

The bigintcalc package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2007/11/11 v1.1

Abstract

This package provides expandable arithmetic operations with big integers that can exceed \TeX 's number limits.

Contents

1	Documentation	2
1.1	Introduction	2
1.2	Conditions	2
1.2.1	Preconditions	2
1.2.2	Postconditions	3
1.3	Error handling	3
1.4	Operations	3
1.4.1	Num	3
1.4.2	Inv, Abs, Sgn	3
1.4.3	Min, Max, Cmp	4
1.4.4	Odd	5
1.4.5	Inc, Dec, Add, Sub	5
1.4.6	Shl, Shr	5
1.4.7	Mul, Sqr, Fac, Pow	6
1.4.8	Div, Mul	6
1.5	Interface for programmers	7
2	Implementation	8
2.1	Reload check and package identification	8
2.2	Catcodes	9
2.3	ε - \TeX detection	9
2.4	Help macros	9
2.5	Expand number	10
2.6	Normalize expanded number	11
2.7	Num	12
2.8	Inv, Abs, Sgn	12
2.9	Cmp, Min, Max	13
2.10	Odd	15
2.11	Inc, Dec	15
2.12	Add, Sub	18
2.13	Shl, Shr	24
2.14	\BIC@Tim	27
2.15	Mul	29
2.16	Sqr	30
2.17	Fac	31
2.18	Pow	32
2.18.1	Help macros	34
2.18.2	Recursive calculation	35

2.19	Div	36
2.20	Mod	42
3	Test	44
3.1	Catcode checks for loading	44
3.2	Macro tests	45
3.2.1	Preamble with test macro definitions	45
3.2.2	Time	49
3.2.3	Test sets	50
4	Installation	58
4.1	Download	58
4.2	Bundle installation	59
4.3	Package installation	59
4.4	Refresh file name databases	59
4.5	Some details for the interested	59
5	History	60
	[2007/09/27 v1.0]	60
	[2007/11/11 v1.1]	60
6	Index	60

1 Documentation

1.1 Introduction

Package `bigintcalc` defines arithmetic operations that deal with big integers. Big integers can be given either as explicit integer number or as macro code that expands to an explicit number. *Big* means that there is no limit on the size of the number. Big integers may exceed \TeX 's range limitation of -2147483647 and 2147483647. Only memory issues will limit the usable range.

In opposite to package `intcalc` unexpandable command tokens are not supported, even if they are valid \TeX numbers like count registers or commands created by `\chardef`. Nevertheless they may be used, if they are prefixed by `\number`.

Also ε - \TeX 's `\numexpr` expressions are not supported directly in the manner of package `intcalc`. However they can be given if `\the\numexpr` or `\number\numexpr` are used.

The operations have the form of macros that take one or two integers as parameter and return the integer result. The macro name is a three letter operation name prefixed by the package name, e.g. `\bigintcalcAdd{10}{43}` returns 53.

The macros are fully expandable, exactly two expansion steps generate the result. Therefore the operations may be used nearly everywhere in \TeX , even inside `\csname`, file names, or other expandable contexts.

1.2 Conditions

1.2.1 Preconditions

- Arguments can be anything that expands to a number that consists of optional signs and digits.
- The arguments and return values must be sound. Zero as divisor or factorials of negative numbers will cause errors.

1.2.2 Postconditions

Additional properties of the macros apart from calculating a correct result (of course ☺):

- The macros are fully expandable. Thus they can be used inside `\edef`, `\csname`, for example.
- Furthermore exactly two expansion steps calculate the result.
- The number consists of one optional minus sign and one or more digits. The first digit is larger than zero for numbers that consists of more than one digit.

In short, the number format is exactly the same as `\number` generates, but without its range limitation. And the tokens (minus sign, digits) have cat-code 12 (other).

- Call by value is simulated. First the arguments are converted to numbers. Then these numbers are used in the calculations.

Remember that arguments may contain expensive macros or ε -TeX expressions. This strategy avoids multiple evaluations of such arguments.

1.3 Error handling

Some errors are detected by the macros, example: division by zero. In this cases an undefined control sequence is called and causes a TeX error message, example: `\BigIntCalcError:DivisionByZero`. The name of the control sequence contains the reason for the error. The TeX error may be ignored. Then the operation returns zero as result. Because the macros are supposed to work in expandible contexts. An traditional error message, however, is not expandable and would break these contexts.

1.4 Operations

Some definition equations below use the function `Int` that converts a real number to an integer. The number is truncated that means rounding to zero:

$$\text{Int}(x) := \begin{cases} \lfloor x \rfloor & \text{if } x \geq 0 \\ \lceil x \rceil & \text{otherwise} \end{cases}$$

1.4.1 Num

`\bigintcalcNum {⟨x⟩}`

Macro `\bigintcalcNum` converts its argument to a normalized integer number without unnecessary leading zeros or signs. The result matches the regular expression:

`0|-?[1-9][0-9]*`

1.4.2 Inv, Abs, Sgn

`\bigintcalcInv {⟨x⟩}`

Macro `\bigintcalcInv` switches the sign.

$$\text{Inv}(x) := -x$$

`\bigintcalcAbs {⟨x⟩}`

Macro `\bigintcalcAbs` returns the absolute value of integer $\langle x \rangle$.

$$\text{Abs}(x) := |x|$$

`\bigintcalcSgn {⟨x⟩}`

Macro `\bigintcalcSgn` encodes the sign of $\langle x \rangle$ as number.

$$\text{Sgn}(x) := \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$$

These return values can easily be distinguished by `\ifcase`:

```
\ifcase\bigintcalcSgn{<x>}
  $x=0$
\or
  $x>0$
\else
  $x<0$
\fi
```

1.4.3 Min, Max, Cmp

`\bigintcalcMin {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcMin` returns the smaller of the two integers.

$$\text{Min}(x, y) := \begin{cases} x & \text{if } x < y \\ y & \text{otherwise} \end{cases}$$

`\bigintcalcMax {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcMax` returns the larger of the two integers.

$$\text{Max}(x, y) := \begin{cases} x & \text{if } x > y \\ y & \text{otherwise} \end{cases}$$

`\bigintcalcCmp {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcCmp` encodes the comparison result as number:

$$\text{Cmp}(x, y) := \begin{cases} -1 & \text{if } x < y \\ 0 & \text{if } x = y \\ 1 & \text{if } x > y \end{cases}$$

These values can be distinguished by `\ifcase`:

```
\ifcase\bigintcalcCmp{<x>}{<y>}
  $x=y$
\or
  $x>y$
\else
  $x<y$
\fi
```

1.4.4 Odd

`\bigintcalcOdd {⟨x⟩}`

$$\text{Odd}(x) := \begin{cases} 1 & \text{if } x \text{ is odd} \\ 0 & \text{if } x \text{ is even} \end{cases}$$

1.4.5 Inc, Dec, Add, Sub

`\bigintcalcInc {⟨x⟩}`

Macro `\bigintcalcInc` increments $\langle x \rangle$ by one.

$$\text{Inc}(x) := x + 1$$

`\bigintcalcDec {⟨x⟩}`

Macro `\bigintcalcDec` decrements $\langle x \rangle$ by one.

$$\text{Dec}(x) := x - 1$$

`\bigintcalcAdd {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcAdd` adds the two numbers.

$$\text{Add}(x, y) := x + y$$

`\bigintcalcSub {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcSub` calculates the difference.

$$\text{Sub}(x, y) := x - y$$

1.4.6 Shl, Shr

`\bigintcalcShl {⟨x⟩}`

Macro `\bigintcalcShl` implements shifting to the left that means the number is multiplied by two. The sign is preserved.

$$\text{Shl}(x) := x * 2$$

`\bigintcalcShr {⟨x⟩}`

Macro `\bigintcalcShr` implements shifting to the right. That is equivalent to an integer division by two. The sign is preserved.

$$\text{Shr}(x) := \text{Int}(x/2)$$

1.4.7 Mul, Sqr, Fac, Pow

`\bigintcalcMul {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcMul` calculates the product of $\langle x \rangle$ and $\langle y \rangle$.

$$\text{Mul}(x, y) := x * y$$

`\bigintcalcSqr {⟨x⟩}`

Macro `\bigintcalcSqr` returns the square product.

$$\text{Sqr}(x) := x^2$$

`\bigintcalcFac {⟨x⟩}`

Macro `\bigintcalcFac` returns the factorial of $\langle x \rangle$. Negative numbers are not permitted.

$$\text{Fac}(x) := x! \quad \text{for } x \geq 0$$

$$(0! = 1)$$

`\bigintcalcPow Mx My`

Macro `\bigintcalcPow` calculates the value of $\langle x \rangle$ to the power of $\langle y \rangle$. The error “division by zero” is thrown if $\langle x \rangle$ is zero and $\langle y \rangle$ is negative. permitted:

$$\text{Pow}(x, y) := \text{Int}(x^y) \quad \text{for } x \neq 0 \text{ or } y \geq 0$$

$$(0^0 = 1)$$

1.4.8 Div, Mod

`\bigintcalcDiv {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcDiv` performs an integer division. Argument $\langle y \rangle$ must not be zero.

$$\text{Div}(x, y) := \text{Int}(x/y) \quad \text{for } y \neq 0$$

`\bigintcalcMod {⟨x⟩} {⟨y⟩}`

Macro `\bigintcalcMod` gets the remainder of the integer division. The sign follows the divisor $\langle y \rangle$. Argument $\langle y \rangle$ must not be zero.

$$\text{Mod}(x, y) := x \% y \quad \text{for } y \neq 0$$

The result ranges:

$$-|y| < \text{Mod}(x, y) \leq 0 \quad \text{for } y < 0$$

$$0 \leq \text{Mod}(x, y) < y \quad \text{for } y \geq 0$$

1.5 Interface for programmers

If the programmer can ensure some more properties about the arguments of the operations, then the following macros are a little more efficient.

In general numbers must obey the following constraints:

- Plain number: digit tokens only, no command tokens.
- Non-negative. Signs are forbidden.
- Delimited by exclamation mark. Curly braces around the number are not allowed and will break the code.

`\BigIntCalcOdd $\langle number \rangle$!`

1/0 is returned if $\langle number \rangle$ is odd/even.

`\BigIntCalcInc $\langle number \rangle$!`

Incrementation.

`\BigIntCalcDec $\langle number \rangle$!`

Decrementation, positive number without zero.

`\BigIntCalcAdd $\langle number A \rangle$! $\langle number B \rangle$!`

Addition, $A \geq B$.

`\BigIntCalcSub $\langle number A \rangle$! $\langle number B \rangle$!`

Subtraction, $A \geq B$.

`\BigIntCalcShl $\langle number \rangle$!`

Left shift (multiplication with two).

`\BigIntCalcShr $\langle number \rangle$!`

Right shift (integer division by two).

`\BigIntCalcMul $\langle number A \rangle$! $\langle number B \rangle$!`

Multiplication, $A \geq B$.

`\BigIntCalcDiv $\langle number A \rangle$! $\langle number B \rangle$!`

Division operation.

`\BigIntCalcMod $\langle number A \rangle$! $\langle number B \rangle$!`

Modulo operation.

2 Implementation

```
1 ⟨*package⟩
```

2.1 Reload check and package identification

Reload check, especially if the package is not used with L^AT_EX.

```
2 \begingroup
3 \catcode44 12 % ,
4 \catcode45 12 % -
5 \catcode46 12 % .
6 \catcode58 12 % :
7 \catcode64 11 % @
8 \expandafter\let\expandafter\x\csname ver@bigintcalc.sty\endcsname
9 \ifcase 0%
10 \ifx\x\relax % plain
11 \else
12 \ifx\x\empty % LaTeX
13 \else
14 1%
15 \fi
16 \fi
17 \else
18 \catcode35 6 % #
19 \catcode123 1 % {
20 \catcode125 2 % }
21 \expandafter\ifx\csname PackageInfo\endcsname\relax
22 \def\x#1#2{%
23 \immediate\write-1{Package #1 Info: #2.}%
24 }%
25 \else
26 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27 \fi
28 \x{bigintcalc}{The package is already loaded}%
29 \endgroup
30 \expandafter\endinput
31 \fi
32 \endgroup
```

Package identification:

```
33 \begingroup
34 \catcode35 6 % #
35 \catcode40 12 % (
36 \catcode41 12 % )
37 \catcode44 12 % ,
38 \catcode45 12 % -
39 \catcode46 12 % .
40 \catcode47 12 % /
41 \catcode58 12 % :
42 \catcode64 11 % @
43 \catcode123 1 % {
44 \catcode125 2 % }
45 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
46 \def\x#1#2#3[#4]{\endgroup
47 \immediate\write-1{Package: #3 #4}%
48 \xdef#1{#4}%
49 }%
50 \else
51 \def\x#1#2[#3]{\endgroup
52 #2[#3]}%
53 \ifx#1\relax
54 \xdef#1{#3}%
55 \fi
56 }%
```



```

57 \fi
58 \expandafter\x\csname ver@bigintcalc.sty\endcsname
59 \ProvidesPackage{bigintcalc}%
60 [2007/11/11 v1.1 Expandable big integer calculations (H0)]

```

2.2 Catcodes

```

61 \begingroup
62 \catcode123 1 % {
63 \catcode125 2 % }
64 \def\x{\endgroup
65   \expandafter\edef\csname BIC@AtEnd\endcsname{%
66     \catcode35 \the\catcode35\relax
67     \catcode64 \the\catcode64\relax
68     \catcode123 \the\catcode123\relax
69     \catcode125 \the\catcode125\relax
70   }%
71 }%
72 \x
73 \catcode35 6 % #
74 \catcode64 11 % @
75 \catcode123 1 % {
76 \catcode125 2 % }
77 \def\TMP@EnsureCode#1#2{%
78   \edef\BIC@AtEnd{%
79     \BIC@AtEnd
80     \catcode#1 \the\catcode#1\relax
81   }%
82   \catcode#1 #2\relax
83 }
84 \TMP@EnsureCode{33}{12}% !
85 \TMP@EnsureCode{36}{14}% $ (comment!)
86 \TMP@EnsureCode{38}{14}% & (comment!)
87 \TMP@EnsureCode{40}{12}% (
88 \TMP@EnsureCode{41}{12}% )
89 \TMP@EnsureCode{42}{12}% *
90 \TMP@EnsureCode{43}{12}% +
91 \TMP@EnsureCode{45}{12}% -
92 \TMP@EnsureCode{46}{12}% .
93 \TMP@EnsureCode{47}{12}% /
94 \TMP@EnsureCode{58}{11}% : (letter!)
95 \TMP@EnsureCode{60}{12}% <
96 \TMP@EnsureCode{61}{12}% =
97 \TMP@EnsureCode{62}{12}% >
98 \TMP@EnsureCode{63}{14}% ? (comment!)
99 \begingroup\expandafter\expandafter\expandafter\endgroup
100 \expandafter\ifx\csname BIC@TestMode\endcsname\relax
101 \else
102   \catcode63=9 % ? (ignore)
103 \fi
104 ? \let\BIC@@TestMode\BIC@TestMode

```

2.3 ε -TeX detection

```

105 \begingroup\expandafter\expandafter\expandafter\endgroup
106 \expandafter\ifx\csname numexpr\endcsname\relax
107   \catcode36=9 % $ (ignore)
108 \else
109   \catcode38=9 % & (ignore)
110 \fi

```

2.4 Help macros

\BIC@Fi

```

111 \let\BIC@Fi\fi

\BIC@AfterFi

112 \def\BIC@AfterFi#1#2\BIC@Fi{\fi#1}%

\BIC@AfterFiFi

113 \def\BIC@AfterFiFi#1#2\BIC@Fi{\fi\fi#1}%

\BIC@AfterFiFiFi

114 \def\BIC@AfterFiFiFi#1#2\BIC@Fi{\fi\fi\fi#1}%

\BIC@Space

115 \begingroup
116 \def\x#1{\endgroup
117 \let\BIC@Space= #1%
118 }%
119 \x{ }

```

2.5 Expand number

```

120 \begingroup\expandafter\expandafter\expandafter\endgroup
121 \expandafter\ifx\csname RequirePackage\endcsname\relax
122 \input pdftexcmds.sty\relax
123 \else
124 \RequirePackage{pdftexcmds}[2007/11/11]%
125 \fi

126 \begingroup\expandafter\expandafter\expandafter\endgroup
127 \expandafter\ifx\csname pdf@escapehex\endcsname\relax

\BIC@Expand

128 \def\BIC@Expand#1{%
129 \romannumeral0%
130 \BIC@@Expand#1!\@nil{}}%
131 }%

\BIC@@Expand

132 \def\BIC@@Expand#1#2\@nil#3{%
133 \expandafter\ifcat\noexpand#1\relax
134 \expandafter\@firstoftwo
135 \else
136 \expandafter\@secondoftwo
137 \fi
138 {%
139 \expandafter\BIC@@Expand#1#2\@nil{#3}%
140 }{%
141 \ifx#1!%
142 \expandafter\@firstoftwo
143 \else
144 \expandafter\@secondoftwo
145 \fi
146 { #3}{%
147 \BIC@@Expand#2\@nil{#3#1}%
148 }%
149 }%
150 }%

\@firstoftwo

151 \expandafter\ifx\csname @firstoftwo\endcsname\relax
152 \long\def\@firstoftwo#1#2{#1}%
153 \fi

```

\@secondoftwo

```
154 \expandafter\ifx\csname @secondoftwo\endcsname\relax
155   \long\def\@secondoftwo#1#2{#2}%
156   \fi

157 \else
```

\BIC@Expand

```
158 \def\BIC@Expand#1{%
159   \romannumeral0\expandafter\expandafter\expandafter\BIC@Space
160   \pdf@unescapehex{%
161     \expandafter\expandafter\expandafter
162     \BIC@StripHexSpace\pdf@escapehex{#1}20\@nil
163   }%
164 }%
```

\BIC@StripHexSpace

```
165 \def\BIC@StripHexSpace#120#2\@nil{%
166   #1%
167   \ifx\#2\%
168   \else
169     \BIC@AfterFi{%
170       \BIC@StripHexSpace#2\@nil
171     }%
172   \BIC@Fi
173   }%

174 \fi
```

2.6 Normalize expanded number

\BIC@Normalize #1: result sign

#2: first token of number

```
175 \def\BIC@Normalize#1#2{%
176   \ifx#2-%
177     \ifx\#1\%
178       \BIC@AfterFiFi{%
179         \BIC@Normalize-%
180       }%
181     \else
182       \BIC@AfterFiFi{%
183         \BIC@Normalize{}%
184       }%
185     \fi
186   \else
187     \ifx#2+%
188       \BIC@AfterFiFi{%
189         \BIC@Normalize{#1}%
190       }%
191     \else
192       \ifx#20%
193         \BIC@AfterFiFiFi{%
194           \BIC@NormalizeZero{#1}%
195         }%
196       \else
197         \BIC@AfterFiFiFi{%
198           \BIC@NormalizeDigits#1#2%
199         }%
200       \fi
201     \fi
202   \BIC@Fi
203 }
```

\BIC@NormalizeZero

```
204 \def\BIC@NormalizeZero#1#2{%
205   \ifx#2!%
206     \BIC@AfterFi{ 0}%
207   \else
208     \ifx#20%
209       \BIC@AfterFiFi{%
210         \BIC@NormalizeZero{#1}%
211       }%
212     \else
213       \BIC@AfterFiFi{%
214         \BIC@NormalizeDigits#1#2%
215       }%
216     \fi
217   \BIC@Fi
218 }
```

\BIC@NormalizeDigits

```
219 \def\BIC@NormalizeDigits#1!{ #1}
```

2.7 Num

\bigintcalcNum

```
220 \def\bigintcalcNum#1{%
221   \romannumeral0%
222   \expandafter\expandafter\expandafter\BIC@Normalize
223   \expandafter\expandafter\expandafter{%
224     \expandafter\expandafter\expandafter}%
225   \BIC@Expand{#1}!%
226 }
```

2.8 Inv, Abs, Sgn

\bigintcalcInv

```
227 \def\bigintcalcInv#1{%
228   \romannumeral0\expandafter\expandafter\expandafter\BIC@Space
229   \bigintcalcNum{-#1}%
230 }
```

\bigintcalcAbs

```
231 \def\bigintcalcAbs#1{%
232   \romannumeral0%
233   \expandafter\expandafter\expandafter\BIC@Abs
234   \bigintcalcNum{#1}%
235 }
```

\BIC@Abs

```
236 \def\BIC@Abs#1{%
237   \ifx#1-%
238     \expandafter\BIC@Space
239   \else
240     \expandafter\BIC@Space
241     \expandafter#1%
242   \fi
243 }
```

\bigintcalcSgn

```
244 \def\bigintcalcSgn#1{%
245   \number
246   \expandafter\expandafter\expandafter\BIC@Sgn
```

```

247 \bigintcalcNum{#1}! %
248 }

```

\BIC@Sgn

```

249 \def\BIC@Sgn#1#2!{%
250 \ifx#1-%
251 -1%
252 \else
253 \ifx#10%
254 0%
255 \else
256 1%
257 \fi
258 \fi
259 }

```

2.9 Cmp, Min, Max

\bigintcalcCmp

```

260 \def\bigintcalcCmp#1#2{%
261 \number
262 \expandafter\expandafter\expandafter\BIC@Cmp
263 \bigintcalcNum{#2}!{#1}%
264 }

```

\BIC@Cmp

```

265 \def\BIC@Cmp#1!#2{%
266 \expandafter\expandafter\expandafter\BIC@@Cmp
267 \bigintcalcNum{#2}!#1!%
268 }

```

\BIC@@Cmp

```

269 \def\BIC@@Cmp#1#2!#3#4!{%
270 \ifx#1-%
271 \ifx#3-%
272 \BIC@AfterFiFi{%
273 \BIC@@Cmp#4!#2!%
274 }%
275 \else
276 \BIC@AfterFiFi{%
277 -1 %
278 }%
279 \fi
280 \else
281 \ifx#3-%
282 \BIC@AfterFiFi{%
283 1 %
284 }%
285 \else
286 \BIC@AfterFiFi{%
287 \BIC@CmpLength#1#2!#3#4!#1#2!#3#4!%
288 }%
289 \fi
290 \BIC@Fi
291 }

```

\BIC@PosCmp

```

292 \def\BIC@PosCmp#1!#2!{%
293 \BIC@CmpLength#1!#2!#1!#2!%
294 }

```

\BIC@CmpLength

```
295 \def\BIC@CmpLength#1#2!#3#4!{%
296   \ifx\#2\%
297     \ifx\#4\%
298       \BIC@AfterFiFi\BIC@CmpDiff
299     \else
300       \BIC@AfterFiFi{%
301         \BIC@CmpResult{-1}%
302       }%
303     \fi
304   \else
305     \ifx\#4\%
306       \BIC@AfterFiFi{%
307         \BIC@CmpResult1%
308       }%
309     \else
310       \BIC@AfterFiFi{%
311         \BIC@CmpLength#2!#4!%
312       }%
313     \fi
314   \BIC@Fi
315 }
```

\BIC@CmpResult

```
316 \def\BIC@CmpResult#1#2!#3!{#1 }
```

\BIC@CmpDiff

```
317 \def\BIC@CmpDiff#1#2!#3#4!{%
318   \ifnum#1<#3 %
319     \BIC@AfterFi{%
320       -1 %
321     }%
322   \else
323     \ifnum#1>#3 %
324       \BIC@AfterFiFi{%
325         1 %
326       }%
327     \else
328       \ifx\#2\%
329         \BIC@AfterFiFiFi{%
330           0 %
331         }%
332       \else
333         \BIC@AfterFiFiFi{%
334           \BIC@CmpDiff#2!#4!%
335         }%
336       \fi
337     \fi
338   \BIC@Fi
339 }
```

\bigintcalcMin

```
340 \def\bigintcalcMin#1{%
341   \romannumeral0%
342   \expandafter\expandafter\expandafter\BIC@MinMax
343   \bigintcalcNum{#1}!-!%
344 }
```

\bigintcalcMax

```
345 \def\bigintcalcMax#1{%
346   \romannumeral0%
```

```

347 \expandafter\expandafter\expandafter\BIC@MinMax
348 \bigintcalcNum{#1}!!%
349 }

\BIC@MinMax #1:  $x$ 
#2: sign for comparison
#3:  $y$ 
350 \def\BIC@MinMax#1!#2!#3{%
351 \expandafter\expandafter\expandafter\BIC@@MinMax
352 \bigintcalcNum{#3}!#1!#2!%
353 }

\BIC@@MinMax #1:  $y$ 
#2:  $x$ 
#3: sign for comparison
354 \def\BIC@@MinMax#1!#2!#3!{%
355 \ifnum\BIC@@Cmp#1!#2!=#31 %
356 \BIC@AfterFi{ #1}%
357 \else
358 \BIC@AfterFi{ #2}%
359 \BIC@Fi
360 }

```

2.10 Odd

```

\bigintcalcOdd
361 \def\bigintcalcOdd#1{%
362 \romannumeral0%
363 \expandafter\expandafter\expandafter\BIC@Odd
364 \bigintcalcAbs{#1}!%
365 }

\BigIntCalcOdd
366 \def\BigIntCalcOdd#1!{%
367 \romannumeral0%
368 \BIC@Odd#1!%
369 }

\BIC@Odd #1:  $x$ 
370 \def\BIC@Odd#1#2{%
371 \ifx#2!%
372 \ifodd#1 %
373 \BIC@AfterFiFi{ 1}%
374 \else
375 \BIC@AfterFiFi{ 0}%
376 \fi
377 \else
378 \expandafter\BIC@Odd\expandafter#2%
379 \BIC@Fi
380 }

```

2.11 Inc, Dec

```

\bigintcalcInc
381 \def\bigintcalcInc#1{%
382 \romannumeral0%
383 \expandafter\expandafter\expandafter\BIC@IncSwitch
384 \bigintcalcNum{#1}!%
385 }

```

\BIC@IncSwitch

```
386 \def\BIC@IncSwitch#1#2!{%
387   \ifcase\BIC@@Cmp#1#2!-1!%
388     \BIC@AfterFi{ 0}%
389   \or
390     \BIC@AfterFi{%
391       \BIC@Inc#1#2!{%}%
392     }%
393   \else
394     \BIC@AfterFi{%
395       \expandafter-\romannumeral0%
396       \BIC@Dec#2!{%}%
397     }%
398   \BIC@Fi
399 }
```

\bigintcalcDec

```
400 \def\bigintcalcDec#1{%
401   \romannumeral0%
402   \expandafter\expandafter\expandafter\BIC@DecSwitch
403   \bigintcalcNum{#1}!%
404 }
```

\BIC@DecSwitch

```
405 \def\BIC@DecSwitch#1#2!{%
406   \ifcase\BIC@Sgn#1#2! %
407     \BIC@AfterFi{ -1}%
408   \or
409     \BIC@AfterFi{%
410       \BIC@Dec#1#2!{%}%
411     }%
412   \else
413     \BIC@AfterFi{%
414       \expandafter-\romannumeral0%
415       \BIC@Inc#2!{%}%
416     }%
417   \BIC@Fi
418 }
```

\BigIntCalcInc

```
419 \def\BigIntCalcInc#1!{%
420   \romannumeral0\BIC@Inc#1!{%}%
421 }
```

\BigIntCalcDec

```
422 \def\BigIntCalcDec#1!{%
423   \romannumeral0\BIC@Dec#1!{%}%
424 }
```

\BIC@Inc

```
425 \def\BIC@Inc#1#2!#3{%
426   \ifx\#2\%
427     \BIC@AfterFi{%
428       \BIC@@Inc1#1#3!{%}%
429     }%
430   \else
431     \BIC@AfterFi{%
432       \BIC@Inc#2!{#1#3}%
433     }%
434   \BIC@Fi
435 }
```


\BIC@@Inc

```
436 \def\BIC@@Inc#1#2#3!#4{%
437   \ifcase#1 %
438     \ifx\\#3\\%
439       \BIC@AfterFiFi{ #2#4}%
440     \else
441       \BIC@AfterFiFi{%
442         \BIC@@Inc0#3!{#2#4}%
443       }%
444     \fi
445   \else
446     \ifnum#2<9 %
447       \BIC@AfterFiFi{%
448 &       \expandafter\BIC@@@Inc\the\numexpr#2+1\relax
449 $       \expandafter\expandafter\expandafter\BIC@@@Inc
450 $       \ifcase#2 \expandafter1%
451 $       \or\expandafter2%
452 $       \or\expandafter3%
453 $       \or\expandafter4%
454 $       \or\expandafter5%
455 $       \or\expandafter6%
456 $       \or\expandafter7%
457 $       \or\expandafter8%
458 $       \or\expandafter9%
459 $?     \else\BigIntCalcError:ThisCannotHappen%
460 $       \fi
461       0#3!{#4}%
462     }%
463   \else
464     \BIC@AfterFiFi{%
465       \BIC@@@Inc01#3!{#4}%
466     }%
467   \fi
468 \BIC@Fi
469 }
```

\BIC@@@Inc

```
470 \def\BIC@@@Inc#1#2#3!#4{%
471   \ifx\\#3\\%
472     \ifnum#2=1 %
473       \BIC@AfterFiFi{ 1#1#4}%
474     \else
475       \BIC@AfterFiFi{ #1#4}%
476     \fi
477   \else
478     \BIC@AfterFi{%
479       \BIC@@Inc#2#3!{#1#4}%
480     }%
481   \BIC@Fi
482 }
```

\BIC@Dec

```
483 \def\BIC@Dec#1#2!#3{%
484   \ifx\\#2\\%
485     \BIC@AfterFi{%
486       \BIC@@Dec1#1#3!{ }%
487     }%
488   \else
489     \BIC@AfterFi{%
490       \BIC@Dec#2!{#1#3}%
491     }%
492   \BIC@Fi
```

```

493 }

\BIC@@Dec

494 \def\BIC@@Dec#1#2#3!#4{%
495   \ifcase#1 %
496     \ifx\#3\%
497       \BIC@AfterFiFi{ #2#4}%
498     \else
499       \BIC@AfterFiFi{%
500         \BIC@@Dec0#3!{#2#4}%
501       }%
502     \fi
503   \else
504     \ifnum#2>0 %
505       \BIC@AfterFiFi{%
506         & \expandafter\BIC@@Dec\the\numexpr#2-1\relax
507         $ \expandafter\expandafter\expandafter\BIC@@Dec
508         $ \ifcase#2
509         $? \BigIntCalcError:ThisCannotHappen%
510         $ \or\expandafter0%
511         $ \or\expandafter1%
512         $ \or\expandafter2%
513         $ \or\expandafter3%
514         $ \or\expandafter4%
515         $ \or\expandafter5%
516         $ \or\expandafter6%
517         $ \or\expandafter7%
518         $ \or\expandafter8%
519         $? \else\BigIntCalcError:ThisCannotHappen%
520         $ \fi
521         0#3!{#4}%
522       }%
523     \else
524       \BIC@AfterFiFi{%
525         \BIC@@Dec91#3!{#4}%
526       }%
527     \fi
528   \BIC@Fi
529 }

\BIC@@@Dec

530 \def\BIC@@@Dec#1#2#3!#4{%
531   \ifx\#3\%
532     \ifcase#1 %
533       \ifx\#4\%
534         \BIC@AfterFiFiFi{ 0}%
535       \else
536         \BIC@AfterFiFiFi{ #4}%
537       \fi
538     \else
539       \BIC@AfterFiFi{ #1#4}%
540     \fi
541   \else
542     \BIC@AfterFi{%
543       \BIC@@Dec#2#3!{#1#4}%
544     }%
545   \BIC@Fi
546 }

```

2.12 Add, Sub

\bigintcalcAdd

```

547 \def\bigintcalcAdd#1{%
548   \romannumeral0%
549   \expandafter\expandafter\expandafter\BIC@Add
550   \bigintcalcNum{#1}!%
551 }

```

\BIC@Add

```

552 \def\BIC@Add#1!#2{%
553   \expandafter\expandafter\expandafter
554   \BIC@AddSwitch\bigintcalcNum{#2}!#1!%
555 }

```

\bigintcalcSub

```

556 \def\bigintcalcSub#1#2{%
557   \romannumeral0%
558   \expandafter\expandafter\expandafter\BIC@Add
559   \bigintcalcNum{-#2}!{#1}%
560 }

```

\BIC@AddSwitch Decision table for \BIC@AddSwitch.

$x < 0$	$y < 0$	$-x > -y$	–	$\text{Add}(-x, -y)$
		else		$\text{Add}(-y, -x)$
	else	$-x > y$	–	$\text{Sub}(-x, y)$
		$-x = y$		0
		else	+	$\text{Sub}(y, -x)$
else	$y < 0$	$x > -y$	+	$\text{Sub}(x, -y)$
		$x = -y$		0
		else	–	$\text{Sub}(-y, x)$
	else	$x > y$	+	$\text{Add}(x, y)$
		else		$\text{Add}(y, x)$

```

561 \def\BIC@AddSwitch#1#2!#3#4!{%
562   \ifx#1-% x < 0
563     \ifx#3-% y < 0
564       \expandafter-\romannumeral0%
565       \ifnum\BIC@PosCmp#2!#4!=1 % -x > -y
566         \BIC@AfterFiFiFi{%
567           \BIC@AddXY#2!#4!!!%
568         }%
569       \else % -x <= -y
570         \BIC@AfterFiFiFi{%
571           \BIC@AddXY#4!#2!!!%
572         }%
573       \fi
574     \else % y >= 0
575       \ifcase\BIC@PosCmp#2!#3#4!% -x = y
576         \BIC@AfterFiFiFi{ 0}%
577       \or % -x > y
578         \expandafter-\romannumeral0%
579         \BIC@AfterFiFiFi{%
580           \BIC@SubXY#2!#3#4!!!%
581         }%
582       \else % -x <= y
583         \BIC@AfterFiFiFi{%
584           \BIC@SubXY#3#4!#2!!!%
585         }%
586       \fi
587     \fi
588   \else % x >= 0
589     \ifx#3-% y < 0

```

```

590     \ifcase\BIC@PosCmp#1#2!#4!% x = -y
591     \BIC@AfterFiFiFi{ 0}%
592     \or % x > -y
593     \BIC@AfterFiFiFi{%
594     \BIC@SubXY#1#2!#4!!!%
595     }%
596     \else % x <= -y
597     \expandafter-\romannumeral0%
598     \BIC@AfterFiFiFi{%
599     \BIC@SubXY#4!#1#2!!!%
600     }%
601     \fi
602     \else % y >= 0
603     \ifnum\BIC@PosCmp#1#2!#3#4!=1 % x > y
604     \BIC@AfterFiFiFi{%
605     \BIC@AddXY#1#2!#3#4!!!%
606     }%
607     \else % x <= y
608     \BIC@AfterFiFiFi{%
609     \BIC@AddXY#3#4!#1#2!!!%
610     }%
611     \fi
612     \fi
613     \BIC@Fi
614 }

\BigIntCalcAdd

615 \def\BigIntCalcAdd#1!#2!{%
616     \romannumeral0\BIC@AddXY#1!#2!!!%
617 }

\BigIntCalcSub

618 \def\BigIntCalcSub#1!#2!{%
619     \romannumeral0\BIC@SubXY#1!#2!!!%
620 }

\BIC@AddXY

621 \def\BIC@AddXY#1#2!#3#4!#5!#6!{%
622     \ifx\#2\%
623     \ifx\#3\%
624     \BIC@AfterFiFiFi{%
625     \BIC@DoAdd0!#1#5!#60!%
626     }%
627     \else
628     \BIC@AfterFiFiFi{%
629     \BIC@DoAdd0!#1#5!#3#6!%
630     }%
631     \fi
632     \else
633     \ifx\#4\%
634     \ifx\#3\%
635     \BIC@AfterFiFiFi{%
636     \BIC@AddXY#2!{}!#1#5!#60!%
637     }%
638     \else
639     \BIC@AfterFiFiFi{%
640     \BIC@AddXY#2!{}!#1#5!#3#6!%
641     }%
642     \fi
643     \else
644     \BIC@AfterFiFiFi{%
645     \BIC@AddXY#2!#4!#1#5!#3#6!%

```

```

646     }%
647     \fi
648     \BIC@Fi
649 }

\BIC@DoAdd #1: carry
#2: reverted result
#3#4: reverted  $x$ 
#5#6: reverted  $y$ 
650 \def\BIC@DoAdd#1#2!#3#4!#5#6!{%
651     \ifx\#4\%
652         \BIC@AfterFi{%
653             & \expandafter\BIC@Space
654             & \the\numexpr#1+#3+#5\relax#2%
655             $ \expandafter\expandafter\expandafter\BIC@AddResult
656             $ \BIC@AddDigit#1#3#5#2%
657         }%
658     \else
659         \BIC@AfterFi{%
660             \expandafter\expandafter\expandafter\BIC@DoAdd
661             \BIC@AddDigit#1#3#5#2!#4!#6!%
662         }%
663     \BIC@Fi
664 }

\BIC@AddResult
665 $ \def\BIC@AddResult#1{%
666 $ \ifx#10%
667 $ \expandafter\BIC@Space
668 $ \else
669 $ \expandafter\BIC@Space\expandafter#1%
670 $ \fi
671 $ }%

\BIC@AddDigit #1: carry
#2: digit of  $x$ 
#3: digit of  $y$ 
672 \def\BIC@AddDigit#1#2#3{%
673     \romannumeral0%
674 & \expandafter\BIC@@AddDigit\the\numexpr#1+#2+#3!%
675 $ \expandafter\BIC@@AddDigit\number%
676 $ \csname
677 $ BIC@AddCarry%
678 $ \ifcase#1 %
679 $ #2%
680 $ \else
681 $ \ifcase#2 1\or2\or3\or4\or5\or6\or7\or8\or9\or10\fi
682 $ \fi
683 $ \endcsname#3!%
684 }

\BIC@@AddDigit
685 \def\BIC@@AddDigit#1!{%
686     \ifnum#1<10 %
687         \BIC@AfterFi{ 0#1}%
688     \else
689         \BIC@AfterFi{ #1}%
690     \BIC@Fi
691 }

\BIC@AddCarry0
692 $ \expandafter\def\csname BIC@AddCarry0\endcsname#1{#1}%

```

\BIC@AddCarry10

```
693 $ \expandafter\def\csname BIC@AddCarry10\endcsname#1{#1}%
```

\BIC@AddCarry[1-9]

```
694 $ \def\BIC@Temp#1#2{%
695 $   \expandafter\def\csname BIC@AddCarry#1\endcsname##1{%
696 $     \ifcase##1 #1\or
697 $       #2%
698 $?   \else\BigIntCalcError:ThisCannotHappen%
699 $     \fi
700 $   }%
701 $ }%
702 $ \BIC@Temp 0{1\or2\or3\or4\or5\or6\or7\or8\or9}%
703 $ \BIC@Temp 1{2\or3\or4\or5\or6\or7\or8\or9\or10}%
704 $ \BIC@Temp 2{3\or4\or5\or6\or7\or8\or9\or10\or11}%
705 $ \BIC@Temp 3{4\or5\or6\or7\or8\or9\or10\or11\or12}%
706 $ \BIC@Temp 4{5\or6\or7\or8\or9\or10\or11\or12\or13}%
707 $ \BIC@Temp 5{6\or7\or8\or9\or10\or11\or12\or13\or14}%
708 $ \BIC@Temp 6{7\or8\or9\or10\or11\or12\or13\or14\or15}%
709 $ \BIC@Temp 7{8\or9\or10\or11\or12\or13\or14\or15\or16}%
710 $ \BIC@Temp 8{9\or10\or11\or12\or13\or14\or15\or16\or17}%
711 $ \BIC@Temp 9{10\or11\or12\or13\or14\or15\or16\or17\or18}%
```

\BIC@SubXY Preconditions:

- $x > y$, $x \geq 0$, and $y \geq 0$
- $\text{digits}(x) = \text{digits}(y)$

```
712 \def\BIC@SubXY#1#2!#3#4!#5!#6!{%
713   \ifx\#2\%
714     \ifx\#3\%
715       \BIC@AfterFiFi{%
716         \BIC@DoSub0!#1#5!#60!%
717       }%
718     \else
719       \BIC@AfterFiFi{%
720         \BIC@DoSub0!#1#5!#3#6!%
721       }%
722     \fi
723   \else
724     \ifx\#4\%
725       \ifx\#3\%
726         \BIC@AfterFiFiFi{%
727           \BIC@SubXY#2!#1#5!#60!%
728         }%
729       \else
730         \BIC@AfterFiFiFi{%
731           \BIC@SubXY#2!#1#5!#3#6!%
732         }%
733       \fi
734     \else
735       \BIC@AfterFiFiFi{%
736         \BIC@SubXY#2!#4!#1#5!#3#6!%
737       }%
738     \fi
739   \BIC@Fi
740 }
```

\BIC@DoSub #1: carry

#2: reverted result

#3#4: reverted x

#5#6: reverted y

```

741 \def\BIC@DoSub#1#2!#3#4!#5#6!{%
742   \ifx\#4\%
743     \BIC@AfterFi{%
744       \expandafter\expandafter\expandafter\BIC@SubResult
745       \BIC@SubDigit#1#3#5#2%
746     }%
747   \else
748     \BIC@AfterFi{%
749       \expandafter\expandafter\expandafter\BIC@DoSub
750       \BIC@SubDigit#1#3#5#2!#4!#6!%
751     }%
752   \BIC@Fi
753 }

```

\BIC@SubResult

```

754 \def\BIC@SubResult#1{%
755   \ifx#10%
756     \expandafter\BIC@SubResult
757   \else
758     \expandafter\BIC@Space\expandafter#1%
759   \fi
760 }

```

\BIC@SubDigit #1: carry
#2: digit of x
#3: digit of y

```

761 \def\BIC@SubDigit#1#2#3{%
762   \romannumeral0%
763 & \expandafter\BIC@@SubDigit\the\numexpr#2-#3-#1!%
764 $ \expandafter\BIC@@AddDigit\number
765 $ \csname
766 $   BIC@SubCarry%
767 $   \ifcase#1 %
768 $     #3%
769 $   \else
770 $     \ifcase#3 1\or2\or3\or4\or5\or6\or7\or8\or9\or10\fi
771 $   \fi
772 $   \endcsname#2!%
773 }

```

\BIC@@SubDigit

```

774 & \def\BIC@@SubDigit#1!{%
775 &   \ifnum#1<0 %
776 &     \BIC@AfterFi{%
777 &       \expandafter\BIC@Space
778 &       \expandafter1\the\numexpr#1+10\relax
779 &     }%
780 &   \else
781 &     \BIC@AfterFi{ 0#1}%
782 &   \BIC@Fi
783 & }%

```

\BIC@SubCarry0

```

784 $ \expandafter\def\csname BIC@SubCarry0\endcsname#1{#1}%

```

\BIC@SubCarry10

```

785 $ \expandafter\def\csname BIC@SubCarry10\endcsname#1{1#1}%

```

\BIC@SubCarry[1-9]

```

786 $ \def\BIC@Temp#1#2{%
787 $   \expandafter\def\csname BIC@SubCarry#1\endcsname##1{%

```

```

788 $ \ifcase##1 #2%
789 $? \else\BigIntCalcError:ThisCannotHappen%
790 $ \fi
791 $ }%
792 $ }%
793 $ \BIC@Temp 1{19\or0\or1\or2\or3\or4\or5\or6\or7\or8}%
794 $ \BIC@Temp 2{18\or19\or0\or1\or2\or3\or4\or5\or6\or7}%
795 $ \BIC@Temp 3{17\or18\or19\or0\or1\or2\or3\or4\or5\or6}%
796 $ \BIC@Temp 4{16\or17\or18\or19\or0\or1\or2\or3\or4\or5}%
797 $ \BIC@Temp 5{15\or16\or17\or18\or19\or0\or1\or2\or3\or4}%
798 $ \BIC@Temp 6{14\or15\or16\or17\or18\or19\or0\or1\or2\or3}%
799 $ \BIC@Temp 7{13\or14\or15\or16\or17\or18\or19\or0\or1\or2}%
800 $ \BIC@Temp 8{12\or13\or14\or15\or16\or17\or18\or19\or0\or1}%
801 $ \BIC@Temp 9{11\or12\or13\or14\or15\or16\or17\or18\or19\or0}%

```

2.13 Shl, Shr

\bigintcalcShl

```

802 \def\bigintcalcShl#1{%
803 \romannumeral0%
804 \expandafter\expandafter\expandafter\BIC@Shl
805 \bigintcalcNum{#1}!%
806 }

```

\BIC@Shl

```

807 \def\BIC@Shl#1#2!{%
808 \ifx#1-%
809 \BIC@AfterFi{%
810 \expandafter-\romannumeral0%
811 & \BIC@@Shl#2!!%
812 $ \BIC@AddXY#2!#2!!!%
813 }%
814 \else
815 \BIC@AfterFi{%
816 & \BIC@@Shl#1#2!!%
817 $ \BIC@AddXY#1#2!#1#2!!!%
818 }%
819 \BIC@Fi
820 }

```

\BigIntCalcShl

```

821 \def\BigIntCalcShl#1!{%
822 \romannumeral0%
823 & \BIC@@Shl#1!!%
824 $ \BIC@AddXY#1!#1!!!%
825 }

```

\BIC@@Shl

```

826 & \def\BIC@@Shl#1#2!{%
827 & \ifx\#2\%
828 & \BIC@AfterFi{%
829 & \BIC@@@Shl0!#1%
830 & }%
831 & \else
832 & \BIC@AfterFi{%
833 & \BIC@@Shl#2!#1%
834 & }%
835 & \BIC@Fi
836 & }%

```



```

\BIC@@@Shl #1: carry
#2: result
#3#4: reverted number
837 & \def\BIC@@@Shl#1#2!#3#4!{%
838 & \ifx\#4\%
839 & \BIC@AfterFi{%
840 & \expandafter\BIC@Space
841 & \the\numexpr#3*2+#1\relax#2%
842 & }%
843 & \else
844 & \BIC@AfterFi{%
845 & \expandafter\BIC@@@Shl\the\numexpr#3*2+#1!#2!#4!%
846 & }%
847 & \BIC@Fi
848 & }%

\BIC@@@Shl
849 & \def\BIC@@@Shl#1!{%
850 & \ifnum#1<10 %
851 & \BIC@AfterFi{%
852 & \BIC@@@Shl0#1%
853 & }%
854 & \else
855 & \BIC@AfterFi{%
856 & \BIC@@@Shl#1%
857 & }%
858 & \BIC@Fi
859 & }%

\bigintcalcShr
860 \def\bigintcalcShr#1{%
861 \romannumeral0%
862 \expandafter\expandafter\expandafter\BIC@Shr
863 \bigintcalcNum{#1}!%
864 }

\BIC@Shr
865 \def\BIC@Shr#1#2!{%
866 \ifx#1-%
867 \expandafter-\romannumeral0%
868 \BIC@AfterFi{%
869 \BIC@@Shr#2!%
870 }%
871 \else
872 \BIC@AfterFi{%
873 \BIC@@Shr#1#2!%
874 }%
875 \BIC@Fi
876 }

\BigIntCalcShr
877 \def\BigIntCalcShr#1!{%
878 \romannumeral0%
879 \BIC@@Shr#1!%
880 }

\BIC@@Shr
881 \def\BIC@@Shr#1#2!{%
882 \ifcase#1 %
883 \BIC@AfterFi{ 0}%
884 \or

```

```

885     \ifx\|#2\%
886     \BIC@AfterFiFi{ 0}%
887     \else
888     \BIC@AfterFiFi{%
889     \BIC@@@Shr#1#2!!%
890     }%
891     \fi
892     \else
893     \BIC@AfterFiFi{%
894     \BIC@@@Shr0#1#2!!%
895     }%
896     \BIC@Fi
897 }

\BIC@@@Shr #1: carry
#2#3: number
#4: result

898 \def\BIC@@@Shr#1#2#3!#4!{%
899     \ifx\|#3\%
900     \ifodd#1#2 %
901     \BIC@AfterFiFi{%
902     & \expandafter\BIC@ShrResult\the\numexpr(#1#2-1)/2\relax
903     $ \expandafter\expandafter\expandafter\BIC@ShrResult
904     $ \csname BIC@ShrDigit#1#2\endcsname
905     #4!%
906     }%
907     \else
908     \BIC@AfterFiFi{%
909     & \expandafter\BIC@ShrResult\the\numexpr#1#2/2\relax
910     $ \expandafter\expandafter\expandafter\BIC@ShrResult
911     $ \csname BIC@ShrDigit#1#2\endcsname
912     #4!%
913     }%
914     \fi
915     \else
916     \ifodd#1#2 %
917     \BIC@AfterFiFi{%
918     & \expandafter\BIC@@@Shr\the\numexpr(#1#2-1)/2\relax1%
919     $ \expandafter\expandafter\expandafter\BIC@@@Shr
920     $ \csname BIC@ShrDigit#1#2\endcsname
921     #3!#4!%
922     }%
923     \else
924     \BIC@AfterFiFi{%
925     & \expandafter\BIC@@@Shr\the\numexpr#1#2/2\relax0%
926     $ \expandafter\expandafter\expandafter\BIC@@@Shr
927     $ \csname BIC@ShrDigit#1#2\endcsname
928     #3!#4!%
929     }%
930     \fi
931     \BIC@Fi
932 }

\BIC@ShrResult

933 & \def\BIC@ShrResult#1#2!{ #2#1}%
934 $ \def\BIC@ShrResult#1#2#3!{ #3#1}%

\BIC@@@Shr #1: new digit
#2: carry
#3: remaining number
#4: result

935 \def\BIC@@@Shr#1#2#3!#4!{%

```

```

936 \BIC@@@Shr#2#3!#4#1!%
937 }

```

\BIC@ShrDigit[00-19]

```

938 $ \def\BIC@Temp#1#2#3#4{%
939 $ \expandafter\def\csname BIC@ShrDigit#1#2\endcsname{#3#4}%
940 $ }%
941 $ \BIC@Temp 0000%
942 $ \BIC@Temp 0101%
943 $ \BIC@Temp 0210%
944 $ \BIC@Temp 0311%
945 $ \BIC@Temp 0420%
946 $ \BIC@Temp 0521%
947 $ \BIC@Temp 0630%
948 $ \BIC@Temp 0731%
949 $ \BIC@Temp 0840%
950 $ \BIC@Temp 0941%
951 $ \BIC@Temp 1050%
952 $ \BIC@Temp 1151%
953 $ \BIC@Temp 1260%
954 $ \BIC@Temp 1361%
955 $ \BIC@Temp 1470%
956 $ \BIC@Temp 1571%
957 $ \BIC@Temp 1680%
958 $ \BIC@Temp 1781%
959 $ \BIC@Temp 1890%
960 $ \BIC@Temp 1991%

```

2.14 \BIC@Tim

\BIC@Tim Macro \BIC@Tim implements “Number *times* digit”.
#1: plain number without sign
#2: digit

\BIC@@Tim #1#2: number
#3: reverted number

```

961 \def\BIC@@Tim#1#2!{%
962 \ifx\#2\%
963 \BIC@AfterFi{%
964 \BIC@ProcessTim0!#1%
965 }%
966 \else
967 \BIC@AfterFi{%
968 \BIC@@Tim#2!#1%
969 }%
970 \BIC@Fi
971 }

```

\BIC@ProcessTim #1: carry
#2: result
#3#4: reverted number
#5: digit

```

972 \def\BIC@ProcessTim#1#2!#3#4!#5{%
973 \ifx\#4\%
974 \BIC@AfterFi{%
975 \expandafter\BIC@Space
976 & \the\numexpr#3*#5+#1\relax
977 $ \romannumeral0\BIC@TimDigit#3#5#1%
978 #2%
979 }%
980 \else

```

```

981     \BIC@AfterFi{%
982     \expandafter\BIC@@ProcessTim
983 &     \the\numexpr#3*#5+#1%
984 $     \romannumeral0\BIC@TimDigit#3#5#1%
985     !#2!#4!#5%
986     }%
987 \BIC@Fi
988 }

\BIC@@ProcessTim #1#2: carry?, new digit
#3: new number
#4: old number
#5: digit
989 \def\BIC@@ProcessTim#1#2!{%
990     \ifx\#2\%
991     \BIC@AfterFi{%
992     \BIC@ProcessTim0#1%
993     }%
994 \else
995     \BIC@AfterFi{%
996     \BIC@ProcessTim#1#2%
997     }%
998 \BIC@Fi
999 }

\BIC@TimDigit #1: digit 0-9
#2: digit 3-9
#3: carry 0-9
1000 $ \def\BIC@TimDigit#1#2#3{%
1001 $     \ifcase#1 % 0
1002 $     \BIC@AfterFi{ #3}%
1003 $     \or % 1
1004 $     \BIC@AfterFi{%
1005 $         \expandafter\BIC@Space
1006 $         \number\csname BIC@AddCarry#2\endcsname#3 %
1007 $     }%
1008 $     \else
1009 $     \ifcase#3 %
1010 $     \BIC@AfterFiFi{%
1011 $         \expandafter\BIC@Space
1012 $         \number\csname BIC@MulDigit#2\endcsname#1 %
1013 $     }%
1014 $     \else
1015 $     \BIC@AfterFiFi{%
1016 $         \expandafter\BIC@Space
1017 $         \romannumeral0%
1018 $         \expandafter\BIC@AddXY
1019 $         \number\csname BIC@MulDigit#2\endcsname#1!%
1020 $         #3!!!%
1021 $     }%
1022 $     \fi
1023 $     \BIC@Fi
1024 $ }%

\BIC@MulDigit[3-9]
1025 $ \def\BIC@Temp#1#2{%
1026 $     \expandafter\def\csname BIC@MulDigit#1\endcsname##1{%
1027 $         \ifcase##1 0%
1028 $         \or ##1%
1029 $         \or ##2%
1030 $     }%
1031 $     \else\BigIntCalcError:ThisCannotHappen%
1032 $     \fi

```

```

1032 $ }%
1033 $ }%
1034 $ \BIC@Temp 3{6\or9\or12\or15\or18\or21\or24\or27}%
1035 $ \BIC@Temp 4{8\or12\or16\or20\or24\or28\or32\or36}%
1036 $ \BIC@Temp 5{10\or15\or20\or25\or30\or35\or40\or45}%
1037 $ \BIC@Temp 6{12\or18\or24\or30\or36\or42\or48\or54}%
1038 $ \BIC@Temp 7{14\or21\or28\or35\or42\or49\or56\or63}%
1039 $ \BIC@Temp 8{16\or24\or32\or40\or48\or56\or64\or72}%
1040 $ \BIC@Temp 9{18\or27\or36\or45\or54\or63\or72\or81}%

```

2.15 Mul

\bigintcalcMul

```

1041 \def\bigintcalcMul#1#2{%
1042   \romannumeral0%
1043   \expandafter\expandafter\expandafter\BIC@Mul
1044   \bigintcalcNum{#1}!{#2}%
1045 }

```

\BIC@Mul

```

1046 \def\BIC@Mul#1!#2{%
1047   \expandafter\expandafter\expandafter\BIC@MulSwitch
1048   \bigintcalcNum{#2}!#1!%
1049 }

```

\BIC@MulSwitch Decision table for \BIC@MulSwitch.

$x = 0$	0			
$x > 0$	$y = 0$	0		
	$y > 0$	$x > y$	+	$\text{Mul}(x, y)$
		else		$\text{Mul}(y, x)$
	$y < 0$	$x > -y$	-	$\text{Mul}(x, -y)$
		else		$\text{Mul}(-y, x)$
$x < 0$	$y = 0$	0		
	$y > 0$	$-x > y$	-	$\text{Mul}(-x, y)$
		else		$\text{Mul}(y, -x)$
	$y < 0$	$-x > -y$	+	$\text{Mul}(-x, -y)$
		else		$\text{Mul}(-y, -x)$

```

1050 \def\BIC@MulSwitch#1#2!#3#4!{%
1051   \ifcase\BIC@Sgn#1#2! % x = 0
1052     \BIC@AfterFi{ 0}%
1053   \or % x > 0
1054     \ifcase\BIC@Sgn#3#4! % y = 0
1055       \BIC@AfterFiFi{ 0}%
1056     \or % y > 0
1057       \ifnum\BIC@PosCmp#1#2!#3#4!=1 % x > y
1058         \BIC@AfterFiFiFi{%
1059           \BIC@ProcessMul0!#1#2!#3#4!%
1060         }%
1061       \else % x <= y
1062         \BIC@AfterFiFiFi{%
1063           \BIC@ProcessMul0!#3#4!#1#2!%
1064         }%
1065       \fi
1066     \else % y < 0
1067       \expandafter-\romannumeral0%
1068       \ifnum\BIC@PosCmp#1#2!#4!=1 % x > -y
1069         \BIC@AfterFiFiFiFi{%
1070           \BIC@ProcessMul0!#1#2!#4!%
1071         }%

```

```

1072     \else % x <= -y
1073         \BIC@AfterFiFiFi{%
1074             \BIC@ProcessMul0!#4!#1#2!%
1075         }%
1076     \fi
1077 \fi
1078 \else % x < 0
1079     \ifcase\BIC@Sgn#3#4! % y = 0
1080         \BIC@AfterFiFiFi{ 0}%
1081     \or % y > 0
1082         \expandafter-\romannumeral0%
1083         \ifnum\BIC@PosCmp#2!#3#4!=1 % -x > y
1084             \BIC@AfterFiFiFi{%
1085                 \BIC@ProcessMul0!#2!#3#4!%
1086             }%
1087         \else % -x <= y
1088             \BIC@AfterFiFiFi{%
1089                 \BIC@ProcessMul0!#3#4!#2!%
1090             }%
1091         \fi
1092     \else % y < 0
1093         \ifnum\BIC@PosCmp#2!#4!=1 % -x > -y
1094             \BIC@AfterFiFiFi{%
1095                 \BIC@ProcessMul0!#2!#4!%
1096             }%
1097         \else % -x <= -y
1098             \BIC@AfterFiFiFi{%
1099                 \BIC@ProcessMul0!#4!#2!%
1100             }%
1101         \fi
1102     \fi
1103 \BIC@Fi
1104 }

```

\BigIntCalcMul

```

1105 \def\BigIntCalcMul#1!#2!{%
1106     \romannumeral0%
1107     \BIC@ProcessMul0!#1!#2!%
1108 }

```

\BIC@ProcessMul

```

#1: result
#2: number  $x$ 
#3#4: number  $y$ 
1109 \def\BIC@ProcessMul#1!#2!#3#4!{%
1110     \ifx\#4\%
1111         \BIC@AfterFiFi{%
1112             \expandafter\expandafter\expandafter\BIC@Space
1113             \bigintcalcAdd{\BIC@Tim#2!#3}{#10}%
1114         }%
1115     \else
1116         \BIC@AfterFiFi{%
1117             \expandafter\expandafter\expandafter\BIC@ProcessMul
1118             \bigintcalcAdd{\BIC@Tim#2!#3}{#10}!#2!#4!%
1119         }%
1120     \BIC@Fi
1121 }

```

2.16 Sqr

\bigintcalcSqr

```

1122 \def\bigintcalcSqr#1{%
1123     \romannumeral0%

```

```

1124 \expandafter\expandafter\expandafter\BIC@Sqr
1125 \bigintcalcNum{#1}!%
1126 }

```

\BIC@Sqr

```

1127 \def\BIC@Sqr#1{%
1128   \ifx#1-%
1129     \expandafter\BIC@@Sqr
1130   \else
1131     \expandafter\BIC@@Sqr\expandafter#1%
1132   \fi
1133 }

```

\BIC@@Sqr

```

1134 \def\BIC@@Sqr#1!{%
1135   \BIC@ProcessMul0!#1!#1!%
1136 }

```

2.17 Fac

\bigintcalcFac

```

1137 \def\bigintcalcFac#1{%
1138   \romannumeral0%
1139   \expandafter\expandafter\expandafter\BIC@Fac
1140   \bigintcalcNum{#1}!%
1141 }

```

\BIC@Fac

```

1142 \def\BIC@Fac#1#2!{%
1143   \ifx#1-%
1144     \BIC@AfterFi{ 0\BigIntCalcError:FacNegative}%
1145   \else
1146     \ifnum\BIC@PosCmp#1#2!13!<0 %
1147       \ifcase#1#2 %
1148         \BIC@AfterFiFiFi{ 1}% 0!
1149         \or\BIC@AfterFiFiFi{ 1}% 1!
1150         \or\BIC@AfterFiFiFi{ 2}% 2!
1151         \or\BIC@AfterFiFiFi{ 6}% 3!
1152         \or\BIC@AfterFiFiFi{ 24}% 4!
1153         \or\BIC@AfterFiFiFi{ 120}% 5!
1154         \or\BIC@AfterFiFiFi{ 720}% 6!
1155         \or\BIC@AfterFiFiFi{ 5040}% 7!
1156         \or\BIC@AfterFiFiFi{ 40320}% 8!
1157         \or\BIC@AfterFiFiFi{ 362880}% 9!
1158         \or\BIC@AfterFiFiFi{ 3628800}% 10!
1159         \or\BIC@AfterFiFiFi{ 39916800}% 11!
1160         \or\BIC@AfterFiFiFi{ 479001600}% 12!
1161       ? \else\BigIntCalcError:ThisCannotHappen%
1162       \fi
1163     \else
1164       \BIC@AfterFiFiFi{%
1165         \BIC@ProcessFac#1#2!479001600!%
1166       }%
1167     \fi
1168   \BIC@Fi
1169 }

```

\BIC@ProcessFac #1: *n*

#2: result

```

1170 \def\BIC@ProcessFac#1!#2!{%
1171   \ifnum\BIC@PosCmp#1!12!=0 %

```

```

1172     \BIC@AfterFi{ #2}%
1173 \else
1174     \BIC@AfterFi{%
1175         \expandafter\BIC@@ProcessFac
1176         \romannumeral0\BIC@ProcessMul0!#2!#1!%
1177         !#1!%
1178     }%
1179 \BIC@Fi
1180 }

```

```

\BIC@@ProcessFac #1: result
#2:  $n$ 
1181 \def\BIC@@ProcessFac#1!#2!{%
1182     \expandafter\BIC@ProcessFac
1183     \romannumeral0\BIC@Dec#2!{%}%
1184     !#1!%
1185 }

```

2.18 Pow

```

\bigintcalcPow #1: basis
#2: power
1186 \def\bigintcalcPow#1{%
1187     \romannumeral0%
1188     \expandafter\expandafter\expandafter\BIC@Pow
1189     \bigintcalcNum{#1}!%
1190 }

\BIC@Pow #1: basis
#2: power
1191 \def\BIC@Pow#1!#2{%
1192     \expandafter\expandafter\expandafter\BIC@PowSwitch
1193     \bigintcalcNum{#2}!#1!%
1194 }

\BIC@PowSwitch #1#2: power  $y$ 
#3#4: basis  $x$ 
Decision table for \BIC@PowSwitch.

```

$y = 0$	1		
$y = 1$	x		
$y = 2$	$x < 0$	$\text{Mul}(-x, -x)$	
	else	$\text{Mul}(x, x)$	
$y < 0$	$x = 0$	DivisionByZero	
	$x = 1$	1	
	$x = -1$	ifodd(y)	-1
		else	1
	else ($ x > 1$)	0	
$y > 2$	$x = 0$	0	
	$x = 1$	1	
	$x = -1$	ifodd(y)	-1
		else	1
	$x < -1$ ($x < 0$)	ifodd(y)	$-\text{Pow}(-x, y)$
		else	$\text{Pow}(-x, y)$
	else ($x > 1$)	$\text{Pow}(x, y)$	

```

1195 \def\BIC@PowSwitch#1#2!#3#4!{%
1196     \ifcase\ifx\\#2\\%
1197         \ifx#100 %  $y = 0$ 
1198         \else\ifx#111 %  $y = 1$ 

```



```

1199         \else\ifx#122 % y = 2
1200         \else4 % y > 2
1201         \fi\fi\fi
1202     \else
1203         \ifx#1-3 % y < 0
1204         \else4 % y > 2
1205         \fi
1206     \fi
1207     \BIC@AfterFi{ 1}% y = 0
1208 \or % y = 1
1209     \BIC@AfterFi{ #3#4}%
1210 \or % y = 2
1211     \ifx#3-% x < 0
1212         \BIC@AfterFiFi{%
1213             \BIC@ProcessMul0!#4!#4!%
1214         }%
1215     \else % x >= 0
1216         \BIC@AfterFiFi{%
1217             \BIC@ProcessMul0!#3#4!#3#4!%
1218         }%
1219     \fi
1220 \or % y < 0
1221     \ifcase\ifx\#4\%
1222         \ifx#300 % x = 0
1223         \else\ifx#311 % x = 1
1224         \else3 % x > 1
1225         \fi\fi
1226     \else
1227         \ifcase\BIC@MinusOne#3#4! %
1228             3 % |x| > 1
1229         \or
1230             2 % x = -1
1231 ?         \else\BigIntCalcError:ThisCannotHappen%
1232         \fi
1233     \fi
1234     \BIC@AfterFiFi{ 0}\BigIntCalcError:DivisionByZero}% x = 0
1235 \or % x = 1
1236     \BIC@AfterFiFi{ 1}% x = 1
1237 \or % x = -1
1238     \ifcase\BIC@ModTwo#2! % even(y)
1239         \BIC@AfterFiFiFi{ 1}%
1240     \or % odd(y)
1241         \BIC@AfterFiFiFi{ -1}%
1242 ?     \else\BigIntCalcError:ThisCannotHappen%
1243     \fi
1244 \or % |x| > 1
1245     \BIC@AfterFiFi{ 0}%
1246 ?     \else\BigIntCalcError:ThisCannotHappen%
1247     \fi
1248 \or % y > 2
1249     \ifcase\ifx\#4\%
1250         \ifx#300 % x = 0
1251         \else\ifx#311 % x = 1
1252         \else4 % x > 1
1253         \fi\fi
1254     \else
1255         \ifx#3-%
1256             \ifcase\BIC@MinusOne#3#4! %
1257                 3 % x < -1
1258             \else
1259                 2 % x = -1
1260             \fi

```

```

1261         \else
1262         4 % x > 1
1263         \fi
1264     \fi
1265     \BIC@AfterFiFi{ 0}% x = 0
1266 \or % x = 1
1267     \BIC@AfterFiFi{ 1}% x = 1
1268 \or % x = -1
1269     \ifcase\BIC@ModTwo#1#2! % even(y)
1270         \BIC@AfterFiFiFi{ 1}%
1271     \or % odd(y)
1272         \BIC@AfterFiFiFi{ -1}%
1273 ?     \else\BigIntCalcError:ThisCannotHappen%
1274     \fi
1275 \or % x < -1
1276     \ifcase\BIC@ModTwo#1#2! % even(y)
1277         \BIC@AfterFiFiFi{%
1278             \BIC@PowRec#4!#1#2!1!%
1279         }%
1280 \or % odd(y)
1281     \expandafter-\romannumeral0%
1282     \BIC@AfterFiFiFi{%
1283         \BIC@PowRec#4!#1#2!1!%
1284     }%
1285 ?     \else\BigIntCalcError:ThisCannotHappen%
1286     \fi
1287 \or % x > 1
1288     \BIC@AfterFiFi{%
1289         \BIC@PowRec#3#4!#1#2!1!%
1290     }%
1291 ?     \else\BigIntCalcError:ThisCannotHappen%
1292     \fi
1293 ? \else\BigIntCalcError:ThisCannotHappen%
1294 \BIC@Fi
1295 }

```

2.18.1 Help macros

\BIC@ModTwo Macro \BIC@ModTwo expects a number without sign and returns digit 1 or 0 if the number is odd or even.

```

1296 \def\BIC@ModTwo#1#2!{%
1297   \ifx\#2\%
1298     \ifodd#1 %
1299       \BIC@AfterFiFi1%
1300     \else
1301       \BIC@AfterFiFi0%
1302     \fi
1303   \else
1304     \BIC@AfterFi{%
1305       \BIC@ModTwo#2!%
1306     }%
1307   \BIC@Fi
1308 }

```

\BIC@MinusOne Macro \BIC@MinusOne expects a number and returns digit 1 if the number equals minus one and returns 0 otherwise.

```

1309 \def\BIC@MinusOne#1#2!{%
1310   \ifx#1-%
1311     \BIC@@MinusOne#2!%
1312   \else
1313     0%
1314   \fi

```

```
1315 }
```

```
\BIC@@MinusOne
```

```
1316 \def\BIC@@MinusOne#1#2!{%
1317   \ifx#11%
1318     \ifx\|#2\|%
1319       1%
1320     \else
1321       0%
1322     \fi
1323   \else
1324     0%
1325   \fi
1326 }
```

2.18.2 Recursive calculation

```
\BIC@PowRec      Pow(x, y) {
                  PowRec(x, y, 1)
                  }
                  PowRec(x, y, r) {
                    if y == 1 then
                      return r
                    else
                      ifodd y then
                        return PowRec(x*x, y div 2, r*x) % y div 2 = (y-1)/2
                      else
                        return PowRec(x*x, y div 2, r)
                      fi
                    fi
                  }
                  #1: x (basis)
                  #2#3: y (power)
                  #4: r (result)
1327 \def\BIC@PowRec#1!#2#3!#4!{%
1328   \ifcase\ifx#21\ifx\|#3\|0 \else1 \fi\else1 \fi % y = 1
1329     \ifnum\BIC@PosCmp#1!#4!=1 % x > r
1330       \BIC@AfterFiFi{%
1331         \BIC@ProcessMul0!#1!#4!%
1332       }%
1333     \else
1334       \BIC@AfterFiFi{%
1335         \BIC@ProcessMul0!#4!#1!%
1336       }%
1337     \fi
1338   \or
1339     \ifcase\BIC@ModTwo#2#3! % even(y)
1340       \BIC@AfterFiFi{%
1341         \expandafter\BIC@@PowRec\romannumeral0%
1342         \BIC@Shr#2#3!%
1343         !#1!#4!%
1344       }%
1345     \or % odd(y)
1346       \ifnum\BIC@PosCmp#1!#4!=1 % x > r
1347         \BIC@AfterFiFiFi{%
1348           \expandafter\BIC@@@PowRec\romannumeral0%
1349           \BIC@ProcessMul0!#1!#4!%
1350           !#1!#2#3!%
1351         }%
1352       \else
1353         \BIC@AfterFiFiFi{%
1354           \expandafter\BIC@@@PowRec\romannumeral0%
```

```

1355         \BIC@ProcessMul0!#1!#4!%
1356         !#1!#2#3!%
1357     }%
1358     \fi
1359 ?   \else\BigIntCalcError:ThisCannotHappen%
1360     \fi
1361 ?   \else\BigIntCalcError:ThisCannotHappen%
1362     \BIC@Fi
1363 }

```

```

\BIC@@PowRec #1:  $y/2$ 
#2:  $x$ 
#3: new  $r$  ( $r$  or  $r * x$ )
1364 \def\BIC@@PowRec#1!#2!#3!{%
1365     \expandafter\BIC@PowRec\romannumeral0%
1366     \BIC@ProcessMul0!#2!#2!%
1367     !#1!#3!%
1368 }

```

```

\BIC@@@PowRec #1:  $r * x$  #2:  $x$  #3:  $y$ 
1369 \def\BIC@@@PowRec#1!#2!#3!{%
1370     \expandafter\BIC@@PowRec\romannumeral0%
1371     \BIC@@Shr#3!%
1372     !#2!#1!%
1373 }

```

2.19 Div

```

\bigintcalcDiv #1:  $x$ 
#2:  $y$  (divisor)
1374 \def\bigintcalcDiv#1{%
1375     \romannumeral0%
1376     \expandafter\expandafter\expandafter\BIC@Div
1377     \bigintcalcNum{#1}!%
1378 }

```

```

\BIC@Div #1:  $x$ 
#2:  $y$ 
1379 \def\BIC@Div#1!#2{%
1380     \expandafter\expandafter\expandafter\BIC@DivSwitchSign
1381     \bigintcalcNum{#2}!#1!%
1382 }

```

```

\BigIntCalcDiv
1383 \def\BigIntCalcDiv#1!#2!{%
1384     \romannumeral0%
1385     \BIC@DivSwitchSign#2!#1!%
1386 }

```

\BIC@DivSwitchSign Decision table for \BIC@DivSwitchSign.

$y = 0$	DivisionByZero	
$y > 0$	$x = 0$	0
	$x > 0$	DivSwitch(+, x, y)
	$x < 0$	DivSwitch(−, $-x, y$)
$y < 0$	$x = 0$	0
	$x > 0$	DivSwitch(−, $x, -y$)
	$x < 0$	DivSwitch(+, $-x, -y$)

```

#1:  $y$  (divisor)
#2:  $x$ 
1387 \def\BIC@DivSwitchSign#1#2!#3#4!{%
1388   \ifcase\BIC@Sgn#1#2! %  $y = 0$ 
1389     \BIC@AfterFi{ 0\BigIntCalcError:DivisionByZero}%
1390   \or %  $y > 0$ 
1391     \ifcase\BIC@Sgn#3#4! %  $x = 0$ 
1392       \BIC@AfterFiFi{ 0}%
1393     \or %  $x > 0$ 
1394       \BIC@AfterFiFi{%
1395         \BIC@DivSwitch{ }#3#4!#1#2!%
1396       }%
1397     \else %  $x < 0$ 
1398       \BIC@AfterFiFi{%
1399         \BIC@DivSwitch{-#4!#1#2!%
1400       }%
1401     \fi
1402   \else %  $y < 0$ 
1403     \ifcase\BIC@Sgn#3#4! %  $x = 0$ 
1404       \BIC@AfterFiFi{ 0}%
1405     \or %  $x > 0$ 
1406       \BIC@AfterFiFi{%
1407         \BIC@DivSwitch{-#3#4!#2!%
1408       }%
1409     \else %  $x < 0$ 
1410       \BIC@AfterFiFi{%
1411         \BIC@DivSwitch{ }#4!#2!%
1412       }%
1413     \fi
1414   \BIC@Fi
1415 }

```

\BIC@DivSwitch Decision table for \BIC@DivSwitch.

$y = x$	sign 1	
$y > x$	0	
$y < x$	$y = 1$	sign x
	$y = 2$	sign Shr(x)
	$y = 4$	sign Shr(Shr(x))
	else	sign ProcessDiv(x, y)

```

#1: sign
#2:  $x$ 
#3#4:  $y$  ( $y \neq 0$ )
1416 \def\BIC@DivSwitch#1#2!#3#4!{%
1417   \ifcase\BIC@PosCmp#3#4!#2!%  $y = x$ 
1418     \BIC@AfterFi{ #1}%
1419   \or %  $y > x$ 
1420     \BIC@AfterFi{ 0}%
1421   \else %  $y < x$ 
1422     \ifx\#1\%
1423       \else
1424         \expandafter-\romannumeral0%
1425       \fi
1426     \ifcase\ifx\#4\%
1427       \ifx#310 %  $y = 1$ 
1428       \else\ifx#321 %  $y = 2$ 
1429       \else\ifx#342 %  $y = 4$ 
1430       \else3 %  $y > 2$ 
1431       \fi\fi\fi
1432     \else
1433       3 %  $y > 2$ 

```

```

1434         \fi
1435         \BIC@AfterFiFi{ #2}% y = 1
1436     \or % y = 2
1437         \BIC@AfterFiFi{%
1438             \BIC@Shr#2!%
1439         }%
1440     \or % y = 4
1441         \BIC@AfterFiFi{%
1442             \expandafter\BIC@Shr\romannumeral0%
1443             \BIC@Shr#2!%
1444         }%
1445     \or % y > 2
1446         \BIC@AfterFiFi{%
1447             \BIC@DivStartX#2!#3#4!!!%
1448         }%
1449 ? \else\BigIntCalcError:ThisCannotHappen%
1450     \fi
1451 \BIC@Fi
1452 }

\BIC@ProcessDiv #1#2:  $x$ 
#3#4:  $y$ 
#5: collect first digits of  $x$ 
#6: corresponding digits of  $y$ 
1453 \def\BIC@DivStartX#1#2!#3#4!#5!#6!{%
1454     \ifx\#4\%
1455         \BIC@AfterFi{%
1456             \BIC@DivStartYii#6#3#4!{#5#1}#2=!%
1457         }%
1458     \else
1459         \BIC@AfterFi{%
1460             \BIC@DivStartX#2!#4!#5#1!#6#3!%
1461         }%
1462     \BIC@Fi
1463 }

\BIC@DivStartYii #1:  $y$ 
#2:  $x, =$ 
1464 \def\BIC@DivStartYii#1!{%
1465     \expandafter\BIC@DivStartYiv\romannumeral0%
1466     \BIC@Shl#1!%
1467     !#1!%
1468 }

\BIC@DivStartYiv #1:  $2y$ 
#2:  $y$ 
#3:  $x, =$ 
1469 \def\BIC@DivStartYiv#1!{%
1470     \expandafter\BIC@DivStartYvi\romannumeral0%
1471     \BIC@Shl#1!%
1472     !#1!%
1473 }

\BIC@DivStartYvi #1:  $4y$ 
#2:  $2y$ 
#3:  $y$ 
#4:  $x, =$ 
1474 \def\BIC@DivStartYvi#1!#2!{%
1475     \expandafter\BIC@DivStartYviii\romannumeral0%
1476     \BIC@AddXY#1!#2!!!%
1477     !#1!#2!%
1478 }

```

```

\BIC@DivStartYviii #1: 6y
                   #2: 4y
                   #3: 2y
                   #4: y
                   #5: x, =
1479 \def\BIC@DivStartYviii#1!#2!{%
1480   \expandafter\BIC@DivStart\romannumeral0%
1481   \BIC@Shl#2!%
1482   !#1!#2!%
1483 }

\BIC@DivStart #1: 8y
              #2: 6y
              #3: 4y
              #4: 2y
              #5: y
              #6: x, =
1484 \def\BIC@DivStart#1!#2!#3!#4!#5!#6!{%
1485   \BIC@ProcessDiv#6!!#5!#4!#3!#2!#1!=%
1486 }

\BIC@ProcessDiv #1#2#3: x, =
                #4: result
                #5: y
                #6: 2y
                #7: 4y
                #8: 6y
                #9: 8y
1487 \def\BIC@ProcessDiv#1#2#3!#4!#5!{%
1488   \ifcase\BIC@PosCmp#5!#1!% y = #1
1489     \ifx#2=%
1490       \BIC@AfterFiFi{\BIC@DivCleanup{#41}}%
1491     \else
1492       \BIC@AfterFiFi{%
1493         \BIC@ProcessDiv#2#3!#41!#5!%
1494       }%
1495     \fi
1496   \or % y > #1
1497     \ifx#2=%
1498       \BIC@AfterFiFi{\BIC@DivCleanup{#40}}%
1499     \else
1500       \ifx\#4\%
1501         \BIC@AfterFiFiFi{%
1502           \BIC@ProcessDiv{#1#2}#3!#5!%
1503         }%
1504       \else
1505         \BIC@AfterFiFiFi{%
1506           \BIC@ProcessDiv{#1#2}#3!#40!#5!%
1507         }%
1508       \fi
1509     \fi
1510   \else % y < #1
1511     \BIC@AfterFi{%
1512       \BIC@@ProcessDiv{#1}#2#3!#4!#5!%
1513     }%
1514   \BIC@Fi
1515 }

\BIC@DivCleanup #1: result
                #2: garbage
1516 \def\BIC@DivCleanup#1#2={ #1}%

```

```

\BIC@@ProcessDiv
1517 \def\BIC@@ProcessDiv#1#2#3!#4!#5!#6!#7!{%
1518 \ifcase\BIC@PosCmp#7!#1!% 4y = #1
1519 \ifx#2=%
1520 \BIC@AfterFiFi{\BIC@DivCleanup{#44}}}%
1521 \else
1522 \BIC@AfterFiFi{%
1523 \BIC@ProcessDiv#2#3!#44!#5!#6!#7!%
1524 }%
1525 \fi
1526 \or % 4y > #1
1527 \ifcase\BIC@PosCmp#6!#1!% 2y = #1
1528 \ifx#2=%
1529 \BIC@AfterFiFiFi{\BIC@DivCleanup{#42}}}%
1530 \else
1531 \BIC@AfterFiFiFi{%
1532 \BIC@ProcessDiv#2#3!#42!#5!#6!#7!%
1533 }%
1534 \fi
1535 \or % 2y > #1
1536 \ifx#2=%
1537 \BIC@AfterFiFiFi{\BIC@DivCleanup{#41}}}%
1538 \else
1539 \BIC@AfterFiFiFi{%
1540 \BIC@DivSub#1!#5!#2#3!#41!#5!#6!#7!%
1541 }%
1542 \fi
1543 \else % 2y < #1
1544 \BIC@AfterFiFi{%
1545 \expandafter\BIC@ProcessDivII\romannumeral0%
1546 \BIC@SubXY#1!#6!!!%
1547 !#2#3!#4!#5!23%
1548 #6!#7!%
1549 }%
1550 \fi
1551 \else % 4y < #1
1552 \BIC@AfterFiFi{%
1553 \BIC@@@ProcessDiv{#1}#2#3!#4!#5!#6!#7!%
1554 }%
1555 \BIC@Fi
1556 }

```

\BIC@DivSub Next token group: #1-#2 and next digit #3.

```

1557 \def\BIC@DivSub#1!#2!#3{%
1558 \expandafter\BIC@ProcessDiv\expandafter{%
1559 \romannumeral0%
1560 \BIC@SubXY#1!#2!!!%
1561 #3%
1562 }%
1563 }

```

\BIC@ProcessDivII #1: $x' - 2y$
#2#3: remaining x , =
#4: result
#5: y
#6: first possible result digit
#7: second possible result digit

```

1564 \def\BIC@ProcessDivII#1!#2#3!#4!#5!#6#7{%
1565 \ifcase\BIC@PosCmp#5!#1!% y = #1
1566 \ifx#2=%
1567 \BIC@AfterFiFi{\BIC@DivCleanup{#4#7}}}%
1568 \else

```



```

1569     \BIC@AfterFiFi{%
1570     \BIC@ProcessDiv#2#3!#4#7!#5!%
1571     }%
1572     \fi
1573 \or % y > #1
1574     \ifx#2=%
1575     \BIC@AfterFiFi{\BIC@DivCleanup{#4#6}}%
1576     \else
1577     \BIC@AfterFiFi{%
1578     \BIC@ProcessDiv{#1#2}#3!#4#6!#5!%
1579     }%
1580     \fi
1581 \else % y < #1
1582     \ifx#2=%
1583     \BIC@AfterFiFi{\BIC@DivCleanup{#4#7}}%
1584     \else
1585     \BIC@AfterFiFi{%
1586     \BIC@DivSub#1!#5!#2#3!#4#7!#5!%
1587     }%
1588     \fi
1589 \BIC@Fi
1590 }

```

```

\BIC@ProcessDivIV #1#2#3:  $x, =, x > 4y$ 
#4: result
#5:  $y$ 
#6:  $2y$ 
#7:  $4y$ 
#8:  $6y$ 
#9:  $8y$ 

1591 \def\BIC@@@ProcessDiv#1#2#3!#4!#5!#6!#7!#8!#9!{%
1592 \ifcase\BIC@PosCmp#8!#1!% 6y = #1
1593     \ifx#2=%
1594     \BIC@AfterFiFi{\BIC@DivCleanup{#46}}%
1595     \else
1596     \BIC@AfterFiFi{%
1597     \BIC@ProcessDiv#2#3!#46!#5!#6!#7!#8!#9!%
1598     }%
1599     \fi
1600 \or % 6y > #1
1601     \BIC@AfterFi{%
1602     \expandafter\BIC@ProcessDivII\romannumeral0%
1603     \BIC@SubXY#1!#7!!!%
1604     !#2#3!#4!#5!45%
1605     #6!#7!#8!#9!%
1606     }%
1607 \else % 6y < #1
1608     \ifcase\BIC@PosCmp#9!#1!% 8y = #1
1609     \ifx#2=%
1610     \BIC@AfterFiFiFi{\BIC@DivCleanup{#48}}%
1611     \else
1612     \BIC@AfterFiFiFi{%
1613     \BIC@ProcessDiv#2#3!#48!#5!#6!#7!#8!#9!%
1614     }%
1615     \fi
1616 \or % 8y > #1
1617     \BIC@AfterFiFi{%
1618     \expandafter\BIC@ProcessDivII\romannumeral0%
1619     \BIC@SubXY#1!#8!!!%
1620     !#2#3!#4!#5!67%
1621     #6!#7!#8!#9!%
1622     }%

```

```

1623     \else % 8y < #1
1624         \BIC@AfterFiFi{%
1625             \expandafter\BIC@ProcessDivII\romannumeral0%
1626             \BIC@SubXY#1!#9!!!%
1627             !#2#3!#4!#5!89%
1628             #6!#7!#8!#9!%
1629         }%
1630     \fi
1631 \BIC@Fi
1632 }

```

2.20 Mod

```

\bigintcalcMod #1: x
#2: y
1633 \def\bigintcalcMod#1{%
1634     \romannumeral0%
1635     \expandafter\expandafter\expandafter\BIC@Mod
1636     \bigintcalcNum{#1}!%
1637 }

\BIC@Mod #1: x
#2: y
1638 \def\BIC@Mod#1!#2{%
1639     \expandafter\expandafter\expandafter\BIC@ModSwitchSign
1640     \bigintcalcNum{#2}!#1!%
1641 }

\BigIntCalcMod
1642 \def\BigIntCalcMod#1!#2!{%
1643     \romannumeral0%
1644     \BIC@ModSwitchSign#2!#1!%
1645 }

```

\BIC@ModSwitchSign Decision table for \BIC@ModSwitchSign.

$y = 0$	DivisionByZero	
$y > 0$	$x = 0$	0
	else	ModSwitch(+, x, y)
$y < 0$	ModSwitch(-, $-x, -y$)	

```

#1#2: y
#3#4: x
1646 \def\BIC@ModSwitchSign#1#2!#3#4!{%
1647     \ifcase\ifx\#2\%
1648         \ifx#100 % y = 0
1649         \else1 % y > 0
1650         \fi
1651     \else
1652         \ifx#1-2 % y < 0
1653         \else1 % y > 0
1654         \fi
1655     \fi
1656     \BIC@AfterFi{ 0\BigIntCalcError:DivisionByZero}%
1657 \or % y > 0
1658     \ifcase\ifx\#4\%
1659         \ifx#300 \else1 \fi\else1 \fi % x = 0
1660     \else
1661         \BIC@AfterFiFi{%
1662             \BIC@ModSwitch{#3#4!#1#2!%
1663         }%

```

```

1664 \fi
1665 \else % y < 0
1666 \ifcase\ifx\#4\%
1667 \ifx#300 % x = 0
1668 \else1 % x > 0
1669 \fi
1670 \else
1671 \ifx#3-2 % x < 0
1672 \else1 % x > 0
1673 \fi
1674 \fi
1675 \BIC@AfterFiFi{ 0}%
1676 \or % x > 0
1677 \BIC@AfterFiFi{%
1678 \BIC@ModSwitch--#3#4!#2!%
1679 }%
1680 \else % x < 0
1681 \BIC@AfterFiFi{%
1682 \BIC@ModSwitch-#4!#2!%
1683 }%
1684 \fi
1685 \BIC@Fi
1686 }

```

\BIC@ModSwitch Decision table for \BIC@ModSwitch.

$y = 1$	0	
$y = 2$	ifodd(x)	sign 1
	else	0
$y > 2$	$x < 0$	$z \leftarrow x - (x/y) * y; \quad (z < 0) ? z + y : z$
	$x > 0$	$x - (x/y) * y$

#1: sign

#2#3: x

#4#5: y

```

1687 \def\BIC@ModSwitch#1#2#3!#4#5!{%
1688 \ifcase\ifx\#5\%
1689 \ifx#410 % y = 1
1690 \else\ifx#421 % y = 2
1691 \else2 % y > 2
1692 \fi\fi
1693 \else2 % y > 2
1694 \fi
1695 \BIC@AfterFi{ 0}% y = 1
1696 \or % y = 2
1697 \ifcase\BIC@ModTwo#2#3! % even(x)
1698 \BIC@AfterFiFi{ 0}%
1699 \or % odd(x)
1700 \BIC@AfterFiFi{ #11}%
1701 ? \else\BigIntCalcError:ThisCannotHappen%
1702 \fi
1703 \or % y > 2
1704 \ifx\#1\%
1705 \else
1706 \expandafter\BIC@Space\romannumeral0%
1707 \expandafter\BIC@ModMinus\romannumeral0%
1708 \fi
1709 \ifx#2-% x < 0
1710 \BIC@AfterFiFi{%
1711 \expandafter\expandafter\expandafter\BIC@ModX
1712 \bigintcalcSub{#2#3}{%
1713 \bigintcalcMul{#4#5}{\bigintcalcDiv{#2#3}{#4#5}}}%

```

```

1714      }!#4#5!%
1715      }%
1716      \else % x > 0
1717      \BIC@AfterFiFi{%
1718      \expandafter\expandafter\expandafter\BIC@Space
1719      \bigintcalcSub{#2#3}{%
1720      \bigintcalcMul{#4#5}{\bigintcalcDiv{#2#3}{#4#5}}%
1721      }%
1722      }%
1723      \fi
1724 ? \else\BigIntCalcError:ThisCannotHappen%
1725 \BIC@Fi
1726 }

\BIC@ModMinus

1727 \def\BIC@ModMinus#1{%
1728 \ifx#10%
1729 \BIC@AfterFi{ 0}%
1730 \else
1731 \BIC@AfterFi{ -#1}%
1732 \BIC@Fi
1733 }

\BIC@ModX #1#2: z
#3: x
1734 \def\BIC@ModX#1#2!#3!{%
1735 \ifx#1-% z < 0
1736 \BIC@AfterFi{%
1737 \expandafter\BIC@Space\romannumeral0%
1738 \BIC@SubXY#3!#2!!!%
1739 }%
1740 \else % z >= 0
1741 \BIC@AfterFi{ #1#2}%
1742 \BIC@Fi
1743 }

1744 \BIC@AtEnd
1745 </package>

```

3 Test

3.1 Catcode checks for loading

```

1746 <*test1>
1747 \catcode'\{=1 %
1748 \catcode'\}=2 %
1749 \catcode'\#=6 %
1750 \catcode'\@=11 %
1751 \expandafter\ifx\csname count@\endcsname\relax
1752 \countdef\count@=255 %
1753 \fi
1754 \expandafter\ifx\csname @gobble\endcsname\relax
1755 \long\def\@gobble#1{}%
1756 \fi
1757 \expandafter\ifx\csname @firstofone\endcsname\relax
1758 \long\def\@firstofone#1{#1}%
1759 \fi
1760 \expandafter\ifx\csname loop\endcsname\relax
1761 \expandafter\@firstofone
1762 \else
1763 \expandafter\@gobble

```

```

1764 \fi
1765 {%
1766   \def\loop#1\repeat{%
1767     \def\body{#1}%
1768     \iterate
1769   }%
1770   \def\iterate{%
1771     \body
1772     \let\next\iterate
1773     \else
1774       \let\next\relax
1775     \fi
1776     \next
1777   }%
1778   \let\repeat=\fi
1779 }%
1780 \def\RestoreCatcodes{}
1781 \count@=0 %
1782 \loop
1783   \edef\RestoreCatcodes{%
1784     \RestoreCatcodes
1785     \catcode\the\count@=\the\catcode\count@\relax
1786   }%
1787 \ifnum\count@<255 %
1788   \advance\count@ 1 %
1789 \repeat
1790
1791 \def\RangeCatcodeInvalid#1#2{%
1792   \count@=#1\relax
1793   \loop
1794     \catcode\count@=15 %
1795     \ifnum\count@<#2\relax
1796       \advance\count@ 1 %
1797     \repeat
1798 }
1799 \def\Test{%
1800   \RangeCatcodeInvalid{0}{47}%
1801   \RangeCatcodeInvalid{58}{64}%
1802   \RangeCatcodeInvalid{91}{96}%
1803   \RangeCatcodeInvalid{123}{255}%
1804   \catcode'\@=12 %
1805   \catcode'\=0 %
1806   \catcode'\{=1 %
1807   \catcode'\}=2 %
1808   \catcode'\#=6 %
1809   \catcode'\[=12 %
1810   \catcode'\]=12 %
1811   \catcode'\%=14 %
1812   \catcode'\ =10 %
1813   \catcode13=5 %
1814   \input bigintcalc.sty\relax
1815   \RestoreCatcodes
1816 }
1817 \Test
1818 \csname @@end\endcsname
1819 \end

```

1820 \langle /test1 \rangle

3.2 Macro tests

3.2.1 Preamble with test macro definitions

1821 \langle *test2 \rangle

```

1822 \NeedsTeXFormat{LaTeX2e}
1823 \nofiles
1824 \documentclass{article}
1825 <noetex> \let\SavedNumexpr\numexpr
1826 <noetex> \let\numexpr\UNDEFINED
1827 \makeatletter
1828 \chardef\BIC@TestMode=1 %
1829 \makeatother
1830 \usepackage{bigintcalc}[2007/11/11]
1831 <noetex> \let\numexpr\SavedNumexpr
1832 \usepackage{qstest}
1833 \IncludeTests{*}
1834 \LogTests{log}{*}{*}
1835 \newcommand*{\TestSpaceAtEnd}[1]{%
1836 <noetex> \let\SavedNumexpr\numexpr
1837 <noetex> \let\numexpr\UNDEFINED
1838 \edef\resultA{#1}%
1839 \edef\resultB{#1 }%
1840 <noetex> \let\numexpr\SavedNumexpr
1841 \Expect*{\resultA\space}*{\resultB}%
1842 }
1843 \newcommand*{\TestResult}[2]{%
1844 <noetex> \let\SavedNumexpr\numexpr
1845 <noetex> \let\numexpr\UNDEFINED
1846 \edef\result{#1}%
1847 <noetex> \let\numexpr\SavedNumexpr
1848 \Expect*{\result}{#2}%
1849 }
1850 \newcommand*{\TestResultTwoExpansions}[2]{%
1851 <*noetex>
1852 \begingroup
1853 \let\numexpr\UNDEFINED
1854 \expandafter\expandafter\expandafter
1855 \endgroup
1856 </noetex>
1857 \expandafter\expandafter\expandafter\Expect
1858 \expandafter\expandafter\expandafter{#1}{#2}%
1859 }
1860 \newcount\TestCount
1861 <etex> \newcommand*{\TestArg}[1]{\numexpr#1\relax}
1862 <noetex> \newcommand*{\TestArg}[1]{#1}
1863 \newcommand*{\TestTeXDivide}[2]{%
1864 \TestCount=\TestArg{#1}\relax
1865 \divide\TestCount by \TestArg{#2}\relax
1866 \Expect*{\bigintcalcDiv{#1}{#2}}*{\the\TestCount}%
1867 }
1868 \newcommand*{\Test}[2]{%
1869 \TestResult{#1}{#2}%
1870 \TestResultTwoExpansions{#1}{#2}%
1871 \TestSpaceAtEnd{#1}%
1872 }
1873 \newcommand*{\TestExch}[2]{\Test{#2}{#1}}
1874 \newcommand*{\TestInv}[2]{%
1875 \Test{\bigintcalcInv{#1}}{#2}%
1876 }
1877 \newcommand*{\TestAbs}[2]{%
1878 \Test{\bigintcalcAbs{#1}}{#2}%
1879 }
1880 \newcommand*{\TestSgn}[2]{%
1881 \Test{\bigintcalcSgn{#1}}{#2}%
1882 }
1883 \newcommand*{\TestMin}[3]{%

```

```

1884 \Test{\bigintcalcMin{#1}{#2}}{#3}%
1885 }
1886 \newcommand*{\TestMax}[3]{%
1887 \Test{\bigintcalcMax{#1}{#2}}{#3}%
1888 }
1889 \newcommand*{\TestCmp}[3]{%
1890 \Test{\bigintcalcCmp{#1}{#2}}{#3}%
1891 }
1892 \newcommand*{\TestOdd}[2]{%
1893 \Test{\bigintcalcOdd{#1}}{#2}%
1894 \edef\x{%
1895 \noexpand\Test{%
1896 \noexpand\BigIntCalcOdd
1897 \bigintcalcAbs{#1}!%
1898 }{#2}%
1899 }%
1900 \x
1901 }
1902 \newcommand*{\TestInc}[2]{%
1903 \Test{\bigintcalcInc{#1}}{#2}%
1904 \ifnum\bigintcalcSgn{#1}>-1 %
1905 \edef\x{%
1906 \noexpand\Test{%
1907 \noexpand\BigIntCalcInc\bigintcalcNum{#1}!%
1908 }{#2}%
1909 }%
1910 \x
1911 \fi
1912 }
1913 \newcommand*{\TestDec}[2]{%
1914 \Test{\bigintcalcDec{#1}}{#2}%
1915 \ifnum\bigintcalcSgn{#1}>0 %
1916 \edef\x{%
1917 \noexpand\Test{%
1918 \noexpand\BigIntCalcDec\bigintcalcNum{#1}!%
1919 }{#2}%
1920 }%
1921 \x
1922 \fi
1923 }
1924 \newcommand*{\TestAdd}[3]{%
1925 \Test{\bigintcalcAdd{#1}{#2}}{#3}%
1926 \ifnum\bigintcalcSgn{#1}>0 %
1927 \ifnum\bigintcalcSgn{#2}> 0 %
1928 \ifnum\bigintcalcCmp{#1}{#2}>0 %
1929 \edef\x{%
1930 \noexpand\Test{%
1931 \noexpand\BigIntCalcAdd
1932 \bigintcalcNum{#1}!\bigintcalcNum{#2}!%
1933 }{#3}%
1934 }%
1935 \x
1936 \else
1937 \edef\x{%
1938 \noexpand\Test{%
1939 \noexpand\BigIntCalcAdd
1940 \bigintcalcNum{#2}!\bigintcalcNum{#1}!%
1941 }{#3}%
1942 }%
1943 \x
1944 \fi
1945 \fi

```

```

1946 \fi
1947 }
1948 \newcommand*\TestSub}[3]{%
1949 \Test{\bigintcalcSub{#1}{#2}}{#3}%
1950 \ifnum\bigintcalcSgn{#1}>0 %
1951 \ifnum\bigintcalcSgn{#2}> 0 %
1952 \ifnum\bigintcalcCmp{#1}{#2}>0 %
1953 \edef\x{%
1954 \noexpand\Test{%
1955 \noexpand\BigIntCalcSub
1956 \bigintcalcNum{#1}!\bigintcalcNum{#2}!%
1957 }{#3}%
1958 }%
1959 \x
1960 \fi
1961 \fi
1962 \fi
1963 }
1964 \newcommand*\TestShl}[2]{%
1965 \Test{\bigintcalcShl{#1}{#2}}%
1966 \edef\x{%
1967 \noexpand\Test{%
1968 \noexpand\BigIntCalcShl\bigintcalcAbs{#1}!%
1969 }{\bigintcalcAbs{#2}}%
1970 }%
1971 \x
1972 }
1973 \newcommand*\TestShr}[2]{%
1974 \Test{\bigintcalcShr{#1}{#2}}%
1975 \edef\x{%
1976 \noexpand\Test{%
1977 \noexpand\BigIntCalcShr\bigintcalcAbs{#1}!%
1978 }{\bigintcalcAbs{#2}}%
1979 }%
1980 \x
1981 }
1982 \newcommand*\TestMul}[3]{%
1983 \Test{\bigintcalcMul{#1}{#2}}{#3}%
1984 \edef\x{%
1985 \noexpand\Test{%
1986 \noexpand\BigIntCalcMul
1987 \bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
1988 }{\bigintcalcAbs{#3}}%
1989 }%
1990 \x
1991 }
1992 \newcommand*\TestSqr}[2]{%
1993 \Test{\bigintcalcSqr{#1}{#2}}%
1994 }
1995 \newcommand*\TestFac}[2]{%
1996 \expandafter\TestExch\expandafter{%
1997 \the\numexpr#2%
1998 }{\bigintcalcFac{#1}}%
1999 }
2000 \newcommand*\TestFacBig}[2]{%
2001 \Test{\bigintcalcFac{#1}{#2}}%
2002 }
2003 \newcommand*\TestPow}[3]{%
2004 \Test{\bigintcalcPow{#1}{#2}}{#3}%
2005 }
2006 \newcommand*\TestDiv}[3]{%
2007 \Test{\bigintcalcDiv{#1}{#2}}{#3}%

```



```

2008 \TestTeXDivide{#1}{#2}%
2009 }
2010 \newcommand*{\TestDivBig}[3]{%
2011 \Test{\bigintcalcDiv{#1}{#2}}{#3}%
2012 \edef\x{%
2013 \noexpand\Test{%
2014 \noexpand\BigIntCalcDiv\bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
2015 }{\bigintcalcAbs{#3}}}%
2016 }%
2017 }
2018 \newcommand*{\TestMod}[3]{%
2019 \Test{\bigintcalcMod{#1}{#2}}{#3}%
2020 \ifcase\ifcase\bigintcalcSgn{#1} 0%
2021 \or
2022 \ifcase\bigintcalcSgn{#2} 1%
2023 \or 0%
2024 \else 1%
2025 \fi
2026 \else
2027 \ifcase\bigintcalcSgn{#2} 1%
2028 \or 1%
2029 \else 0%
2030 \fi
2031 \fi\relax
2032 \edef\x{%
2033 \noexpand\Test{%
2034 \noexpand\BigIntCalcMod
2035 \bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
2036 }{\bigintcalcAbs{#3}}}%
2037 }%
2038 \x
2039 \fi
2040 }

```

3.2.2 Time

```

2041 \begingroup\expandafter\expandafter\expandafter\endgroup
2042 \expandafter\ifx\csname pdfresettimer\endcsname\relax
2043 \else
2044 \makeatletter
2045 \newcount\SummaryTime
2046 \newcount\TestTime
2047 \SummaryTime=\z@
2048 \newcommand*{\PrintTime}[2]{%
2049 \typeout{%
2050 [Time #1: \strip@pt\dimexpr\number#2sp\relax\space s]%
2051 }%
2052 }%
2053 \newcommand*{\StartTime}[1]{%
2054 \renewcommand*{\TimeDescription}{#1}%
2055 \pdfresettimer
2056 }%
2057 \newcommand*{\TimeDescription}{}%
2058 \newcommand*{\StopTime}{%
2059 \TestTime=\pdfelapsedtime
2060 \global\advance\SummaryTime\TestTime
2061 \PrintTime\TimeDescription\TestTime
2062 }%
2063 \let\saved@qstest\qstest
2064 \let\saved@endqstest\endqstest
2065 \def\qstest#1#2{%
2066 \saved@qstest{#1}{#2}%
2067 \StartTime{#1}%
2068 }%

```

```

2069 \def\endqstest{%
2070   \StopTime
2071   \saved@endqstest
2072 }%
2073 \AtEndDocument{%
2074   \PrintTime{summary}\SummaryTime
2075 }%
2076 \makeatother
2077 \fi

```

3.2.3 Test sets

```

2078 \makeatletter
2079
2080 \begin{qstest}{inv}{inv}%
2081   \TestInv{0}{0}%
2082   \TestInv{1}{-1}%
2083   \TestInv{-1}{1}%
2084   \TestInv{10}{-10}%
2085   \TestInv{-10}{10}%
2086   \TestInv{2147483647}{-2147483647}%
2087   \TestInv{-2147483647}{2147483647}%
2088   \TestInv{12345678901234567890}{-12345678901234567890}%
2089   \TestInv{-12345678901234567890}{12345678901234567890}%
2090   \TestInv{ 0 }{0}%
2091   \TestInv{ 1 }{-1}%
2092   \TestInv{--1}{-1}%
2093   \TestInv{\number\z@}{0}%
2094   \TestInv{\ifx\relax\relax1\fi}{-1}%
2095   \TestInv{\ifx\relax\relax-\fi\ifx234\else1\fi}{1}%
2096 \end{qstest}
2097
2098 \begin{qstest}{abs}{abs}%
2099   \TestAbs{0}{0}%
2100   \TestAbs{1}{1}%
2101   \TestAbs{-1}{1}%
2102   \TestAbs{10}{10}%
2103   \TestAbs{-10}{10}%
2104   \TestAbs{2147483647}{2147483647}%
2105   \TestAbs{-2147483647}{-2147483647}%
2106   \TestAbs{12345678901234567890}{12345678901234567890}%
2107   \TestAbs{-12345678901234567890}{-12345678901234567890}%
2108   \TestAbs{ 0 }{0}%
2109   \TestAbs{ 1 }{1}%
2110   \TestAbs{--1}{1}%
2111   \TestAbs{--++1}{1}%
2112   \TestAbs{00000000000}{0}%
2113   \TestAbs{00000001000}{1000}%
2114   \TestAbs{\ifx\relax\relax 0\else 1\fi}{0}%
2115 \end{qstest}
2116
2117 \begin{qstest}{sign}{sign}%
2118   \TestSgn{0}{0}%
2119   \TestSgn{1}{1}%
2120   \TestSgn{-1}{-1}%
2121   \TestSgn{10}{1}%
2122   \TestSgn{-10}{-1}%
2123   \TestSgn{2147483647}{1}%
2124   \TestSgn{-2147483647}{-1}%
2125   \TestSgn{12345678901234567890}{1}%
2126   \TestSgn{-12345678901234567890}{-1}%
2127   \TestSgn{ 0 }{0}%
2128   \TestSgn{ 2 }{1}%
2129   \TestSgn{ -2 }{-1}%

```

```

2130 \TestSgn{--2}{1}%
2131 \TestSgn{\number\z@}{0}%
2132 \TestSgn{\number\@ne}{1}%
2133 \TestSgn{\number\m@ne}{-1}%
2134 \TestSgn{%
2135   -++\number\z@\number\z@
2136   \iftrue1\fi\iftrue2\fi\iftrue3\fi
2137 }{1}%
2138 \end{qstest}
2139
2140 \begin{qstest}{min}{min}%
2141 \TestMin{0}{1}{0}%
2142 \TestMin{1}{0}{0}%
2143 \TestMin{-10}{-20}{-20}%
2144 \TestMin{ 1 }{ 2 }{1}%
2145 \TestMin{ 2 }{ 1 }{1}%
2146 \TestMin{1}{1}{1}%
2147 \TestMin{\number\z@}{\number\@ne}{0}%
2148 \TestMin{\number\@ne}{\number\m@ne}{-1}%
2149 \end{qstest}
2150
2151 \begin{qstest}{max}{max}%
2152 \TestMax{0}{1}{1}%
2153 \TestMax{1}{0}{1}%
2154 \TestMax{-10}{-20}{-10}%
2155 \TestMax{ 1 }{ 2 }{2}%
2156 \TestMax{ 2 }{ 1 }{2}%
2157 \TestMax{1}{1}{1}%
2158 \TestMax{\number\z@}{\number\@ne}{1}%
2159 \TestMax{\number\@ne}{\number\m@ne}{1}%
2160 \end{qstest}
2161
2162 \begin{qstest}{cmp}{cmp}%
2163 \TestCmp{0}{0}{0}%
2164 \TestCmp{-21}{17}{-1}%
2165 \TestCmp{3}{4}{-1}%
2166 \TestCmp{-10}{-10}{0}%
2167 \TestCmp{-10}{-11}{1}%
2168 \TestCmp{100}{5}{1}%
2169 \TestCmp{9}{10}{-1}%
2170 \TestCmp{10}{9}{1}%
2171 \TestCmp{ 3 }{ 3 }{0}%
2172 \TestCmp{-9}{-10}{1}%
2173 \TestCmp{-10}{-9}{-1}%
2174 \TestCmp{-3}{-3}{0}%
2175 \TestCmp{0}{-2}{1}%
2176 \TestCmp{0}{2}{-1}%
2177 \TestCmp{2}{0}{1}%
2178 \TestCmp{-2}{0}{-1}%
2179 \TestCmp{12}{11}{1}%
2180 \TestCmp{11}{12}{-1}%
2181 \TestCmp{2147483647}{-2147483647}{1}%
2182 \TestCmp{-2147483647}{2147483647}{-1}%
2183 \TestCmp{2147483647}{2147483647}{0}%
2184 \TestCmp{\number\z@}{\number\@ne}{-1}%
2185 \TestCmp{\number\@ne}{\number\m@ne}{1}%
2186 \TestCmp{ 4 }{ 5 }{-1}%
2187 \TestCmp{ -3 }{ -7 }{1}%
2188 \end{qstest}
2189
2190 \begin{qstest}{odd}{odd}
2191 \tracingmacros=1

```

```

2192 \TestOdd{0}{0}%
2193 \TestOdd{1}{1}%
2194 \TestOdd{2}{0}%
2195 \TestOdd{3}{1}%
2196 \TestOdd{14}{0}%
2197 \TestOdd{15}{1}%
2198 \TestOdd{12345678901234567896}{0}%
2199 \TestOdd{12345678901234567897}{1}%
2200 \end{qstest}
2201
2202 \begin{qstest}{inc}{inc}%
2203 \TestInc{0}{1}%
2204 \TestInc{1}{2}%
2205 \TestInc{-1}{0}%
2206 \TestInc{10}{11}%
2207 \TestInc{-10}{-9}%
2208 \TestInc{ 3 }{4}%
2209 \TestInc{999}{1000}%
2210 \TestInc{-1000}{-999}%
2211 \TestInc{129}{130}%
2212 \TestInc{2147483646}{2147483647}%
2213 \TestInc{-2147483647}{-2147483646}%
2214 \TestInc{12345678901234567890}{12345678901234567891}%
2215 \TestInc{9999999999999999999}{10000000000000000000}%
2216 \TestInc{-12345678901234567891}{-12345678901234567890}%
2217 \TestInc{-10000000000000000000}{-9999999999999999999}%
2218 \end{qstest}
2219
2220 \begin{qstest}{dec}{dec}%
2221 \TestDec{0}{-1}%
2222 \TestDec{1}{0}%
2223 \TestDec{-1}{-2}%
2224 \TestDec{10}{9}%
2225 \TestDec{-10}{-11}%
2226 \TestDec{1000}{999}%
2227 \TestDec{-999}{-1000}%
2228 \TestDec{130}{129}%
2229 \TestDec{2147483647}{2147483646}%
2230 \TestDec{-2147483646}{-2147483647}%
2231 \TestDec{12345678901234567891}{12345678901234567890}%
2232 \TestDec{10000000000000000000}{9999999999999999999}%
2233 \TestDec{-12345678901234567890}{-12345678901234567891}%
2234 \TestDec{-9999999999999999999}{-10000000000000000000}%
2235 \end{qstest}
2236
2237 \begin{qstest}{add}{add}%
2238 \TestAdd{0}{0}{0}%
2239 \TestAdd{1}{0}{1}%
2240 \TestAdd{0}{1}{1}%
2241 \TestAdd{1}{2}{3}%
2242 \TestAdd{-1}{-1}{-2}%
2243 \TestAdd{2147483646}{1}{2147483647}%
2244 \TestAdd{-2147483647}{2147483647}{0}%
2245 \TestAdd{20}{-5}{15}%
2246 \TestAdd{-4}{-1}{-5}%
2247 \TestAdd{-1}{-4}{-5}%
2248 \TestAdd{-4}{1}{-3}%
2249 \TestAdd{-1}{4}{3}%
2250 \TestAdd{4}{-1}{3}%
2251 \TestAdd{1}{-4}{-3}%
2252 \TestAdd{-4}{-1}{-5}%
2253 \TestAdd{-1}{-4}{-5}%

```

```

2254 \TestAdd{ -4 }{ -1 }{-5}%
2255 \TestAdd{ -1 }{ -4 }{-5}%
2256 \TestAdd{ -4 }{ 1 }{-3}%
2257 \TestAdd{ -1 }{ 4 }{3}%
2258 \TestAdd{ 4 }{ -1 }{3}%
2259 \TestAdd{ 1 }{ -4 }{-3}%
2260 \TestAdd{ -4 }{ -1 }{-5}%
2261 \TestAdd{ -1 }{ -4 }{-5}%
2262 \TestAdd{876543210}{111111111}{987654321}%
2263 \TestAdd{999999999}{2}{1000000001}%
2264 \end{qstest}
2265
2266 \begin{qstest}{sub}{sub}
2267 \TestSub{0}{0}{0}%
2268 \TestSub{1}{0}{1}%
2269 \TestSub{1}{2}{-1}%
2270 \TestSub{-1}{-1}{0}%
2271 \TestSub{2147483646}{-1}{2147483647}%
2272 \TestSub{-2147483647}{-2147483647}{0}%
2273 \TestSub{-4}{-1}{-3}%
2274 \TestSub{-1}{-4}{3}%
2275 \TestSub{-4}{1}{-5}%
2276 \TestSub{-1}{4}{-5}%
2277 \TestSub{4}{-1}{5}%
2278 \TestSub{1}{-4}{5}%
2279 \TestSub{-4}{-1}{-3}%
2280 \TestSub{-1}{-4}{3}%
2281 \TestSub{ -4 }{ -1 }{-3}%
2282 \TestSub{ -1 }{ -4 }{3}%
2283 \TestSub{ -4 }{ 1 }{-5}%
2284 \TestSub{ -1 }{ 4 }{-5}%
2285 \TestSub{ 4 }{ -1 }{5}%
2286 \TestSub{ 1 }{ -4 }{5}%
2287 \TestSub{ -4 }{ -1 }{-3}%
2288 \TestSub{ -1 }{ -4 }{3}%
2289 \TestSub{1000000000}{2}{999999998}%
2290 \TestSub{987654321}{111111111}{876543210}%
2291 \end{qstest}
2292
2293 \begin{qstest}{shl}{shl}
2294 \TestShl{0}{0}%
2295 \TestShl{1}{2}%
2296 \TestShl{2}{4}%
2297 \TestShl{5621}{11242}%
2298 \TestShl{1073741823}{2147483646}%
2299 \end{qstest}
2300
2301 \begin{qstest}{shr}{shr}
2302 \TestShr{0}{0}%
2303 \TestShr{1}{0}%
2304 \TestShr{2}{1}%
2305 \TestShr{3}{1}%
2306 \TestShr{4}{2}%
2307 \TestShr{5}{2}%
2308 \TestShr{6}{3}%
2309 \TestShr{7}{3}%
2310 \TestShr{8}{4}%
2311 \TestShr{9}{4}%
2312 \TestShr{10}{5}%
2313 \TestShr{11}{5}%
2314 \TestShr{12}{6}%
2315 \TestShr{13}{6}%

```

```

2316 \TestShr{14}{7}%
2317 \TestShr{15}{7}%
2318 \TestShr{16}{8}%
2319 \TestShr{17}{8}%
2320 \TestShr{18}{9}%
2321 \TestShr{19}{9}%
2322 \TestShr{20}{10}%
2323 \TestShr{21}{10}%
2324 \TestShr{22}{11}%
2325 \TestShr{11241}{5620}%
2326 \TestShr{73054202}{36527101}%
2327 \TestShr{2147483646}{1073741823}%
2328 \end{qstest}
2329
2330 \begin{qstest}{mul}{mul}
2331 \TestMul{0}{0}{0}%
2332 \TestMul{1}{0}{0}%
2333 \TestMul{0}{1}{0}%
2334 \TestMul{1}{1}{1}%
2335 \TestMul{3}{1}{3}%
2336 \TestMul{1}{-3}{-3}%
2337 \TestMul{-4}{-5}{20}%
2338 \TestMul{3}{7}{21}%
2339 \TestMul{7}{3}{21}%
2340 \TestMul{3}{-7}{-21}%
2341 \TestMul{7}{-3}{-21}%
2342 \TestMul{-3}{7}{-21}%
2343 \TestMul{-7}{3}{-21}%
2344 \TestMul{-3}{-7}{21}%
2345 \TestMul{-7}{-3}{21}%
2346 \TestMul{12}{11}{132}%
2347 \TestMul{999}{333}{332667}%
2348 \TestMul{1000}{4321}{4321000}%
2349 \TestMul{12345}{173955}{2147474475}%
2350 \TestMul{1073741823}{2}{2147483646}%
2351 \TestMul{2}{1073741823}{2147483646}%
2352 \TestMul{-1073741823}{2}{-2147483646}%
2353 \TestMul{2}{-1073741823}{-2147483646}%
2354 \TestMul{6706022400}{13}{87178291200}%
2355 \end{qstest}
2356
2357 \begin{qstest}{sqr}{sqr}
2358 \TestSqr{0}{0}%
2359 \TestSqr{1}{1}%
2360 \TestSqr{2}{4}%
2361 \TestSqr{3}{9}%
2362 \TestSqr{4}{16}%
2363 \TestSqr{9}{81}%
2364 \TestSqr{10}{100}%
2365 \TestSqr{46340}{2147395600}%
2366 \TestSqr{-1}{1}%
2367 \TestSqr{-2}{4}%
2368 \TestSqr{-46340}{2147395600}%
2369 \end{qstest}
2370
2371 \begin{qstest}{fac}{fac}
2372 \TestFac{0}{1}%
2373 \TestFac{1}{1}%
2374 \TestFac{2}{2}%
2375 \TestFac{3}{2*3}%
2376 \TestFac{4}{2*3*4}%
2377 \TestFac{5}{2*3*4*5}%

```

```

2378 \TestFac{6}{2*3*4*5*6}%
2379 \TestFac{7}{2*3*4*5*6*7}%
2380 \TestFac{8}{2*3*4*5*6*7*8}%
2381 \TestFac{9}{2*3*4*5*6*7*8*9}%
2382 \TestFac{10}{2*3*4*5*6*7*8*9*10}%
2383 \TestFac{11}{2*3*4*5*6*7*8*9*10*11}%
2384 \TestFac{12}{2*3*4*5*6*7*8*9*10*11*12}%
2385 \TestFacBig{13}{6227020800}%
2386 \TestFacBig{14}{87178291200}%
2387 \TestFacBig{15}{1307674368000}%
2388 \TestFacBig{16}{20922789888000}%
2389 \TestFacBig{17}{355687428096000}%
2390 \TestFacBig{18}{6402373705728000}%
2391 \TestFacBig{19}{121645100408832000}%
2392 \TestFacBig{20}{2432902008176640000}%
2393 \TestFacBig{21}{51090942171709440000}%
2394 \TestFacBig{22}{112400072777607680000}%
2395 \end{qstest}
2396
2397 \begin{qstest}{pow}{pow}
2398 \TestPow{-2}{0}{1}%
2399 \TestPow{-1}{0}{1}%
2400 \TestPow{0}{0}{1}%
2401 \TestPow{1}{0}{1}%
2402 \TestPow{2}{0}{1}%
2403 \TestPow{3}{0}{1}%
2404 \TestPow{-2}{1}{-2}%
2405 \TestPow{-1}{1}{-1}%
2406 \TestPow{1}{1}{1}%
2407 \TestPow{2}{1}{2}%
2408 \TestPow{3}{1}{3}%
2409 \TestPow{-2}{2}{4}%
2410 \TestPow{-1}{2}{1}%
2411 \TestPow{0}{2}{0}%
2412 \TestPow{1}{2}{1}%
2413 \TestPow{2}{2}{4}%
2414 \TestPow{3}{2}{9}%
2415 \TestPow{0}{1}{0}%
2416 \TestPow{1}{-2}{1}%
2417 \TestPow{1}{-1}{1}%
2418 \TestPow{-1}{-2}{1}%
2419 \TestPow{-1}{-1}{-1}%
2420 \TestPow{-1}{3}{-1}%
2421 \TestPow{-1}{4}{1}%
2422 \TestPow{-2}{-1}{0}%
2423 \TestPow{-2}{-2}{0}%
2424 \TestPow{2}{3}{8}%
2425 \TestPow{2}{4}{16}%
2426 \TestPow{2}{5}{32}%
2427 \TestPow{2}{6}{64}%
2428 \TestPow{2}{7}{128}%
2429 \TestPow{2}{8}{256}%
2430 \TestPow{2}{9}{512}%
2431 \TestPow{2}{10}{1024}%
2432 \TestPow{-2}{3}{-8}%
2433 \TestPow{-2}{4}{16}%
2434 \TestPow{-2}{5}{-32}%
2435 \TestPow{-2}{6}{64}%
2436 \TestPow{-2}{7}{-128}%
2437 \TestPow{-2}{8}{256}%
2438 \TestPow{-2}{9}{-512}%
2439 \TestPow{-2}{10}{1024}%

```

```

2440 \TestPow{3}{3}{27}%
2441 \TestPow{3}{4}{81}%
2442 \TestPow{3}{5}{243}%
2443 \TestPow{-3}{3}{-27}%
2444 \TestPow{-3}{4}{81}%
2445 \TestPow{-3}{5}{-243}%
2446 \TestPow{2}{30}{1073741824}%
2447 \TestPow{-3}{19}{-1162261467}%
2448 \TestPow{5}{13}{1220703125}%
2449 \TestPow{-7}{11}{-1977326743}%
2450 \end{qstest}
2451
2452 \begin{qstest}{div}{div}
2453 \TestDiv{1}{1}{1}%
2454 \TestDiv{2}{1}{2}%
2455 \TestDiv{-2}{1}{-2}%
2456 \TestDiv{2}{-1}{-2}%
2457 \TestDiv{-2}{-1}{2}%
2458 \TestDiv{15}{2}{7}%
2459 \TestDiv{-16}{2}{-8}%
2460 \TestDiv{1}{2}{0}%
2461 \TestDiv{1}{3}{0}%
2462 \TestDiv{2}{3}{0}%
2463 \TestDiv{-2}{3}{0}%
2464 \TestDiv{2}{-3}{0}%
2465 \TestDiv{-2}{-3}{0}%
2466 \TestDiv{13}{3}{4}%
2467 \TestDiv{-13}{-3}{4}%
2468 \TestDiv{-13}{3}{-4}%
2469 \TestDiv{-6}{5}{-1}%
2470 \TestDiv{-5}{5}{-1}%
2471 \TestDiv{-4}{5}{0}%
2472 \TestDiv{-3}{5}{0}%
2473 \TestDiv{-2}{5}{0}%
2474 \TestDiv{-1}{5}{0}%
2475 \TestDiv{0}{5}{0}%
2476 \TestDiv{1}{5}{0}%
2477 \TestDiv{2}{5}{0}%
2478 \TestDiv{3}{5}{0}%
2479 \TestDiv{4}{5}{0}%
2480 \TestDiv{5}{5}{1}%
2481 \TestDiv{6}{5}{1}%
2482 \TestDiv{-5}{4}{-1}%
2483 \TestDiv{-4}{4}{-1}%
2484 \TestDiv{-3}{4}{0}%
2485 \TestDiv{-2}{4}{0}%
2486 \TestDiv{-1}{4}{0}%
2487 \TestDiv{0}{4}{0}%
2488 \TestDiv{1}{4}{0}%
2489 \TestDiv{2}{4}{0}%
2490 \TestDiv{3}{4}{0}%
2491 \TestDiv{4}{4}{1}%
2492 \TestDiv{5}{4}{1}%
2493 \TestDiv{12345}{678}{18}%
2494 \TestDiv{32372}{5952}{5}%
2495 \TestDiv{284271294}{18162}{15651}%
2496 \TestDiv{217652429}{12561}{17327}%
2497 \TestDiv{462028434}{5439}{84947}%
2498 \TestDiv{2147483647}{1000}{2147483}%
2499 \TestDiv{2147483647}{-1000}{-2147483}%
2500 \TestDiv{-2147483647}{1000}{-2147483}%
2501 \TestDiv{-2147483647}{-1000}{2147483}%

```



```

2502 \TestDiv{0}{3}{0}%
2503 \TestDiv{1}{3}{0}%
2504 \TestDiv{2}{3}{0}%
2505 \TestDiv{3}{3}{1}%
2506 \TestDiv{4}{3}{1}%
2507 \TestDiv{5}{3}{1}%
2508 \TestDiv{6}{3}{2}%
2509 \TestDiv{7}{3}{2}%
2510 \TestDiv{8}{3}{2}%
2511 \TestDiv{9}{3}{3}%
2512 \TestDiv{10}{3}{3}%
2513 \TestDiv{11}{3}{3}%
2514 \TestDiv{12}{3}{4}%
2515 \TestDiv{13}{3}{4}%
2516 \TestDiv{14}{3}{4}%
2517 \TestDiv{15}{3}{5}%
2518 \TestDiv{16}{3}{5}%
2519 \TestDiv{17}{3}{5}%
2520 \TestDiv{18}{3}{6}%
2521 \TestDiv{19}{3}{6}%
2522 \TestDiv{20}{3}{6}%
2523 \TestDiv{21}{3}{7}%
2524 \TestDiv{22}{3}{7}%
2525 \TestDiv{23}{3}{7}%
2526 \TestDiv{24}{3}{8}%
2527 \TestDiv{25}{3}{8}%
2528 \TestDiv{26}{3}{8}%
2529 \TestDiv{27}{3}{9}%
2530 \TestDiv{28}{3}{9}%
2531 \TestDiv{29}{3}{9}%
2532 \TestDiv{30}{3}{10}%
2533 \TestDiv{31}{3}{10}%
2534 \TestDivBig{17363436332507}{24702}{702916214}%
2535 \end{qstest}
2536
2537 \begin{qstest}{mod}{mod}
2538 \TestMod{-6}{5}{4}%
2539 \TestMod{-5}{5}{0}%
2540 \TestMod{-4}{5}{1}%
2541 \TestMod{-3}{5}{2}%
2542 \TestMod{-2}{5}{3}%
2543 \TestMod{-1}{5}{4}%
2544 \TestMod{0}{5}{0}%
2545 \TestMod{1}{5}{1}%
2546 \TestMod{2}{5}{2}%
2547 \TestMod{3}{5}{3}%
2548 \TestMod{4}{5}{4}%
2549 \TestMod{5}{5}{0}%
2550 \TestMod{6}{5}{1}%
2551 \TestMod{-5}{4}{3}%
2552 \TestMod{-4}{4}{0}%
2553 \TestMod{-3}{4}{1}%
2554 \TestMod{-2}{4}{2}%
2555 \TestMod{-1}{4}{3}%
2556 \TestMod{0}{4}{0}%
2557 \TestMod{1}{4}{1}%
2558 \TestMod{2}{4}{2}%
2559 \TestMod{3}{4}{3}%
2560 \TestMod{4}{4}{0}%
2561 \TestMod{5}{4}{1}%
2562 \TestMod{-6}{-5}{-1}%
2563 \TestMod{-5}{-5}{0}%

```

```

2564 \TestMod{-4}{-5}{-4}%
2565 \TestMod{-3}{-5}{-3}%
2566 \TestMod{-2}{-5}{-2}%
2567 \TestMod{-1}{-5}{-1}%
2568 \TestMod{0}{-5}{0}%
2569 \TestMod{1}{-5}{-4}%
2570 \TestMod{2}{-5}{-3}%
2571 \TestMod{3}{-5}{-2}%
2572 \TestMod{4}{-5}{-1}%
2573 \TestMod{5}{-5}{0}%
2574 \TestMod{6}{-5}{-4}%
2575 \TestMod{-5}{-4}{-1}%
2576 \TestMod{-4}{-4}{0}%
2577 \TestMod{-3}{-4}{-3}%
2578 \TestMod{-2}{-4}{-2}%
2579 \TestMod{-1}{-4}{-1}%
2580 \TestMod{0}{-4}{0}%
2581 \TestMod{1}{-4}{-3}%
2582 \TestMod{2}{-4}{-2}%
2583 \TestMod{3}{-4}{-1}%
2584 \TestMod{4}{-4}{0}%
2585 \TestMod{5}{-4}{-3}%
2586 \TestMod{2147483647}{1000}{647}%
2587 \TestMod{2147483647}{-1000}{-353}%
2588 \TestMod{-2147483647}{1000}{353}%
2589 \TestMod{-2147483647}{-1000}{-647}%
2590 \TestMod{ 0 }{ 4 }{0}%
2591 \TestMod{ 1 }{ 4 }{1}%
2592 \TestMod{ -1 }{ 4 }{3}%
2593 \TestMod{ 0 }{ -4 }{0}%
2594 \TestMod{ 1 }{ -4 }{-3}%
2595 \TestMod{ -1 }{ -4 }{-1}%
2596 \TestMod{18362}{25}{12}%
2597 \end{qstest}
2598
2599 \newcommand*{\TestError}[2]{%
2600 \begin{group}
2601 \expandafter\def\csname BigIntCalcError:#1\endcsname{%
2602 \Expect*{#2}{0}%
2603 \expandafter\def\csname BigIntCalcError:#1\endcsname{ERROR}%
2604 \Expect*{#2}{0ERROR}%
2605 \end{group}
2606 }
2607 \begin{qstest}{error}{error}
2608 \TestError{FacNegative}{\bigintcalcFac{-1}}%
2609 \TestError{FacNegative}{\bigintcalcFac{-2147483647}}%
2610 \TestError{DivisionByZero}{\bigintcalcPow{0}{-1}}%
2611 \TestError{DivisionByZero}{\bigintcalcDiv{1}{0}}%
2612 \TestError{DivisionByZero}{\bigintcalcMod{1}{0}}%
2613 \end{qstest}
2614
2615 \begin{document}
2616 \end{document}
2617 \end{test2}

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/bigintcalc.dtx](http://ftp.ctan.org/contrib/oberdiek/bigintcalc.dtx) The source file.

¹<http://ftp.ctan.org/tex-archive/>

[CTAN:macros/latex/contrib/oberdiek/bigintcalc.pdf](#) Documentation.

Bundle. All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

TDS refers to the standard “A Directory Structure for T_EX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `oberdiek-tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek-tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain-T_EX:

```
tex bigintcalc.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

<code>bigintcalc.sty</code>	→ <code>tex/generic/oberdiek/bigintcalc.sty</code>
<code>bigintcalc.pdf</code>	→ <code>doc/latex/oberdiek/bigintcalc.pdf</code>
<code>test/bigintcalc-test1.tex</code>	→ <code>doc/latex/oberdiek/test/bigintcalc-test1.tex</code>
<code>test/bigintcalc-test2.tex</code>	→ <code>doc/latex/oberdiek/test/bigintcalc-test2.tex</code>
<code>test/bigintcalc-test3.tex</code>	→ <code>doc/latex/oberdiek/test/bigintcalc-test3.tex</code>
<code>bigintcalc.dtx</code>	→ <code>source/latex/oberdiek/bigintcalc.dtx</code>

If you have a `docstrip.cfg` that configures and enables `docstrip`’s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

4.4 Refresh file name databases

If your T_EX distribution (teT_EX, mikT_EX, ...) relies on file name databases, you must refresh these. For example, teT_EX users run `texhash` or `mktextlsr`.

4.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk bigintcalc.pdf unpack_files output .
```

Unpacking with L^AT_EX. The .dtx chooses its action depending on the format:

plain-T_EX: Run docstrip and extract the files.

L^AT_EX: Generate the documentation.

If you insist on using L^AT_EX for docstrip (really, docstrip does not need L^AT_EX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{bigintcalc.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the .dtx or the .drv to generate the documentation. The process can be configured by the configuration file ltxdoc.cfg. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfL^AT_EX:

```
pdflatex bigintcalc.dtx
makeindex -s gind.ist bigintcalc.idx
pdflatex bigintcalc.dtx
makeindex -s gind.ist bigintcalc.idx
pdflatex bigintcalc.dtx
```

5 History

[2007/09/27 v1.0]

- First version.

[2007/11/11 v1.1]

- Use of package pdftexcmds for L^AT_EX support.

6 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols		
\#	1749, 1808	1196, 1221, 1249, 1297, 1318, 1328, 1422, 1426, 1454, 1500,
\%	1811	1647, 1658, 1666, 1688, 1704, 1805
\@	1750, 1804	\{ 1747, 1806
\@firstofone	1758, 1761	\} 1748, 1807
\@firstoftwo	134, 142, <u>151</u>	\] 1810
\@gobble	1755, 1763	
\@ne	2132,	
	2147, 2148, 2158, 2159, 2184, 2185	_ 1812
\@nil	130, 132, 139, 147, 162, 165, 170	
\@secondoftwo	136, 144, <u>154</u>	A
\[1809	\advance 1788, 1796, 2060
\	167, 177, 296, 297, 305, 328, 426, 438, 471, 484, 496, 531, 533, 622, 623, 633, 634, 651, 713, 714, 724, 725, 742, 827, 838, 885, 899, 962, 973, 990, 1110,	\AtEndDocument 2073
		B
		\begin 2080, 2098, 2117, 2140, 2151, 2162, 2190, 2202, 2220, 2237,

2266, 2293, 2301, 2330, 2357,
 2371, 2397, 2452, 2537, 2607, 2615
 \BIC@@@Shl 845, 849
 \BIC@@@Shr ... 918, 919, 925, 926, 935
 \BIC@@@Dec 506, 507, 525, 530
 \BIC@@@Inc 448, 449, 465, 470
 \BIC@@@PowRec 1348, 1354, 1369
 \BIC@@@ProcessDiv 1553, 1591
 \BIC@@@Shl 829, 837, 852, 856
 \BIC@@@Shr 889, 894, 898, 936
 \BIC@@AddDigit 674, 675, 685, 764
 \BIC@@Cmp 266, 269, 355, 387
 \BIC@@Dec 486, 494, 543
 \BIC@@Expand 130, 132
 \BIC@@Inc 428, 436, 479
 \BIC@@MinMax 351, 354
 \BIC@@MinusOne 1311, 1316
 \BIC@@PowRec 1341, 1364, 1370
 \BIC@@ProcessDiv 1512, 1517
 \BIC@@ProcessFac 1175, 1181
 \BIC@@ProcessTim 982, 989
 \BIC@@Shl 811, 816, 823, 826
 \BIC@@Shr 869, 873, 879,
 881, 1342, 1371, 1438, 1442, 1443
 \BIC@@Sqr 1129, 1131, 1134
 \BIC@@SubDigit 763, 774
 \BIC@@TestMode 104
 \BIC@@Tim 961
 \BIC@Abs 233, 236
 \BIC@Add 549, 552, 558
 \BIC@AddCarry0 692
 \BIC@AddCarry10 693
 \BIC@AddCarry[1-9] 694
 \BIC@AddDigit 656, 661, 672
 \BIC@AddResult 655, 665
 \BIC@AddSwitch 554, 561
 \BIC@AddXY 567, 571, 605, 609,
 616, 621, 812, 817, 824, 1018, 1476
 \BIC@AfterFi 112,
 169, 206, 319, 356, 358, 388,
 390, 394, 407, 409, 413, 427,
 431, 478, 485, 489, 542, 652,
 659, 687, 689, 743, 748, 776,
 781, 809, 815, 828, 832, 839,
 844, 851, 855, 868, 872, 883,
 893, 963, 967, 974, 981, 991,
 995, 1002, 1004, 1052, 1111,
 1116, 1144, 1172, 1174, 1207,
 1209, 1304, 1389, 1418, 1420,
 1455, 1459, 1511, 1552, 1601,
 1656, 1695, 1729, 1731, 1736, 1741
 \BIC@AfterFiFi
 . 113, 178, 182, 188, 209, 213,
 272, 276, 282, 286, 298, 300,
 306, 310, 324, 373, 375, 439,
 441, 447, 464, 473, 475, 497,
 499, 505, 524, 539, 624, 628,
 644, 715, 719, 735, 886, 888,
 901, 908, 917, 924, 1010, 1015,
 1055, 1080, 1164, 1212, 1216,
 1234, 1236, 1245, 1265, 1267,
 1288, 1299, 1301, 1330, 1334,
 1340, 1392, 1394, 1398, 1404,
 1406, 1410, 1435, 1437, 1441,
 1446, 1490, 1492, 1498, 1520,
 1522, 1544, 1567, 1569, 1575,
 1577, 1583, 1585, 1594, 1596,
 1617, 1624, 1659, 1661, 1675,
 1677, 1681, 1698, 1700, 1710, 1717
 \BIC@AfterFiFiFi 114, 193, 197, 329,
 333, 534, 536, 566, 570, 576,
 579, 583, 591, 593, 598, 604,
 608, 635, 639, 726, 730, 1058,
 1062, 1069, 1073, 1084, 1088,
 1094, 1098, 1148, 1149, 1150,
 1151, 1152, 1153, 1154, 1155,
 1156, 1157, 1158, 1159, 1160,
 1239, 1241, 1270, 1272, 1277,
 1282, 1347, 1353, 1501, 1505,
 1529, 1531, 1537, 1539, 1610, 1612
 \BIC@AtEnd 78, 79, 1744
 \BIC@Cmp 262, 265
 \BIC@CmpDiff 298, 317
 \BIC@CmpLength 287, 293, 295
 \BIC@CmpResult 301, 307, 316
 \BIC@Dec 396, 410, 423, 483, 1183
 \BIC@DecSwitch 402, 405
 \BIC@Div 1376, 1379
 \BIC@DivCleanup
 . 1490, 1498, 1516, 1520, 1529,
 1537, 1567, 1575, 1583, 1594, 1610
 \BIC@DivStart 1480, 1484
 \BIC@DivStartX 1447, 1453, 1460
 \BIC@DivStartYii 1456, 1464
 \BIC@DivStartYiv 1465, 1469
 \BIC@DivStartYvi 1470, 1474
 \BIC@DivStartYviii 1475, 1479
 \BIC@DivSub 1540, 1557, 1586
 \BIC@DivSwitch
 1395, 1399, 1407, 1411, 1416
 \BIC@DivSwitchSign .. 1380, 1385, 1387
 \BIC@DoAdd 625, 629, 650
 \BIC@DoSub 716, 720, 741
 \BIC@Expand 128, 158, 225
 \BIC@Fac 1139, 1142
 \BIC@Fi 111, 112, 113, 114,
 172, 202, 217, 290, 314, 338,
 359, 379, 398, 417, 434, 468,
 481, 492, 528, 545, 613, 648,
 663, 690, 739, 752, 782, 819,
 835, 847, 858, 875, 896, 931,
 970, 987, 998, 1023, 1103, 1120,
 1168, 1179, 1294, 1307, 1362,
 1414, 1451, 1462, 1514, 1555,
 1589, 1631, 1685, 1725, 1732, 1742
 \BIC@Inc 391, 415, 420, 425
 \BIC@IncSwitch 383, 386
 \BIC@MinMax 342, 347, 350
 \BIC@MinusOne 1227, 1256, 1309
 \BIC@Mod 1635, 1638
 \BIC@ModMinus 1707, 1727
 \BIC@ModSwitch . 1662, 1678, 1682, 1687
 \BIC@ModSwitchSign .. 1639, 1644, 1646

\BIC@ModTwo	1238, 1269, 1276, <u>1296</u> , 1339, 1697	\bigintcalcAbs .. 4, <u>231</u> , 364, 1878,	1897, 1968, 1969, 1977, 1978,
\BIC@ModX	1711, <u>1734</u>	1987, 1988, 2014, 2015, 2035, 2036	
\BIC@Mul	1043, <u>1046</u>	\BigIntCalcAdd 7, <u>615</u> , 1931, 1939	
\BIC@MulDigit[3-9]	<u>1025</u>	\bigintcalcAdd 5, <u>547</u> , 1113, 1118, 1925	
\BIC@MulSwitch	1047, <u>1050</u>	\bigintcalcCmp 4, <u>260</u> , 1890, 1928, 1952	
\BIC@Normalize	<u>175</u> , <u>222</u>	\BigIntCalcDec	7, <u>422</u> , 1918
\BIC@NormalizeDigits ..	198, 214, <u>219</u>	\bigintcalcDec	5, <u>400</u> , 1914
\BIC@NormalizeZero	194, <u>204</u>	\BigIntCalcDiv	7, <u>1383</u> , 2014
\BIC@Odd	363, 368, <u>370</u>	\bigintcalcDiv	6, <u>1374</u> ,
\BIC@PosCmp	<u>292</u> , 565, 575, 590,	1713, 1720, 1866, 2007, 2011, 2611	
603, 1057, 1068, 1083, 1093,		\BigIntCalcError	459,
1146, 1171, 1329, 1346, 1417,		509, 519, 698, 789, 1030, 1144,	
1488, 1518, 1527, 1565, 1592, 1608		1161, 1231, 1234, 1242, 1246,	
\BIC@Pow	1188, <u>1191</u>	1273, 1285, 1291, 1293, 1359,	
\BIC@PowRec 1278, 1283, 1289, <u>1327</u> , <u>1365</u>		1361, 1389, 1449, 1656, 1701, 1724	
\BIC@PowSwitch	1192, <u>1195</u>	\bigintcalcFac	
\BIC@ProcessDiv 6, <u>1137</u> , 1998, 2001, 2608, 2609	
..... <u>1453</u> , <u>1485</u> , <u>1487</u> , 1523,		\BigIntCalcInc	7, <u>419</u> , 1907
1532, 1558, 1570, 1578, 1597, 1613		\bigintcalcInc	5, <u>381</u> , 1903
\BIC@ProcessDivII		\bigintcalcInv	3, <u>227</u> , 1875
..... <u>1545</u> , <u>1564</u> , 1602, 1618, 1625		\bigintcalcMax	4, <u>345</u> , 1887
\BIC@ProcessDivIV	<u>1591</u>	\bigintcalcMin	4, <u>340</u> , 1884
\BIC@ProcessFac	1165, <u>1170</u> , 1182	\BigIntCalcMod	7, <u>1642</u> , 2034
\BIC@ProcessMul .	1059, 1063, 1070,	\bigintcalcMod ...	6, <u>1633</u> , 2019, 2612
1074, 1085, 1089, 1095, 1099,		\BigIntCalcMul	7, <u>1105</u> , 1986
1107, <u>1109</u> , 1135, 1176, 1213,		\bigintcalcMul 6, <u>1041</u> , 1713, 1720, 1983	
1217, 1331, 1335, 1349, 1355, 1366		\bigintcalcNum ..	3, <u>220</u> , 229, 234,
\BIC@ProcessTim ...	964, <u>972</u> , 992, 996	247, 263, 267, 343, 348, 352,	
\BIC@Sgn	246, <u>249</u> , 406,	384, 403, 550, 554, 559, 805,	
1051, 1054, 1079, 1388, 1391, 1403		863, 1044, 1048, 1125, 1140,	
\BIC@Shl ...	804, <u>807</u> , 1466, 1471, 1481	1189, 1193, 1377, 1381, 1636,	
\BIC@Shr	862, <u>865</u>	1640, 1907, 1918, 1932, 1940, 1956	
\BIC@ShrDigit[00-19]	<u>938</u>	\BigIntCalcOdd	7, <u>366</u> , 1896
\BIC@ShrResult 902, 903, 909, 910, <u>933</u>		\bigintcalcOdd	5, <u>361</u> , 1893
\BIC@Space	<u>115</u> ,	\bigintcalcPow ...	6, <u>1186</u> , 2004, 2610
159, 228, 238, 240, 653, 667,		\bigintcalcSgn	
669, 758, 777, 840, 975, 1005,		4, <u>244</u> , 1881, 1904, 1915, 1926,	
1011, 1016, 1112, 1706, 1718, 1737		1927, 1950, 1951, 2020, 2022, 2027	
\BIC@Sqr	1124, <u>1127</u>	\BigIntCalcShl	7, <u>821</u> , 1968
\BIC@StripHexSpace	162, <u>165</u>	\bigintcalcShl	5, <u>802</u> , 1965
\BIC@SubCarry0	<u>784</u>	\BigIntCalcShr	7, <u>877</u> , 1977
\BIC@SubCarry10	<u>785</u>	\bigintcalcShr	5, <u>860</u> , 1974
\BIC@SubCarry[1-9]	<u>786</u>	\bigintcalcSqr	6, <u>1122</u> , 1993
\BIC@SubDigit	745, 750, <u>761</u>	\BigIntCalcSub	7, <u>618</u> , 1955
\BIC@SubResult	744, <u>754</u>	\bigintcalcSub 5, <u>556</u> , 1712, 1719, 1949	
\BIC@SubXY		\body	1767, 1771
.. 580, 584, 594, 599, 619, <u>712</u> ,			
1546, 1560, 1603, 1619, 1626, 1738			
\BIC@Temp	694, 702,		
703, 704, 705, 706, 707, 708,			
709, 710, 711, 786, 793, 794,			
795, 796, 797, 798, 799, 800,			
801, 938, 941, 942, 943, 944,			
945, 946, 947, 948, 949, 950,			
951, 952, 953, 954, 955, 956,			
957, 958, 959, 960, 1025, 1034,			
1035, 1036, 1037, 1038, 1039, 1040			
\BIC@TestMode	104, 1828		
\BIC@Tim	<u>961</u> , 1113, 1118		
\BIC@TimDigit	977, 984, <u>1000</u>		

C	
\catcode ...	3, 4, 5, 6, 7, 18, 19, 20,
	34, 35, 36, 37, 38, 39, 40, 41, 42,
	43, 44, 62, 63, 66, 67, 68, 69, 73,
	74, 75, 76, 80, 82, 102, 107, 109,
	1747, 1748, 1749, 1750, 1785,
	1794, 1804, 1805, 1806, 1807,
	1808, 1809, 1810, 1811, 1812, 1813
\chardef	1828
\count@	1752, 1781, 1785,
	1787, 1788, 1792, 1794, 1795, 1796
\countdef	1752
\csname .	8, 21, 45, 58, 65, 100, 106,
	121, 127, 151, 154, 676, 692,
	693, 695, 765, 784, 785, 787,

\romannumeral 129, 159, 221, 228, 232,
 341, 346, 362, 367, 382, 395,
 401, 414, 420, 423, 548, 557,
 564, 578, 597, 616, 619, 673,
 762, 803, 810, 822, 861, 867,
 878, 977, 984, 1017, 1042, 1067,
 1082, 1106, 1123, 1138, 1176,
 1183, 1187, 1281, 1341, 1348,
 1354, 1365, 1370, 1375, 1384,
 1424, 1442, 1465, 1470, 1475,
 1480, 1545, 1559, 1602, 1618,
 1625, 1634, 1643, 1706, 1707, 1737

S

\saved@endqstest 2064, 2071
 \saved@qstest 2063, 2066
 \SavedNumexpr
 1825, 1831, 1836, 1840, 1844, 1847
 \space 1841, 2050
 \StartTime 2053, 2067
 \StopTime 2058, 2070
 \strip@pt 2050
 \SummaryTime ... 2045, 2047, 2060, 2074

T

\Test 1799, 1817, 1868, 1873,
 1875, 1878, 1881, 1884, 1887,
 1890, 1893, 1895, 1903, 1906,
 1914, 1917, 1925, 1930, 1938,
 1949, 1954, 1965, 1967, 1974,
 1976, 1983, 1985, 1993, 2001,
 2004, 2007, 2011, 2013, 2019, 2033
 \TestAbs 1877,
 2099, 2100, 2101, 2102, 2103,
 2104, 2105, 2106, 2107, 2108,
 2109, 2110, 2111, 2112, 2113, 2114
 \TestAdd 1924,
 2238, 2239, 2240, 2241, 2242,
 2243, 2244, 2245, 2246, 2247,
 2248, 2249, 2250, 2251, 2252,
 2253, 2254, 2255, 2256, 2257,
 2258, 2259, 2260, 2261, 2262, 2263
 \TestArg 1861, 1862, 1864, 1865
 \TestCmp 1889, 2163, 2164, 2165, 2166,
 2167, 2168, 2169, 2170, 2171,
 2172, 2173, 2174, 2175, 2176,
 2177, 2178, 2179, 2180, 2181,
 2182, 2183, 2184, 2185, 2186, 2187
 \TestCount 1860, 1864, 1865, 1866
 \TestDec .. 1913, 2221, 2222, 2223,
 2224, 2225, 2226, 2227, 2228,
 2229, 2230, 2231, 2232, 2233, 2234
 \TestDiv 2006,
 2453, 2454, 2455, 2456, 2457,
 2458, 2459, 2460, 2461, 2462,
 2463, 2464, 2465, 2466, 2467,
 2468, 2469, 2470, 2471, 2472,
 2473, 2474, 2475, 2476, 2477,
 2478, 2479, 2480, 2481, 2482,
 2483, 2484, 2485, 2486, 2487,
 2488, 2489, 2490, 2491, 2492,
 2493, 2494, 2495, 2496, 2497,
 2498, 2499, 2500, 2501, 2502,
 2503, 2504, 2505, 2506, 2507,
 2508, 2509, 2510, 2511, 2512,
 2513, 2514, 2515, 2516, 2517,
 2518, 2519, 2520, 2521, 2522,
 2523, 2524, 2525, 2526, 2527,
 2528, 2529, 2530, 2531, 2532, 2533
 \TestDivBig 2010, 2534
 \TestError
 2599, 2608, 2609, 2610, 2611, 2612
 \TestExch 1873, 1996
 \TestFac 1995, 2372, 2373,
 2374, 2375, 2376, 2377, 2378,
 2379, 2380, 2381, 2382, 2383, 2384
 \TestFacBig
 . 2000, 2385, 2386, 2387, 2388,
 2389, 2390, 2391, 2392, 2393, 2394
 \TestInc 1902, 2203, 2204, 2205, 2206,
 2207, 2208, 2209, 2210, 2211,
 2212, 2213, 2214, 2215, 2216, 2217
 \TestInv 1874, 2081, 2082, 2083, 2084,
 2085, 2086, 2087, 2088, 2089,
 2090, 2091, 2092, 2093, 2094, 2095
 \TestMax 1886, 2152, 2153,
 2154, 2155, 2156, 2157, 2158, 2159
 \TestMin 1883, 2141, 2142,
 2143, 2144, 2145, 2146, 2147, 2148
 \TestMod .. 2018, 2538, 2539, 2540,
 2541, 2542, 2543, 2544, 2545,
 2546, 2547, 2548, 2549, 2550,
 2551, 2552, 2553, 2554, 2555,
 2556, 2557, 2558, 2559, 2560,
 2561, 2562, 2563, 2564, 2565,
 2566, 2567, 2568, 2569, 2570,
 2571, 2572, 2573, 2574, 2575,
 2576, 2577, 2578, 2579, 2580,
 2581, 2582, 2583, 2584, 2585,
 2586, 2587, 2588, 2589, 2590,
 2591, 2592, 2593, 2594, 2595, 2596
 \TestMul .. 1982, 2331, 2332, 2333,
 2334, 2335, 2336, 2337, 2338,
 2339, 2340, 2341, 2342, 2343,
 2344, 2345, 2346, 2347, 2348,
 2349, 2350, 2351, 2352, 2353, 2354
 \TestOdd 1892, 2192, 2193,
 2194, 2195, 2196, 2197, 2198, 2199
 \TestPow 2003, 2398,
 2399, 2400, 2401, 2402, 2403,
 2404, 2405, 2406, 2407, 2408,
 2409, 2410, 2411, 2412, 2413,
 2414, 2415, 2416, 2417, 2418,
 2419, 2420, 2421, 2422, 2423,
 2424, 2425, 2426, 2427, 2428,
 2429, 2430, 2431, 2432, 2433,
 2434, 2435, 2436, 2437, 2438,
 2439, 2440, 2441, 2442, 2443,
 2444, 2445, 2446, 2447, 2448, 2449
 \TestResult 1843, 1869
 \TestResultTwoExpansions . 1850, 1870
 \TestSgn 1880, 2118,
 2119, 2120, 2121, 2122, 2123,
 2124, 2125, 2126, 2127, 2128,
 2129, 2130, 2131, 2132, 2133, 2134

<code>\TestShl</code>	<code>\TimeDescription</code> . . .
1964, 2294, 2295, 2296, 2297, 2298	2054, 2057, 2061
<code>\TestShr</code>	<code>\TMP@EnsureCode</code> 77, 84, 85, 86, 87, 88,
1973,	89, 90, 91, 92, 93, 94, 95, 96, 97, 98
2302, 2303, 2304, 2305, 2306,	<code>\tracingmacros</code>
2307, 2308, 2309, 2310, 2311,	2191
2312, 2313, 2314, 2315, 2316,	<code>\typeout</code>
2317, 2318, 2319, 2320, 2321,	2049
2322, 2323, 2324, 2325, 2326, 2327	
<code>\TestSpaceAtEnd</code>	U
1835, 1871	<code>\UNDEFINED</code>
<code>\TestSqr</code>	1826, 1837, 1845, 1853
1992,	<code>\usepackage</code>
2358, 2359, 2360, 2361, 2362,	1830, 1832
2363, 2364, 2365, 2366, 2367, 2368	
<code>\TestSub</code>	W
1948, 2267, 2268, 2269,	<code>\write</code>
2270, 2271, 2272, 2273, 2274,	23, 47
2275, 2276, 2277, 2278, 2279,	
2280, 2281, 2282, 2283, 2284,	X
2285, 2286, 2287, 2288, 2289, 2290	<code>\x</code>
<code>\TestTeXDivide</code>	8, 10, 12, 22, 26,
1863, 2008	28, 46, 51, 58, 64, 72, 116, 119,
<code>\TestTime</code>	1894, 1900, 1905, 1910, 1916,
2046, 2059, 2060, 2061	1921, 1929, 1935, 1937, 1943,
<code>\the</code>	1953, 1959, 1966, 1971, 1975,
66, 67,	1980, 1984, 1990, 2012, 2032, 2038
68, 69, 80, 448, 506, 654, 674,	
763, 778, 841, 845, 902, 909,	Z
918, 925, 976, 983, 1785, 1866, 1997	<code>\z@</code>
	2047,
	2093, 2131, 2135, 2147, 2158, 2184