3.3 Die maximale Zahl, die wir erreichen können, wenn wir $1/8 + 1/8 + \cdots + 1/8$ zusammen rechnen, ist 4.0.

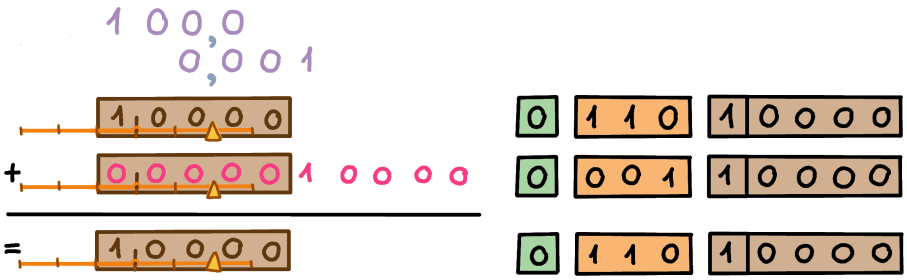
Zum einen, wenn wir die 4.0 erreicht haben, kommen wir nicht mehr weiter. Das sehen wir, wenn wir $4.0 + 1/8$ ausrechnen. Wie gewöhnlich schreiben wir zuerst die Summanden untereinander.



Wenn wir den Kasten von $1/8$ unter den Kasten von 4.0 verschieben, sehen wir, dass alle signifikanten Stellen von $1/8$ verloren gehen, auch die führende Eins.



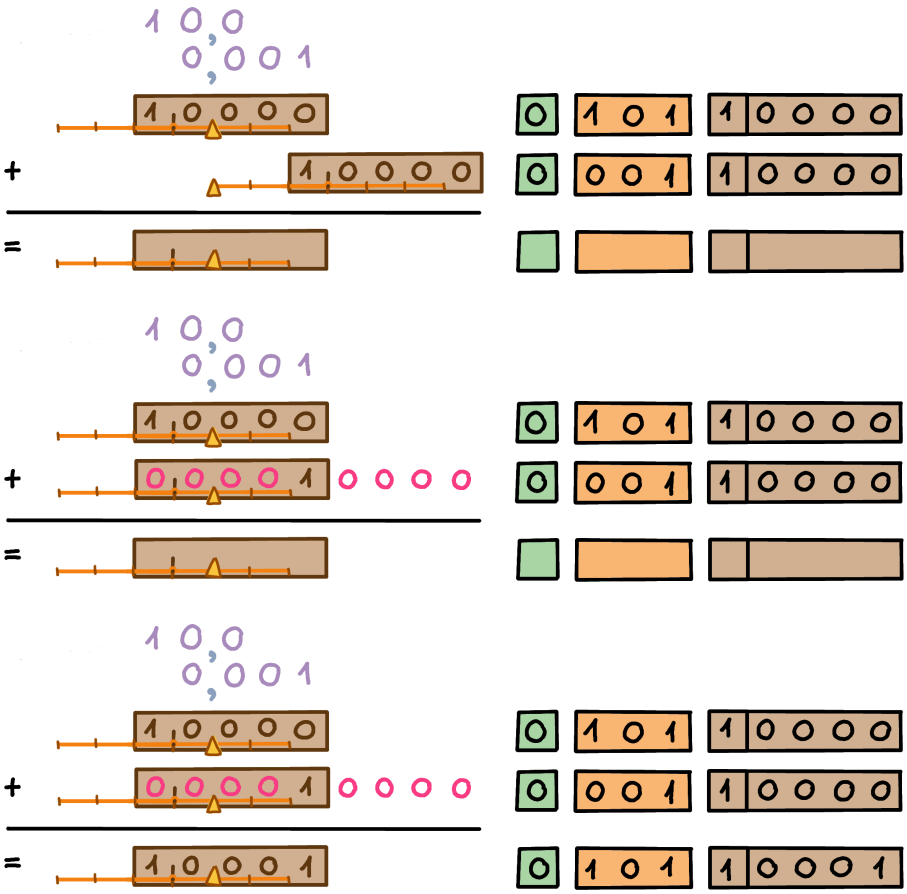
Deswegen, wenn wir $4.0 + 1/8$ ausrechnen, kriegen wir 4.0.



Egal wie viele 1/8 rechnen wir zusammen, bleiben wir immer bei 4.0.

Jetzt bleibt uns zu zeigen, dass wir die 4.0 auch tatsächlich erreichen können. Das Problem bei der 4.0 ist, dass alle signifikanten Stellen von 1/8 verloren gehen. Das passiert, weil der Unterschied zwischen dem Exponenten von 4.0 und dem Exponenten von 1/8 die ganze Mantissenlänge beträgt. Das passiert bei einem kleineren Exponenten nicht. Zum Beispiel, wenn wir $2.0+1/8$ ausrechnen, sehen wir, dass das Ergebnis wie erwartet 17/8 ist.

Um zu zeigen, dass das Problem erst bei $4.0 + 1/8$ auftritt, rechnen wir $2.0 + 1/8$. Das Ergebnis ist wie erwartet 17/8.



Wir erreichen also die 4.0 nach 32 Summanden und kommen dann nicht mehr weiter.

Aufgabe 3.4

(a) Der Wert der Bits in der Mantisse hängt vom Exponenten ab. Zum Beispiel, dieselbe Mantisse 1.0000 mit unterschiedlichen Exponenten kann 4, 2, 1, 1/2, 1/4 und 1/8 darstellen. Wir wollen nicht, dass $1 + 2$ das gleiche Ergebnis liefert die $1 + 1/4$. Wir wollen nur Bits mit dem gleichen Wert zusammen addieren. Deswegen müssen wir vor der Addition sicherstellen, dass die Kasten der beiden Summanden exakt untereinander stehen.

(b) Die Aussage von Gregory ist falsch. Der Kasten vom Ergebnis kann sich bewegen bezüglich des Kastens vom grössten Summanden. Dies passiert, zum Beispiel, wenn man $2.5 + 1.75$ ausrechnet.

(c) Hannah hat teilweise recht. Die Addition bei den Fliesskommazahlen ist kommutativ aber nicht assoziativ.

Wenn wir zwei Zahlen zusammen addieren und diese zwei Zahlen vertauschen, kriegen wir das gleiche Ergebnis auch bei Fliesskommazahlen.

Wenn wir aber die Reihenfolge verändern, in welcher die Zahlen zusammengerechnet werden, können wir unterschiedlich Ergebnisse bekommen. Das passiert, weil wir nur dann den exakten Wert ausrechnen können, wenn die Grössenordnung der Teilsummanden ähnlich ist.

Aufgabe 3.5 Nein, das Programm der Ameisenkönigin wird unendlich lange laufen und die Anzahl Ameisen, die es braucht, um 10 Reiskörnchen zu transportieren, nie ausgeben. Das Problem ist analog zu dem, was wir in Aufgabe 3.3 gesehen haben. Das Programm läuft wie erwartet bis wir die 8.0 erreichen. Wenn wir aber 1/4 dazu rechnen, dann verlieren wir alle signifikanten Stellen von 1/4 und die 8.0 bleibt unverändert.