Winter 2017 COMP-610
Information Structures I
Prof. Claude Crépeau
Grade School Algorithms
Grade School Algorithms

Representation quite inefficient
"+" easy to describe
Grade School Algorithms

Representation quite inefficient

"×" easy to describe
Inefficient Representation

1 000 000 kids meditate for world peace
@ Phra Shammakaya temple Thailand
Roman Grade School

Representation very efficient
Roman Grade School

MMCCCXXIV
+ MCMXXXVII

Representation very efficient
"+" not too complicated to describe

http://turner.faculty.swau.edu/mathematics/materialslibrary/roman/
Roman Grade School

MMCCCXXIV
×  MCMXXXVII

Representation very efficient
"×" complicated to describe

http://turner.faculty.swau.edu/mathematics/materialslibrary/roman/
Grade School Algorithms: Addition

Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.
Algorithm 1 Addition (base 10): Add two \( N \) digit numbers \( a \) and \( b \) which are represented as arrays of digits.
Grade School Algorithms

Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

\[
\begin{array}{c}
2343 \\
+ 4519 \\
\hline
12
\end{array}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

\[
\begin{array}{c}
2343 \\
\underline{+} \quad 4519 \\
\hline
1 \\
\underline{2796} \\
\end{array}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.

\[
\begin{array}{c}
0 \\
2343 \\
+ \\
4519 \\
\hline
62 \\
\end{array}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

\[
\begin{array}{c}
\text{Algorithm 1 Addition (base 10)}: \text{Add two } N \text{ digit numbers } a \text{ and } b \text{ which are represented as arrays of digits}
\end{array}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.

\[
\begin{array}{c}
2343 \\
+ 4519 \\
\hline
06862
\end{array}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

\[
\begin{align*}
[2][3][4][3] & \quad + \quad [4][5][1][9] & \quad + & \quad 0 \quad 0 \quad 0 \quad 1 \\
0[6][8][6][2] & \quad + \quad 6862
\end{align*}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

\[
\begin{align*}
carry & \leftarrow 0 \\
\text{for } i & \leftarrow 0 \text{ to } N-1 \text{ do} \\
\quad r[i] & \leftarrow R[a[i], b[i], carry] \\
\quad carry & \leftarrow L[a[i], b[i], carry] \\
\text{end for} \\
r[N] & \leftarrow carry
\end{align*}
\]
Grade School Algorithms

**Algorithm 1** Addition (base 10): Add two \( N \) digit numbers \( a \) and \( b \) which are represented as arrays of digits

\[
L[i,j,0] \quad R[i,j,0]
\]

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></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>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>R</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></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>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>0</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>0</td>
<td>1</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>0</td>
<td>1</td>
<td>2</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>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>9</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>
</tr>
<tr>
<td>9</td>
<td>9</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>
</tr>
</tbody>
</table>
Algorithm 1: Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.
Grade School Algorithms

**Algorithm 1** Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

$L[i,j,1]$ \quad $R[i,j,1]$
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.

$L[i,j,1]$  

$R[i,j,1]$
Grade School Algorithms

**Algorithm 1** Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

$$(3+9+1) \mod 10 = (3+9+1) / 10 = 13$$
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits.

\[
\begin{align*}
(A+B+C) / 10 &= E \\
E.D &= (A+B+C) \mod 10
\end{align*}
\]
Algorithm 1 Addition (base 10): Add two $N$ digit numbers $a$ and $b$ which are represented as arrays of digits

---

carry = 0

for $i = 0$ to $N - 1$ do
    $r[i] \leftarrow (a[i] + b[i] + carry) \mod 10$
    carry $\leftarrow (a[i] + b[i] + carry)/10$
end for

$r[N] \leftarrow carry$

---
Algorithm 1 Addition (base $\beta$): Add two $N$ $\beta$-git numbers $a$ and $b$ which are represented as arrays of $\beta$-gits.

\[
\begin{array}{c}
\text{(A+B+C) / } \beta = E + D = (A+B+C) \% \beta
\end{array}
\]
Algorithm 1 Addition (base $\beta$): Add two $\beta$-git numbers $a$ and $b$ which are represented as arrays of $\beta$-gits

\begin{align*}
\text{carry} &= 0 \\
\text{for } i = 0 \text{ to } N - 1 \text{ do } \\
& \quad r[i] \leftarrow (a[i] + b[i] + \text{carry}) \mod \beta \\
& \quad \text{carry} \leftarrow (a[i] + b[i] + \text{carry}) / \beta \\
\text{end for} \\
& \quad r[N] \leftarrow \text{carry}
\end{align*}
Example: addition base 8

\[(1205)_8 + (736)_8\]
Example: addition base 8

\[(1205)_{8}
\]

\[+(736)_{8}\]

\[\underline{\hspace{1cm}}\]

\[(13)_{8}\]
Example: addition base 8

\[
\begin{array}{c}
\text{(1205)}_8 \\
+ \text{(736)}_8 \\
\hline
\text{(3)}_8
\end{array}
\]

+ 0 1 2 3 4 5 6 7
0 0 1 2 3 4 5 6 7
1 1 2 3 4 5 6 7 10
2 2 3 4 5 6 7 10 11
3 3 4 5 6 7 10 11 12
4 4 5 6 7 10 11 12 13
5 5 6 7 10 11 12 13 14
6 6 7 10 11 12 13 14 15
7 7 10 11 12 13 14 15 16
Example: addition base 8

\[
\begin{array}{c}
0 & 1 \\
(1205)_8 \\
+ & (736)_8 \\
\hline \\
(43)_8
\end{array}
\]
Example: addition base 8

\[
\begin{array}{c}
(1205)_8 \\
+ \quad (736)_8 \\
\hline
(1143)_8
\end{array}
\]
Example: addition base 8

\[
\begin{array}{c}
1 & 0 & 1 \\
(1205)_8 \\
+ & (736)_8 \\
\hline
(143)_8 \\
\end{array}
\]
Example: addition base 8

\[
\begin{array}{c}
  1 & 0 & 1 \\
\hline
(1205)_8 \\
+ (736)_8 \\
\hline
(2143)_8 \\
\end{array}
\]
Example: addition base 8

\[
\begin{array}{cccc}
1 & 0 & 1 \\
(1205)_8 & + & (736)_8 \\
\hline
(2143)_8 \\
\end{array}
\]

(1123 in base ten)
Example: addition base 8

\[ \begin{array}{c}
101 \\
(1205)_{8} \\
+ (736)_{8} \\
\hline \\
(2143)_{8} = (1123)_{x}
\end{array} \]
Grade School Algorithms:
Subtraction

6343
—
4519
Grade School Algorithms

\[
\begin{array}{c}
\text{31} \\
\hline
\text{6343} \\
- \\
\text{4519} \\
\hline
\text{1824}
\end{array}
\]
Grade School Algorithms

\[
\begin{array}{c}
6343 \\
- 4519 \\
\hline
4
\end{array}
\]
Grade School Algorithms

\[ \frac{31}{6343} - \frac{4519}{24} \]
Grade School Algorithms

\[
\begin{array}{c}
5_1 & 3_1 \\
6343 \\
- \\
4519 \\
\hline
24
\end{array}
\]
Grade School Algorithms

\[
\begin{array}{c}
  5131 \\
  \hline
  6343 \\
  \hline
  4519 \\
  \hline
  824
\end{array}
\]
Grade School Algorithms

\[
\begin{array}{c}
5 \quad 1 \\
6343 \\
\hline
4519 \\
1824
\end{array}
\]
Grade School Algorithms: Multiplication

Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

$$
\begin{array}{c}
\text{352} \\
\times \quad 4 \\
\hline
\end{array}
$$
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
\text{352} \\
\times 4 \\
\hline
\text{8}
\end{array}
\]
Grade School Algorithms

Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

$352 \times 4 = 8$
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
352 \\
\times \\
4 \\
\hline
208
\end{array}
\]
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
20 \\
\times 4 \\
\hline
08
\end{array}
\]
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[ 20 \times 352 = 1408 \]
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

352
\[ \times \]
96
---
1408
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
352 \\
\times \ 964 \\
\hline
1408 \\
21120
\end{array}
\]
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
352 \\
\times \quad 964 \\
\hline
1408 \\
21120 \\
316800
\end{array}
\]
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{array}{c}
[3][5][2] \\
\times [9][6][4] \\
\hline
[3][3][9][3][2][8] \\
\end{array}
\]

352
$x$ 964

\[
\begin{array}{c}
[3][3][9][3][2][8] \\
\end{array}
\]

1408

\[
\begin{array}{c}
[3][3][9][3][2][8] \\
\end{array}
\]

21120

316800

\[
\begin{array}{c}
[3][3][9][3][2][8] \\
\end{array}
\]

339328
Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

\[
\begin{align*}
\text{for } j & = 0 \text{ to } N - 1 \text{ do} \\
\quad \text{carry} & \leftarrow 0 \\
\quad \text{for } i & = 0 \text{ to } N - 1 \text{ do} \\
\quad & \quad \text{prod} \leftarrow (a[i] \times b[j] + \text{carry}) \\
\quad & \quad \text{tmp}[j][i + j] \leftarrow \text{prod} \mod 10 \\
\quad & \quad \text{carry} \leftarrow \text{prod} / 10 \\
\quad \text{end for} \\
\quad \text{tmp}[j][N + j] \leftarrow \text{carry} & \\
\text{end for} \\
\text{carry} & \leftarrow 0 \\
\text{for } i & = 0 \text{ to } 2 \times N - 1 \text{ do} \\
\quad \text{sum} & \leftarrow \text{carry} \\
\quad \text{for } j & = 0 \text{ to } N - 1 \text{ do} \\
\quad & \quad \text{sum} \leftarrow \text{sum} + \text{tmp}[j][i] \\
\quad \text{end for} \\
\quad r[i] & \leftarrow \text{sum} \mod 10 \\
\quad \text{carry} & \leftarrow \text{sum} / 10 \\
\text{end for} \\
\text{r}[2 \times N] & \leftarrow \text{carry}
\end{align*}
\]
Multiplication

\[
\text{for } j = 0 \text{ to } N - 1 \text{ do}
\]
\[
carry \leftarrow 0
\]
\[
\text{for } i = 0 \text{ to } N - 1 \text{ do}
\]
\[
prod \leftarrow (a[i] \ast b[j] + carry)
\]
\[
tmp[j][i + j] \leftarrow prod \% 10
\]
\[
carry \leftarrow prod / 10
\]
\[
\text{end for}
\]
\[
tmp[j][N + j] \leftarrow carry
\]
\[
\text{end for}
\]
Multiplication

carry $\leftarrow 0$

for $i = 0$ to $2 \times N - 1$ do
    sum $\leftarrow$ carry
    for $j = 0$ to $N - 1$ do
        sum $\leftarrow$ sum + tmp[j][i]
    end for
    r[i] $\leftarrow$ sum\%10
    carry $\leftarrow$ sum/10
end for

r[2 \times N] $\leftarrow$ carry
Algorithm 2 Multiplication (base $\beta$) of two numbers $a$ and $b$

for $j = 0$ to $N - 1$ do
  $carry \leftarrow 0$
  for $i = 0$ to $N - 1$ do
    $prod \leftarrow (a[i] \times b[j] + carry)$
    $tmp[j][i + j] \leftarrow prod \mod \beta$
    $carry \leftarrow prod / \beta$
  end for
  $tmp[j][N + j] \leftarrow carry$
end for

$carry \leftarrow 0$
for $i = 0$ to $2 \times N - 1$ do
  $sum \leftarrow carry$
  for $j = 0$ to $N - 1$ do
    $sum \leftarrow sum + tmp[j][i]$
    $r[i] \leftarrow sum \mod \beta$
    $carry \leftarrow sum / \beta$
  end for
  $r[2 \times N] \leftarrow carry$
Multiplication base 8

\[
\begin{array}{c}
(1205)_8 \\
\times \ (736)_8 \\
\hline
(7436)_8 \\
(36170)_8 \\
(1064300)_8 \\
\hline
(1132126)_8 = (308310)_x
\end{array}
\]
Grade School Algorithms:
Long Division

| 723 | 41672542996 |
Grade School Algorithms

5 . . .

723 | 41672542996
5...

723 | 41672542996

3615
Grade School Algorithms

5 . . .

-------------
723 | 41672542996
3615
-----
552 . . .
Grade School Algorithms

5 ... 

723 | 5522542996
Grade School Algorithms

723 | 5522542996
5061

461 ...
Grade School Algorithms

\[ \begin{array}{c|c}
57638372 & 50 \\
723 & \\
\end{array} \]
Grade School Algorithms

\[ 41672542996 \div 723 = 57638372 \]

\[ 41672542996 \mod 723 = 50 \]
Grade School Algorithms

0.

723 | 50

Multiplication Table
Grade School Algorithms

\[
\begin{array}{c|c|c}
\hline
723 & 0.0 & 500 \\
\hline
\end{array}
\]
Grade School Algorithms

723 | 50
500
5000
-4338
662
Grade School Algorithms

0.069

\[
\begin{array}{c}
723 \\
\mid \\
662 \\
6620 \\
\hline
6507 \\
\hline
113
\end{array}
\]
Grade School Algorithms

\[ \begin{array}{c|c}
0.0691 & 113 \\
723 & 1130 \\
\hline
\end{array} \]

\[ \begin{array}{c|c}
 & -723 \\
\hline
\end{array} \]

\[ \begin{array}{c|c}
 & 407 \\
\hline
\end{array} \]
Grade School Algorithms

\[
\begin{array}{c}
0.06915 \\
\hline
723 \\
\hline
407 \\
4070 \\
\hline
-3615 \\
\hline
455
\end{array}
\]
Grade School Algorithms

0.069156

723 | 455

4550

-4338

212
Grade School Algorithms

0.069156293222683264177040110650
069156293222683264177040110650...

| 723 | 50 |

Continue the expansion until a remainder is repeated.
Grade School Algorithms

\[ \begin{array}{c}
0.069156293222683264177040110650 \\
0.069156293222683264177040110650 \ldots
\end{array} \]

\[ \begin{array}{c}
723 \\
\hline
50
\end{array} \]

\[ \begin{array}{c}
\text{Client Name:} \\
\text{Multiplication Table}
\end{array} \]

\[ \begin{array}{c}
\text{Source: Ibanamet.} \\
\text{www.icanhaveaninternet.com}
\end{array} \]

\[ = 0.069156293222683264177040110650 \]
Grade School Algorithms

\[ 723 \div 41672542996 = 57638372.069156293222683264177040110650 \]
Analysis of Algorithms
Analysis of Addition

\[
\text{Time}(N) = c_1 + c_2 \times N
\]
Analysis of Multiplication

for $j = 0$ to $N - 1$ do

$\text{cst} \{ \text{carry} \leftarrow 0$ 

for $i = 0$ to $N - 1$ do

$\text{prod} \leftarrow (a[i] \times b[j] + \text{carry})$

$\text{tmp}[j][i + j] \leftarrow \text{prod}\%10$

$\text{carry} \leftarrow \text{prod}/10$

end for

end for

$\text{cst} \{ \text{tmp}[j][N + j] \leftarrow \text{carry}$

end for
Analysis of Multiplication

```latex
\begin{align*}
\text{cst}\{ 
&\text{carry} \leftarrow 0 \\
&\text{for } i = 0 \text{ to } 2 \times N - 1 \text{ do} \\
&\quad \text{cst}\{ 
&\quad \text{sum} \leftarrow \text{carry} \\
&\quad \text{for } j = 0 \text{ to } N - 1 \text{ do} \\
&\quad\quad \text{cst}\{ 
&\quad\quad \text{sum} \leftarrow \text{sum} + \text{tmp}[j][i] \\
&\quad\quad \text{end for} \\
&\quad \text{end for} \\
&\quad r[i] \leftarrow \text{sum}\%10 \\
&\quad \text{cst}\{ 
&\quad \text{carry} \leftarrow \text{sum}/10 \\
&\quad \text{end for} \\
&\text{end for} \\
&r[2 \times N] \leftarrow \text{carry}
\end{align*}
```
Analysis of Algorithms

Algorithm 2 Multiplication (base 10) of two numbers $a$ and $b$

for $j = 0$ to $N - 1$ do
  carry $\leftarrow 0$
  for $i = 0$ to $N - 1$ do
    prod $\leftarrow (a[i] \times b[j] + carry)$
    tmp[$j$][$i + j$] $\leftarrow$ prod$\%10$
    carry $\leftarrow$ prod/10
  end for
  tmp[$j$][$N + j$] $\leftarrow$ carry
end for

carry $\leftarrow 0$
for $i = 0$ to $2 \times N - 1$ do
  sum $\leftarrow$ carry
  for $j = 0$ to $N - 1$ do
    sum $\leftarrow$ sum + tmp[$j$][$i$]
  end for
  r[$i$] $\leftarrow$ sum$\%10$
  carry $\leftarrow$ sum/10
end for
r[$2 \times N$] $\leftarrow$ carry

$\text{Time}(N) = C_1 + C_2 \times N + C_3 \times N^2$
Analysis of Algorithms

Addition

\[ \text{Time}(N) = c_1 + c_2 \times N \]

Multiplication

\[ \text{Time}(N) = c_1 + c_2 \times N + c_3 \times N^2 \]
Analysis of Algorithms

Addition

\[ \text{Time}(N) \text{ is } O(N) \]
\[ \text{Time}(N) \in O(N) \]

Multiplication

\[ \text{Time}(N) \text{ is } O(N^2) \]
\[ \text{Time}(N) \in O(N^2) \]
Analysis of Algorithms

- Unary Representation
- Multiplication
- Decimal/binary Representation
- Addition

Graph showing the growth of different functions:
- $50x$ (blue line)
- $x^2$ (red line)
- $2x$ (green line)

X-axis: 1 to 10
Y-axis: 250 to 2000
Today, the best known Multiplication algorithm has running time \( O(N \times 2^{\log^* N}) \) barely slower than Addition…

(\( \log^* N = \) number of log until \(< 1 \) )