MATH 112
Section 3.1: Understanding Addition

Prof. Jonathan Duncan

Walla Walla University

Autumn Quarter, 2006
Introduction to Addition

Interpretations of Addition

Addition Properties and Algorithms

Addition Algorithms

Mental Addition Algorithms

Conclusion
Don’t We Know This Already?

In chapter three we will examine addition, subtraction, multiplication, and division of whole numbers. These four operations form the core of elementary school mathematics.

What Are We Trying To Teach You?

We assume that you already know how to add, supply, multiply and divide, so what is it we are trying to teach?

- Interpretations of various operations
- Connections between operations
- Non-standard algorithms
- An understanding of why standard algorithms work
Addition as the Union of Sets

One of the most basic models of addition is the set model. This is often a good starting point for children.

Example
Billy has three oranges and Jane has 4. Jane gives Billy her oranges. How many oranges does Billy have now?

The Set Model
If $A$ and $B$ are disjoint sets with $a$ and $b$ elements respectively, then $a + b$ is the number of elements in $A \cup B$. 
Addition as a Number Line

Another model of addition which works well for certain applications is the number line model.

Example
Sam walks home from school each day. One day he stops at the candy store. If he walks two miles from the school to the store and three miles from the store home, how many miles did he walk from the school to his house?

The Number Line Model
Starting with a number line with zero marked on the line, measure off the distance \( a \) from zero. Then, starting at the point \( a \), measure off the distance \( b \) and place a point on the number line. The sum \( a + b \) is the total distance from zero to the point marked.
Interpretations of Addition

Bringing the Models Together

What do these two models have in common? They both use pictorial explanations to demonstrate the addition process.

Modeling Addition

In general, $a + b$ is modeled by combining two parts to make a whole as shown below.

Advantages and Disadvantages

The model above has both advantages and disadvantages.

- **Advantages:** easy to understand, connections to real world
- **Disadvantages:** not easy to use in algorithms
As we saw in chapter two, algebra can be used to express properties which hold for all numbers. We do this now with addition.

Properties of Addition

For whole numbers $a$, $b$, and $c$ the following properties of addition hold.

- **Identity Property**: $0 + a = a$
- **Commutative Property**: $a + b = b + a$
- **Associative Property**: $(a + b) + c = a + (b + c)$
- **Closure**: $a + b$ is another whole number
- **Counting**: $a + 1$ is the next whole number after $a$
The Base 10 Addition Table

Sometimes we can see patterns in an operation by looking at the operation table.

<table>
<thead>
<tr>
<th>+</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
</tr>
</tbody>
</table>
Modeling Base 10 Addition

As you have seen in your lab work, we can use manipulatives to model numerals (think of flats, longs, and units). These manipulatives are also useful to model operations such as addition.

Example

Use base 10 blocks to model each addition problem.

\[
\begin{array}{c}
2 & 4 \\
+ & 3 & 7 \\
\hline
6 & 1 \\
\end{array}
\quad
\begin{array}{c}
5 & 6 \\
+ & 4 & 5 \\
\hline
1 & 0 & 1 \\
\end{array}
\]
The Standard Algorithm

There are many different algorithms for addition. Some of you may not have learned what I call the “standard algorithm” in school. I call it standard only because it is what has been most commonly taught in the U.S. in the last 20 years.

Standard Addition Algorithm

In the standard algorithm for addition, single digits are added in columns from right to left. When the sum of a column is greater than 9, the second (and possibly third) digits are carried over to the next column to be included in the next sum.

Example

Use the standard addition algorithm to find the following sum.

\[
\begin{array}{c}
4 & 8 & 2 \\
3 & 9 & 5 \\
+ & 1 & 7 & 9 \\
\end{array}
\]
Another popular method for adding multi-digit numbers is named for the scratch mark used during the addition process.

**Scratch Addition Algorithm**

In the scratch algorithm for addition, single digits are added in columns until a sum greater than 9 is reached. Then, the digit at which that sum is reached is scratched out, and the last digit of the sum is written next to it. When the end of a column is reached, the scratch marks are tallied and added to the top of the next column to be included in its sum.

**Example**

Use the scratch addition algorithm to find the following sum.

\[
\begin{array}{c}
4\ 8\ 2 \\
3\ 9\ 5 \\
+\ 1\ 7\ 9 \\
\hline
1\ 7\ 9
\end{array}
\]
The “Adding Up” Algorithm

The adding up algorithm is not as well known, but does have some unique features such as adding from left-to-right instead of right-to-left.

Adding Up Addition Algorithm

In the adding up algorithm for addition, columns are added from left-to-right and the sum is written above the column. If the sum in the first column is two or more digits, all digits are written down. In subsequent columns if the sum is more than one digit, the extra digits are added to the sum of the previously added column.

Example

Use the adding up addition algorithm to find the following sum.

\[
\begin{array}{c}
4 & 8 & 2 \\
3 & 9 & 5 \\
+ & 1 & 7 & 9 \\
\end{array}
\]
The lattice method is similar to the standard method, but no carrying is done. Instead, the sum is found using a two-step process.

**Lattice Addition Algorithm**

In the lattice algorithm for addition, columns are added from left-to-right or right-to-left. Two-digit sums (or possibly three) are written in a lattice below each column. Once all columns have been added and the sums entered into the lattice, the diagonals of the lattice are added together to get the final sum.

**Example**

Use the lattice addition algorithm to find the following sum.

\[
\begin{array}{ccc}
4 & 8 & 2 \\
3 & 9 & 5 \\
+ & 1 & 7 & 9 \\
\end{array}
\]
Justifying the Algorithms

Why do these algorithms all work? Let’s look at a simpler example in which we pay special attention to the place value.

Example

\[ 267 + 133 = (2 \times 100 + 6 \times 10 + 7 \times 1) + (1 \times 100 + 3 \times 10 + 3 \times 1) \]
\[ = (2 \times 100 + 1 \times 100) + (6 \times 10 + 3 \times 10) + (7 \times 1 + 3 \times 1) \]
\[ = (2 + 1) \times 100 + (6 + 3) \times 10 + (7 + 3) \times 1 \]
\[ = 3 \times 100 + 9 \times 10 + 10 \times 1 \]
\[ = 3 \times 100 + 10 \times 10 \]
\[ = 4 \times 100 \]
\[ = 400 \]
Algorithms for Mental Addition

The algorithms we discussed up until now are designed for each computation on paper or in a setting where one can easily keep track of each step of the computation. When we want to add numbers quickly without such tools, we may need different algorithms.

Example

Find each of the sums below in your head as quickly and accurately as possible.

1. $37 + 42$
2. $59 + 35$
3. $66 + 29$
4. $186 + 125$
5. $656 + 227$
Leading Digits Method

One method of mental addition is to add starting with the leading digit, similar to what was done in the adding up algorithm.

Example

To add $37 + 42$ using the leading digit method, follow these steps:

- First add the leading digits to get $3 + 4 = 7$
- Next add the final digits $7 + 2 = 9$
- Finally, add ten times the leading digit sum to the final digit sum giving $70 + 9 = 79$

Example

Try this method with the other sums: $59 + 35$, $66 + 29$, and $186 + 125$. 
Compensation Method

Another mental addition method is to make a harder addition problem easier by moving a piece of one number into the other.

Example

To add $59 + 35$ using the compensation method, follow these steps:

- First note that 59 is very close to the nice round number 60.
- Compensate for 59’s deficiencies by moving one unit from 35 into the 59, making the problem $60 + 34$.
- Finally, use the leading digit method to get $90 + 4 = 94$.

Example

Use compensation to find the other sums: $66 + 29$, $186 + 125$, and $656 + 227$. 
Break and Bridge Method

In the break and bridge method we make inconvenient addition problems more convenient by breaking them into pieces and then bridging the pieces back together.

**Example**

To add $186 + 125$ using the break and bridge method, follow these steps:

- First break 125 into two pieces, 120 and 5.
- Next add 186 and 120 using the leading digit method to get 306.
- Finally, bridge the left over 5 back in to get 311.

**Example**

Use the break and bridge method to find the other sums: $37 + 42$, $66 + 29$, and $656 + 227$. 
Important Concepts

Things to Remember from Section 3.1

1. Ways to model addition
2. Properties of addition
3. Addition algorithms and why they work
4. Mental addition algorithms