

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	10
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	19
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	46
3.2.1	Preamble with test macro definitions	46
3.2.2	Time	49
3.2.3	Test sets	50
4	Installation	59
4.1	Download	59
4.2	Bundle installation	59
4.3	Package installation	59
4.4	Refresh file name databases	59
4.5	Some details for the interested	60
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 expandable 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\undefined
54 \xdef#1{#3}%
55 \fi
56 \ifx#1\relax
```

```

57      \xdef#1{#3}%
58      \fi
59  }%
60  \fi
61  \expandafter\x\csname ver@bigintcalc.sty\endcsname
62  \ProvidesPackage{bigintcalc}%
63  [2007/11/11 v1.1 Expandable big integer calculations (H0)]

```

2.2 Catcodes

```

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

```

2.3 ϵ -TeX detection

```

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

```

2.4 Help macros

```

\BIC@Fi
114 \let\BIC@Fi\fi

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

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

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

\BIC@Space
118 \begingroup
119   \def\x#1{\endgroup
120     \let\BIC@Space= #1%
121   }%
122 \x{ }

```

2.5 Expand number

```

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

129 \begingroup\expandafter\expandafter\expandafter\endgroup
130 \expandafter\ifx\csname pdf@escapehex\endcsname\relax

\BIC@Expand
131   \def\BIC@Expand#1{%
132     \romannumeral0%
133     \BIC@@Expand#1!\@nil{}}%
134   }%

\BIC@@Expand
135   \def\BIC@@Expand#1#2\@nil#3{%
136     \expandafter\ifcat\noexpand#1\relax
137       \expandafter\@firstoftwo
138     \else
139       \expandafter\@secondoftwo
140     \fi
141     {%
142       \expandafter\BIC@@Expand#1#2\@nil{#3}%
143     }{%
144       \ifx#1!%
145         \expandafter\@firstoftwo
146       \else
147         \expandafter\@secondoftwo
148       \fi
149       { #3}{%
150         \BIC@@Expand#2\@nil{#3#1}%
151       }%
152     }%
153   }%

```

```

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

\@secondoftwo
157 \expandafter\ifx\csname @secondoftwo\endcsname\relax
158 \long\def\@secondoftwo#1#2{#2}%
159 \fi

160 \else

\BIC@Expand
161 \def\BIC@Expand#1{%
162 \romannumeral0\expandafter\expandafter\expandafter\BIC@Space
163 \pdf@unescapehex{%
164 \expandafter\expandafter\expandafter
165 \BIC@StripHexSpace\pdf@escapehex{#1}20\@nil
166 }%
167 }%

\BIC@StripHexSpace
168 \def\BIC@StripHexSpace#120#2\@nil{%
169 #1%
170 \ifx\#2\%
171 \else
172 \BIC@AfterFi{%
173 \BIC@StripHexSpace#2\@nil
174 }%
175 \BIC@Fi
176 }%

177 \fi

```

2.6 Normalize expanded number

```

\BIC@Normalize #1: result sign
#2: first token of number
178 \def\BIC@Normalize#1#2{%
179 \ifx#2-%
180 \ifx\#1\%
181 \BIC@AfterFiFi{%
182 \BIC@Normalize-%
183 }%
184 \else
185 \BIC@AfterFiFi{%
186 \BIC@Normalize{%
187 }%
188 \fi
189 \else
190 \ifx#2+%
191 \BIC@AfterFiFi{%
192 \BIC@Normalize{#1}%
193 }%
194 \else
195 \ifx#20%
196 \BIC@AfterFiFiFi{%
197 \BIC@NormalizeZero{#1}%
198 }%
199 \else
200 \BIC@AfterFiFiFi{%

```

```

201          \BIC@NormalizeDigits#1#2%
202      }%
203      \fi
204      \fi
205      \BIC@Fi
206  }

```

\BIC@NormalizeZero

```

207 \def\BIC@NormalizeZero#1#2{%
208   \ifx#2!%
209     \BIC@AfterFi{ 0}%
210   \else
211     \ifx#20%
212       \BIC@AfterFiFi{%
213         \BIC@NormalizeZero{#1}%
214       }%
215     \else
216       \BIC@AfterFiFi{%
217         \BIC@NormalizeDigits#1#2%
218       }%
219     \fi
220     \BIC@Fi
221  }

```

\BIC@NormalizeDigits

```

222 \def\BIC@NormalizeDigits#1!{ #1}

```

2.7 Num

\bigintcalcNum

```

223 \def\bigintcalcNum#1{%
224   \romannumeral0%
225   \expandafter\expandafter\expandafter\BIC@Normalize
226   \expandafter\expandafter\expandafter{%
227     \expandafter\expandafter\expandafter}%
228   \BIC@Expand{#1}!%
229  }

```

2.8 Inv, Abs, Sgn

\bigintcalcInv

```

230 \def\bigintcalcInv#1{%
231   \romannumeral0\expandafter\expandafter\expandafter\BIC@Space
232   \bigintcalcNum{-#1}%
233  }

```

\bigintcalcAbs

```

234 \def\bigintcalcAbs#1{%
235   \romannumeral0%
236   \expandafter\expandafter\expandafter\BIC@Abs
237   \bigintcalcNum{#1}%
238  }

```

\BIC@Abs

```

239 \def\BIC@Abs#1{%
240   \ifx#1-%
241     \expandafter\BIC@Space
242   \else
243     \expandafter\BIC@Space
244     \expandafter#1%

```

```

245 \fi
246 }

\bigintcalcSgn
247 \def\bigintcalcSgn#1{%
248 \number
249 \expandafter\expandafter\expandafter\BIC@Sgn
250 \bigintcalcNum{#1}! %
251 }

\BIC@Sgn
252 \def\BIC@Sgn#1#2!{%
253 \ifx#1-%
254 -1%
255 \else
256 \ifx#10%
257 0%
258 \else
259 1%
260 \fi
261 \fi
262 }

```

2.9 Cmp, Min, Max

```

\bigintcalcCmp
263 \def\bigintcalcCmp#1#2{%
264 \number
265 \expandafter\expandafter\expandafter\BIC@Cmp
266 \bigintcalcNum{#2}!{#1}%
267 }

\BIC@Cmp
268 \def\BIC@Cmp#1!#2{%
269 \expandafter\expandafter\expandafter\BIC@@Cmp
270 \bigintcalcNum{#2}!#1!%
271 }

\BIC@@Cmp
272 \def\BIC@@Cmp#1#2!#3#4!{%
273 \ifx#1-%
274 \ifx#3-%
275 \BIC@AfterFiFi{%
276 \BIC@@Cmp#4!#2!%
277 }%
278 \else
279 \BIC@AfterFiFi{%
280 -1 %
281 }%
282 \fi
283 \else
284 \ifx#3-%
285 \BIC@AfterFiFi{%
286 1 %
287 }%
288 \else
289 \BIC@AfterFiFi{%
290 \BIC@CmpLength#1#2!#3#4!#1#2!#3#4!%
291 }%
292 \fi
293 \BIC@Fi
294 }

```

```

\BIC@PosCmp
295 \def\BIC@PosCmp#1!#2!{%
296   \BIC@CmpLength#1!#2!#1!#2!%
297 }

\BIC@CmpLength
298 \def\BIC@CmpLength#1#2!#3#4!{%
299   \ifx\#2\%
300     \ifx\#4\%
301       \BIC@AfterFiFi\BIC@CmpDiff
302     \else
303       \BIC@AfterFiFi{%
304         \BIC@CmpResult{-1}%
305       }%
306     \fi
307   \else
308     \ifx\#4\%
309       \BIC@AfterFiFi{%
310         \BIC@CmpResult1%
311       }%
312     \else
313       \BIC@AfterFiFi{%
314         \BIC@CmpLength#2!#4!%
315       }%
316     \fi
317   \BIC@Fi
318 }

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

\BIC@CmpDiff
320 \def\BIC@CmpDiff#1#2!#3#4!{%
321   \ifnum#1<#3 %
322     \BIC@AfterFi{%
323       -1 %
324     }%
325   \else
326     \ifnum#1>#3 %
327       \BIC@AfterFiFi{%
328         1 %
329       }%
330     \else
331       \ifx\#2\%
332         \BIC@AfterFiFiFi{%
333           0 %
334         }%
335       \else
336         \BIC@AfterFiFiFi{%
337           \BIC@CmpDiff#2!#4!%
338         }%
339       \fi
340     \fi
341   \BIC@Fi
342 }

\bigintcalcMin
343 \def\bigintcalcMin#1{%
344   \romannumeral0%
345   \expandafter\expandafter\expandafter\BIC@MinMax
346   \bigintcalcNum{#1}!-!%
347 }

```

```

\bigintcalcMax
348 \def\bigintcalcMax#1{%
349   \romannumeral0%
350   \expandafter\expandafter\expandafter\BIC@MinMax
351   \bigintcalcNum{#1}!%!%
352 }

\BIC@MinMax #1:  $x$ 
#2: sign for comparison
#3:  $y$ 
353 \def\BIC@MinMax#1!#2!#3{%
354   \expandafter\expandafter\expandafter\BIC@MinMax
355   \bigintcalcNum{#3}!#1!#2!%
356 }

\BIC@@MinMax #1:  $y$ 
#2:  $x$ 
#3: sign for comparison
357 \def\BIC@@MinMax#1!#2!#3!{%
358   \ifnum\BIC@@Cmp#1!#2!=#31 %
359     \BIC@AfterFi{ #1}%
360   \else
361     \BIC@AfterFi{ #2}%
362   \BIC@Fi
363 }

```

2.10 Odd

```

\bigintcalcOdd
364 \def\bigintcalcOdd#1{%
365   \romannumeral0%
366   \expandafter\expandafter\expandafter\BIC@Odd
367   \bigintcalcAbs{#1}!%!%
368 }

\BigIntCalcOdd
369 \def\BigIntCalcOdd#1!{%
370   \romannumeral0%
371   \BIC@Odd#1!%
372 }

\BIC@Odd #1:  $x$ 
373 \def\BIC@Odd#1#2{%
374   \ifx#2!%
375     \ifodd#1 %
376       \BIC@AfterFiFi{ 1}%
377     \else
378       \BIC@AfterFiFi{ 0}%
379     \fi
380   \else
381     \expandafter\BIC@Odd\expandafter#2%
382   \BIC@Fi
383 }

```

2.11 Inc, Dec

```

\bigintcalcInc
384 \def\bigintcalcInc#1{%
385   \romannumeral0%
386   \expandafter\expandafter\expandafter\BIC@IncSwitch

```

```

387 \bigintcalcNum{#1}!%
388 }

\BIC@IncSwitch

389 \def\BIC@IncSwitch#1#2!{%
390 \ifcase\BIC@@Cmp#1#2!-1!%
391 \BIC@AfterFi{ 0}%
392 \or
393 \BIC@AfterFi{%
394 \BIC@Inc#1#2!{%}%
395 }%
396 \else
397 \BIC@AfterFi{%
398 \expandafter-\romannumeral0%
399 \BIC@Dec#2!{%}%
400 }%
401 \BIC@Fi
402 }

\bigintcalcDec

403 \def\bigintcalcDec#1{%
404 \romannumeral0%
405 \expandafter\expandafter\expandafter\BIC@DecSwitch
406 \bigintcalcNum{#1}!%
407 }

\BIC@DecSwitch

408 \def\BIC@DecSwitch#1#2!{%
409 \ifcase\BIC@Sgn#1#2! %
410 \BIC@AfterFi{ -1}%
411 \or
412 \BIC@AfterFi{%
413 \BIC@Dec#1#2!{%}%
414 }%
415 \else
416 \BIC@AfterFi{%
417 \expandafter-\romannumeral0%
418 \BIC@Inc#2!{%}%
419 }%
420 \BIC@Fi
421 }

\BigIntCalcInc

422 \def\BigIntCalcInc#1!{%
423 \romannumeral0\BIC@Inc#1!{%}%
424 }

\BigIntCalcDec

425 \def\BigIntCalcDec#1!{%
426 \romannumeral0\BIC@Dec#1!{%}%
427 }

\BIC@Inc

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

```

```

436     }%
437     \BIC@Fi
438 }

\BIC@@Inc

439 \def\BIC@@Inc#1#2#3!#4{%
440     \ifcase#1 %
441         \ifx\\#3\\%
442             \BIC@AfterFiFi{ #2#4}%
443         \else
444             \BIC@AfterFiFi{%
445                 \BIC@@Inc0#3!{#2#4}%
446             }%
447         \fi
448     \else
449         \ifnum#2<9 %
450             \BIC@AfterFiFi{%
451 &             \expandafter\BIC@@Inc\the\numexpr#2+1\relax
452 $             \expandafter\expandafter\expandafter\BIC@@Inc
453 $             \ifcase#2 \expandafter1%
454 $             \or\expandafter2%
455 $             \or\expandafter3%
456 $             \or\expandafter4%
457 $             \or\expandafter5%
458 $             \or\expandafter6%
459 $             \or\expandafter7%
460 $             \or\expandafter8%
461 $             \or\expandafter9%
462 $?             \else\BigIntCalcError:ThisCannotHappen%
463 $             \fi
464             0#3!{#4}%
465         }%
466     \else
467         \BIC@AfterFiFi{%
468             \BIC@@Inc01#3!{#4}%
469         }%
470     \fi
471     \BIC@Fi
472 }

\BIC@@@Inc

473 \def\BIC@@@Inc#1#2#3!#4{%
474     \ifx\\#3\\%
475         \ifnum#2=1 %
476             \BIC@AfterFiFi{ 1#1#4}%
477         \else
478             \BIC@AfterFiFi{ #1#4}%
479         \fi
480     \else
481         \BIC@AfterFi{%
482             \BIC@@Inc#2#3!{#1#4}%
483         }%
484     \BIC@Fi
485 }

\BIC@Dec

486 \def\BIC@Dec#1#2!#3{%
487     \ifx\\#2\\%
488         \BIC@AfterFi{%
489             \BIC@@Dec1#1#3!{ }%
490         }%
491     \else

```

```

492 \BIC@AfterFi{%
493 \BIC@Dec#2!{#1#3}%
494 }%
495 \BIC@Fi
496 }

```

\BIC@@Dec

```

497 \def\BIC@@Dec#1#2#3!#4{%
498 \ifcase#1 %
499 \ifx\#3\%
500 \BIC@AfterFiFi{ #2#4}%
501 \else
502 \BIC@AfterFiFi{%
503 \BIC@@Dec0#3!{#2#4}%
504 }%
505 \fi
506 \else
507 \ifnum#2>0 %
508 \BIC@AfterFiFi{%
509 & \expandafter\BIC@@Dec\the\numexpr#2-1\relax
510 $ \expandafter\expandafter\expandafter\BIC@@Dec
511 $ \ifcase#2
512 $? \BigIntCalcError:ThisCannotHappen%
513 $ \or\expandafter0%
514 $ \or\expandafter1%
515 $ \or\expandafter2%
516 $ \or\expandafter3%
517 $ \or\expandafter4%
518 $ \or\expandafter5%
519 $ \or\expandafter6%
520 $ \or\expandafter7%
521 $ \or\expandafter8%
522 $? \else\BigIntCalcError:ThisCannotHappen%
523 $ \fi
524 0#3!{#4}%
525 }%
526 \else
527 \BIC@AfterFiFi{%
528 \BIC@@Dec91#3!{#4}%
529 }%
530 \fi
531 \BIC@Fi
532 }

```

\BIC@@@Dec

```

533 \def\BIC@@@Dec#1#2#3!#4{%
534 \ifx\#3\%
535 \ifcase#1 %
536 \ifx\#4\%
537 \BIC@AfterFiFiFi{ 0}%
538 \else
539 \BIC@AfterFiFiFi{ #4}%
540 \fi
541 \else
542 \BIC@AfterFiFi{ #1#4}%
543 \fi
544 \else
545 \BIC@AfterFi{%
546 \BIC@@Dec#2#3!{#1#4}%
547 }%
548 \BIC@Fi
549 }

```

2.12 Add, Sub

\bigintcalcAdd

```
550 \def\bigintcalcAdd#1{%
551   \romannumeral0%
552   \expandafter\expandafter\expandafter\BIC@Add
553   \bigintcalcNum{#1}!%
554 }
```

\BIC@Add

```
555 \def\BIC@Add#1!#2{%
556   \expandafter\expandafter\expandafter
557   \BIC@AddSwitch\bigintcalcNum{#2}!#1!%
558 }
```

\bigintcalcSub

```
559 \def\bigintcalcSub#1#2{%
560   \romannumeral0%
561   \expandafter\expandafter\expandafter\BIC@Add
562   \bigintcalcNum{-#2}!{#1}%
563 }
```

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

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

```

590     \fi
591 \else % x >= 0
592     \ifx#3-% y < 0
593         \ifcase\BIC@PosCmp#1#2!#4!% x = -y
594             \BIC@AfterFiFiFi{ 0}%
595         \or % x > -y
596             \BIC@AfterFiFiFi{%
597                 \BIC@SubXY#1#2!#4!!!%
598             }%
599         \else % x <= -y
600             \expandafter-\romannumeral0%
601             \BIC@AfterFiFiFi{%
602                 \BIC@SubXY#4!#1#2!!!%
603             }%
604     \fi
605 \else % y >= 0
606     \ifnum\BIC@PosCmp#1#2!#3#4!=1 % x > y
607         \BIC@AfterFiFiFi{%
608             \BIC@AddXY#1#2!#3#4!!!%
609         }%
610     \else % x <= y
611         \BIC@AfterFiFiFi{%
612             \BIC@AddXY#3#4!#1#2!!!%
613         }%
614     \fi
615 \fi
616 \BIC@Fi
617 }

```

\BigIntCalcAdd

```

618 \def\BigIntCalcAdd#1#2!{%
619     \romannumeral0\BIC@AddXY#1!#2!!!%
620 }

```

\BigIntCalcSub

```

621 \def\BigIntCalcSub#1#2!{%
622     \romannumeral0\BIC@SubXY#1!#2!!!%
623 }

```

\BIC@AddXY

```

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

```

```

646     \else
647         \BIC@AfterFiFi{%
648             \BIC@AddXY#2!#4!#1#5!#3#6!%
649         }%
650     \fi
651 \BIC@Fi
652 }

\BIC@DoAdd #1: carry
#2: reverted result
#3#4: reverted  $x$ 
#5#6: reverted  $y$ 
653 \def\BIC@DoAdd#1#2!#3#4!#5#6!{%
654     \ifx\#4\%
655         \BIC@AfterFiFi{%
656             \expandafter\BIC@Space
657             \the\numexpr#1+#3+#5\relax#2%
658             \expandafter\expandafter\expandafter\BIC@AddResult
659             \BIC@AddDigit#1#3#5#2%
660         }%
661     \else
662         \BIC@AfterFiFi{%
663             \expandafter\expandafter\expandafter\BIC@DoAdd
664             \BIC@AddDigit#1#3#5#2!#4!#6!%
665         }%
666     \BIC@Fi
667 }

\BIC@AddResult
668 $ \def\BIC@AddResult#1{%
669 $     \ifx#10%
670 $         \expandafter\BIC@Space
671 $     \else
672 $         \expandafter\BIC@Space\expandafter#1%
673 $     \fi
674 $ }%

\BIC@AddDigit #1: carry
#2: digit of  $x$ 
#3: digit of  $y$ 
675 \def\BIC@AddDigit#1#2#3{%
676     \romannumeral0%
677 & \expandafter\BIC@@AddDigit\the\numexpr#1+#2+#3!%
678 $ \expandafter\BIC@@AddDigit\number%
679 $ \csname
680 $     BIC@AddCarry%
681 $     \ifcase#1 %
682 $         #2%
683 $     \else
684 $         \ifcase#2 1\or2\or3\or4\or5\or6\or7\or8\or9\or10\fi
685 $     \fi
686 $ \endcsname#3!%
687 }

\BIC@@AddDigit
688 \def\BIC@@AddDigit#1!{%
689     \ifnum#1<10 %
690         \BIC@AfterFiFi{ 0#1}%
691     \else
692         \BIC@AfterFiFi{ #1}%
693     \BIC@Fi
694 }

```

\BIC@AddCarry0

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

\BIC@AddCarry10

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

\BIC@AddCarry[1-9]

```

697 $ \def\BIC@Temp#1#2{%
698 $   \expandafter\def\csname BIC@AddCarry#1\endcsname##1{%
699 $     \ifcase##1 #1\or
700 $       #2%
701 $?   \else\BigIntCalcError:ThisCannotHappen%
702 $     \fi
703 $   }%
704 $ }%
705 $ \BIC@Temp 0{1\or2\or3\or4\or5\or6\or7\or8\or9}%
706 $ \BIC@Temp 1{2\or3\or4\or5\or6\or7\or8\or9\or10}%
707 $ \BIC@Temp 2{3\or4\or5\or6\or7\or8\or9\or10\or11}%
708 $ \BIC@Temp 3{4\or5\or6\or7\or8\or9\or10\or11\or12}%
709 $ \BIC@Temp 4{5\or6\or7\or8\or9\or10\or11\or12\or13}%
710 $ \BIC@Temp 5{6\or7\or8\or9\or10\or11\or12\or13\or14}%
711 $ \BIC@Temp 6{7\or8\or9\or10\or11\or12\or13\or14\or15}%
712 $ \BIC@Temp 7{8\or9\or10\or11\or12\or13\or14\or15\or16}%
713 $ \BIC@Temp 8{9\or10\or11\or12\or13\or14\or15\or16\or17}%
714 $ \BIC@Temp 9{10\or11\or12\or13\or14\or15\or16\or17\or18}%

```

\BIC@SubXY Preconditions:

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

```

715 \def\BIC@SubXY#1#2!#3#4!#5!#6!{%
716   \ifx\#2\%
717     \ifx\#3\%
718       \BIC@AfterFiFi{%
719         \BIC@DoSub0!#1#5!#60!%
720       }%
721     \else
722       \BIC@AfterFiFi{%
723         \BIC@DoSub0!#1#5!#3#6!%
724       }%
725     \fi
726   \else
727     \ifx\#4\%
728       \ifx\#3\%
729         \BIC@AfterFiFiFi{%
730           \BIC@SubXY#2!{ }!#1#5!#60!%
731         }%
732       \else
733         \BIC@AfterFiFiFi{%
734           \BIC@SubXY#2!{ }!#1#5!#3#6!%
735         }%
736       \fi
737     \else
738       \BIC@AfterFiFi{%
739         \BIC@SubXY#2!#4!#1#5!#3#6!%
740       }%
741     \fi
742   \BIC@Fi
743 }

```

```

\BIC@DoSub #1: carry
#2: reverted result
#3#4: reverted x
#5#6: reverted y
744 \def\BIC@DoSub#1#2!#3#4!#5#6!{%
745   \ifx\#4\%
746     \BIC@AfterFi{%
747       \expandafter\expandafter\expandafter\BIC@SubResult
748       \BIC@SubDigit#1#3#5#2%
749     }%
750   \else
751     \BIC@AfterFi{%
752       \expandafter\expandafter\expandafter\BIC@DoSub
753       \BIC@SubDigit#1#3#5#2!#4!#6!%
754     }%
755   \BIC@Fi
756 }

\BIC@SubResult
757 \def\BIC@SubResult#1{%
758   \ifx#10%
759     \expandafter\BIC@SubResult
760   \else
761     \expandafter\BIC@Space\expandafter#1%
762   \fi
763 }

\BIC@SubDigit #1: carry
#2: digit of  $x$ 
#3: digit of  $y$ 
764 \def\BIC@SubDigit#1#2#3{%
765   \romannumeral0%
766   \expandafter\BIC@SubDigit\the\numexpr#2-#3-#1!%
767 $ \expandafter\BIC@@AddDigit\number
768 $   \csname
769 $     BIC@SubCarry%
770 $     \ifcase#1 %
771 $       #3%
772 $     \else
773 $       \ifcase#3 1\or2\or3\or4\or5\or6\or7\or8\or9\or10\fi
774 $     \fi
775 $     \endcsname#2!%
776 }

\BIC@@SubDigit
777 & \def\BIC@@SubDigit#1!{%
778 &   \ifnum#1<0 %
779 &     \BIC@AfterFi{%
780 &       \expandafter\BIC@Space
781 &       \expandafter1\the\numexpr#1+10\relax
782 &     }%
783 &   \else
784 &     \BIC@AfterFi{ 0#1}%
785 &   \BIC@Fi
786 & }%

\BIC@SubCarry0
787 $ \expandafter\def\csname BIC@SubCarry0\endcsname#1{#1}%

\BIC@SubCarry10
788 $ \expandafter\def\csname BIC@SubCarry10\endcsname#1{#1}%

```

\BIC@SubCarry[1-9]

```
789 $ \def\BIC@Temp#1#2{%
790 $   \expandafter\def\csname BIC@SubCarry#1\endcsname##1{%
791 $     \ifcase##1 #2%
792 $?   \else\BigIntCalcError:ThisCannotHappen%
793 $     \fi
794 $   }%
795 $ }%
796 $ \BIC@Temp 1{19\or0\or1\or2\or3\or4\or5\or6\or7\or8}%
797 $ \BIC@Temp 2{18\or19\or0\or1\or2\or3\or4\or5\or6\or7}%
798 $ \BIC@Temp 3{17\or18\or19\or0\or1\or2\or3\or4\or5\or6}%
799 $ \BIC@Temp 4{16\or17\or18\or19\or0\or1\or2\or3\or4\or5}%
800 $ \BIC@Temp 5{15\or16\or17\or18\or19\or0\or1\or2\or3\or4}%
801 $ \BIC@Temp 6{14\or15\or16\or17\or18\or19\or0\or1\or2\or3}%
802 $ \BIC@Temp 7{13\or14\or15\or16\or17\or18\or19\or0\or1\or2}%
803 $ \BIC@Temp 8{12\or13\or14\or15\or16\or17\or18\or19\or0\or1}%
804 $ \BIC@Temp 9{11\or12\or13\or14\or15\or16\or17\or18\or19\or0}%

```

2.13 Shl, Shr

\bigintcalcShl

```
805 \def\bigintcalcShl#1{%
806   \romannumeral0%
807   \expandafter\expandafter\expandafter\BIC@Shl
808   \bigintcalcNum{#1}!%
809 }

```

\BIC@Shl

```
810 \def\BIC@Shl#1#2!{%
811   \ifx#1-%
812     \BIC@AfterFi{%
813       \expandafter-\romannumeral0%
814 &     \BIC@@Shl#2!!%
815 $     \BIC@AddXY#2!#2!!!%
816   }%
817   \else
818     \BIC@AfterFi{%
819 &     \BIC@@Shl#1#2!!%
820 $     \BIC@AddXY#1#2!#1#2!!!%
821   }%
822   \BIC@Fi
823 }

```

\BigIntCalcShl

```
824 \def\BigIntCalcShl#1!{%
825   \romannumeral0%
826 & \BIC@@Shl#1!!%
827 $ \BIC@AddXY#1!#1!!!%
828 }

```

\BIC@@Shl

```
829 & \def\BIC@@Shl#1#2!{%
830 &   \ifx\#2\%
831 &     \BIC@AfterFi{%
832 &       \BIC@@@Shl0!#1%
833 &     }%
834 &   \else
835 &     \BIC@AfterFi{%
836 &       \BIC@@Shl#2!#1%
837 &     }%
838 &   \BIC@Fi
839 & }%

```

```

\BIC@@@Shl #1: carry
#2: result
#3#4: reverted number
840 & \def\BIC@@@Shl#1#2!#3#4!{%
841 & \ifx\#4\%
842 & \BIC@AfterFi{%
843 & \expandafter\BIC@Space
844 & \the\numexpr#3*2+#1\relax#2%
845 & }%
846 & \else
847 & \BIC@AfterFi{%
848 & \expandafter\BIC@@@Shl\the\numexpr#3*2+#1!#2!#4!%
849 & }%
850 & \BIC@Fi
851 & }%

\BIC@@@Shl
852 & \def\BIC@@@Shl#1!{%
853 & \ifnum#1<10 %
854 & \BIC@AfterFi{%
855 & \BIC@@Shl0#1%
856 & }%
857 & \else
858 & \BIC@AfterFi{%
859 & \BIC@@Shl#1%
860 & }%
861 & \BIC@Fi
862 & }%

\bigintcalcShr
863 \def\bigintcalcShr#1{%
864 \romannumeral0%
865 \expandafter\expandafter\expandafter\BIC@Shr
866 \bigintcalcNum{#1}!%
867 }

\BIC@Shr
868 \def\BIC@Shr#1#2!{%
869 \ifx#1-%
870 \expandafter-\romannumeral0%
871 \BIC@AfterFi{%
872 \BIC@@Shr#2!%
873 }%
874 \else
875 \BIC@AfterFi{%
876 \BIC@@Shr#1#2!%
877 }%
878 \BIC@Fi
879 }

\BigIntCalcShr
880 \def\BigIntCalcShr#1!{%
881 \romannumeral0%
882 \BIC@@Shr#1!%
883 }

\BIC@@Shr
884 \def\BIC@@Shr#1#2!{%
885 \ifcase#1 %
886 \BIC@AfterFi{ 0}%
887 \or

```

```

888     \ifx\|#2\|%
889     \BIC@AfterFiFi{ 0}%
890     \else
891     \BIC@AfterFiFi{%
892     \BIC@@@Shr#1#2!!%
893     }%
894     \fi
895     \else
896     \BIC@AfterFiFi{%
897     \BIC@@@Shr0#1#2!!%
898     }%
899     \BIC@Fi
900 }

\BIC@@@Shr #1: carry
#2#3: number
#4: result
901 \def\BIC@@@Shr#1#2#3!#4!{%
902     \ifx\|#3\|%
903     \ifodd#1#2 %
904     \BIC@AfterFiFi{%
905     & \expandafter\BIC@ShrResult\the\numexpr(#1#2-1)/2\relax
906     $ \expandafter\expandafter\expandafter\BIC@ShrResult
907     $ \csname BIC@ShrDigit#1#2\endcsname
908     #4!%
909     }%
910     \else
911     \BIC@AfterFiFi{%
912     & \expandafter\BIC@ShrResult\the\numexpr#1#2/2\relax
913     $ \expandafter\expandafter\expandafter\BIC@ShrResult
914     $ \csname BIC@ShrDigit#1#2\endcsname
915     #4!%
916     }%
917     \fi
918     \else
919     \ifodd#1#2 %
920     \BIC@AfterFiFi{%
921     & \expandafter\BIC@@@Shr\the\numexpr(#1#2-1)/2\relax1%
922     $ \expandafter\expandafter\expandafter\BIC@@@Shr
923     $ \csname BIC@ShrDigit#1#2\endcsname
924     #3!#4!%
925     }%
926     \else
927     \BIC@AfterFiFi{%
928     & \expandafter\BIC@@@Shr\the\numexpr#1#2/2\relax0%
929     $ \expandafter\expandafter\expandafter\BIC@@@Shr
930     $ \csname BIC@ShrDigit#1#2\endcsname
931     #3!#4!%
932     }%
933     \fi
934     \BIC@Fi
935 }

\BIC@ShrResult
936 & \def\BIC@ShrResult#1#2!{ #2#1}%
937 $ \def\BIC@ShrResult#1#2#3!{ #3#1}%

\BIC@@@Shr #1: new digit
#2: carry
#3: remaining number
#4: result
938 \def\BIC@@@Shr#1#2#3!#4!{%

```

```

939 \BIC@@@Shr#2#3!#4#1!%
940 }

```

\BIC@ShrDigit[00-19]

```

941 $ \def\BIC@Temp#1#2#3#4{%
942 $ \expandafter\def\csname BIC@ShrDigit#1#2\endcsname{#3#4}%
943 $ }%
944 $ \BIC@Temp 0000%
945 $ \BIC@Temp 0101%
946 $ \BIC@Temp 0210%
947 $ \BIC@Temp 0311%
948 $ \BIC@Temp 0420%
949 $ \BIC@Temp 0521%
950 $ \BIC@Temp 0630%
951 $ \BIC@Temp 0731%
952 $ \BIC@Temp 0840%
953 $ \BIC@Temp 0941%
954 $ \BIC@Temp 1050%
955 $ \BIC@Temp 1151%
956 $ \BIC@Temp 1260%
957 $ \BIC@Temp 1361%
958 $ \BIC@Temp 1470%
959 $ \BIC@Temp 1571%
960 $ \BIC@Temp 1680%
961 $ \BIC@Temp 1781%
962 $ \BIC@Temp 1890%
963 $ \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

```

964 \def\BIC@@Tim#1#2!{%
965 \ifx\#2\%
966 \BIC@AfterFi{%
967 \BIC@ProcessTim0!#1%
968 }%
969 \else
970 \BIC@AfterFi{%
971 \BIC@@Tim#2!#1%
972 }%
973 \BIC@Fi
974 }

```

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

```

975 \def\BIC@ProcessTim#1#2!#3#4!#5{%
976 \ifx\#4\%
977 \BIC@AfterFi{%
978 \expandafter\BIC@Space
979 & \the\numexpr#3*#5+#1\relax
980 $ \romannumeral0\BIC@TimDigit#3#5#1%
981 #2%
982 }%
983 \else

```

```

984     \BIC@AfterFi{%
985     \expandafter\BIC@@ProcessTim
986 &     \the\numexpr#3*#5+#1%
987 $     \romannumeral0\BIC@TimDigit#3#5#1%
988     !#2!#4!#5%
989     }%
990     \BIC@Fi
991 }

\BIC@@ProcessTim #1#2: carry?, new digit
#3: new number
#4: old number
#5: digit
992 \def\BIC@@ProcessTim#1#2!{%
993     \ifx\#2\%
994     \BIC@AfterFi{%
995     \BIC@ProcessTim0#1%
996     }%
997     \else
998     \BIC@AfterFi{%
999     \BIC@ProcessTim#1#2%
1000     }%
1001     \BIC@Fi
1002 }

\BIC@TimDigit #1: digit 0-9
#2: digit 3-9
#3: carry 0-9
1003 $ \def\BIC@TimDigit#1#2#3{%
1004 $     \ifcase#1 % 0
1005 $         \BIC@AfterFi{ #3}%
1006 $     \or % 1
1007 $         \BIC@AfterFi{%
1008 $             \expandafter\BIC@Space
1009 $             \number\csname BIC@AddCarry#2\endcsname#3 %
1010 $         }%
1011 $     \else
1012 $         \ifcase#3 %
1013 $             \BIC@AfterFiFi{%
1014 $                 \expandafter\BIC@Space
1015 $                 \number\csname BIC@MulDigit#2\endcsname#1 %
1016 $             }%
1017 $         \else
1018 $             \BIC@AfterFiFi{%
1019 $                 \expandafter\BIC@Space
1020 $                 \romannumeral0%
1021 $                 \expandafter\BIC@AddXY
1022 $                 \number\csname BIC@MulDigit#2\endcsname#1!%
1023 $                 #3!!!%
1024 $             }%
1025 $         \fi
1026 $     \BIC@Fi
1027 $ }%

\BIC@MulDigit[3-9]
1028 $ \def\BIC@Temp#1#2{%
1029 $     \expandafter\def\csname BIC@MulDigit#1\endcsname##1{%
1030 $         \ifcase##1 0%
1031 $         \or ##1%
1032 $         \or ##2%
1033 $     }? \else\BigIntCalcError:ThisCannotHappen%
1034 $     \fi

```

```

1035 $ }%
1036 $ }%
1037 $ \BIC@Temp 3{6\or9\or12\or15\or18\or21\or24\or27}%
1038 $ \BIC@Temp 4{8\or12\or16\or20\or24\or28\or32\or36}%
1039 $ \BIC@Temp 5{10\or15\or20\or25\or30\or35\or40\or45}%
1040 $ \BIC@Temp 6{12\or18\or24\or30\or36\or42\or48\or54}%
1041 $ \BIC@Temp 7{14\or21\or28\or35\or42\or49\or56\or63}%
1042 $ \BIC@Temp 8{16\or24\or32\or40\or48\or56\or64\or72}%
1043 $ \BIC@Temp 9{18\or27\or36\or45\or54\or63\or72\or81}%

```

2.15 Mul

\bigintcalcMul

```

1044 \def\bigintcalcMul#1#2{%
1045   \romannumeral0%
1046   \expandafter\expandafter\expandafter\BIC@Mul
1047   \bigintcalcNum{#1}!{#2}%
1048 }

```

\BIC@Mul

```

1049 \def\BIC@Mul#1!#2{%
1050   \expandafter\expandafter\expandafter\BIC@MulSwitch
1051   \bigintcalcNum{#2}!#1!%
1052 }

```

\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		$\text{Mul}(-y, -x)$

```

1053 \def\BIC@MulSwitch#1#2!#3#4!{%
1054   \ifcase\BIC@Sgn#1#2! % x = 0
1055     \BIC@AfterFi{ 0}%
1056   \or % x > 0
1057     \ifcase\BIC@Sgn#3#4! % y = 0
1058       \BIC@AfterFiFi{ 0}%
1059     \or % y > 0
1060       \ifnum\BIC@PosCmp#1#2!#3#4!=1 % x > y
1061         \BIC@AfterFiFiFi{%
1062           \BIC@ProcessMul0!#1#2!#3#4!%
1063         }%
1064       \else % x <= y
1065         \BIC@AfterFiFiFi{%
1066           \BIC@ProcessMul0!#3#4!#1#2!%
1067         }%
1068     \fi
1069   \else % y < 0
1070     \expandafter-\romannumeral0%
1071     \ifnum\BIC@PosCmp#1#2!#4!=1 % x > -y
1072       \BIC@AfterFiFiFi{%
1073         \BIC@ProcessMul0!#1#2!#4!%
1074       }%

```

```

1075     \else % x <= -y
1076         \BIC@AfterFiFiFi{%
1077             \BIC@ProcessMul0!#4!#1#2!%
1078         }%
1079     \fi
1080 \fi
1081 \else % x < 0
1082     \ifcase\BIC@Sgn#3#4! % y = 0
1083         \BIC@AfterFiFiFi{ 0}%
1084     \or % y > 0
1085         \expandafter-\romannumeral0%
1086         \ifnum\BIC@PosCmp#2!#3#4!=1 % -x > y
1087             \BIC@AfterFiFiFi{%
1088                 \BIC@ProcessMul0!#2!#3#4!%
1089             }%
1090         \else % -x <= y
1091             \BIC@AfterFiFiFi{%
1092                 \BIC@ProcessMul0!#3#4!#2!%
1093             }%
1094         \fi
1095     \else % y < 0
1096         \ifnum\BIC@PosCmp#2!#4!=1 % -x > -y
1097             \BIC@AfterFiFiFi{%
1098                 \BIC@ProcessMul0!#2!#4!%
1099             }%
1100         \else % -x <= -y
1101             \BIC@AfterFiFiFi{%
1102                 \BIC@ProcessMul0!#4!#2!%
1103             }%
1104         \fi
1105     \fi
1106 \BIC@Fi
1107 }

```

\BigIntCalcMul

```

1108 \def\BigIntCalcMul#1!#2!{%
1109     \romannumeral0%
1110     \BIC@ProcessMul0!#1!#2!%
1111 }

```

\BIC@ProcessMul

```

#1: result
#2: number  $x$ 
#3#4: number  $y$ 
1112 \def\BIC@ProcessMul#1!#2!#3#4!{%
1113     \ifx\#4\%
1114         \BIC@AfterFiFi{%
1115             \expandafter\expandafter\expandafter\BIC@Space
1116             \bigintcalcAdd{\BIC@Tim#2!#3}{#10}%
1117         }%
1118     \else
1119         \BIC@AfterFiFi{%
1120             \expandafter\expandafter\expandafter\BIC@ProcessMul
1121             \bigintcalcAdd{\BIC@Tim#2!#3}{#10}!#2!#4!%
1122         }%
1123     \BIC@Fi
1124 }

```

2.16 Sqr

\bigintcalcSqr

```

1125 \def\bigintcalcSqr#1{%
1126     \romannumeral0%

```

```

1127 \expandafter\expandafter\expandafter\BIC@Sqr
1128 \bigintcalcNum{#1}!%
1129 }

```

\BIC@Sqr

```

1130 \def\BIC@Sqr#1{%
1131   \ifx#1-%
1132     \expandafter\BIC@@Sqr
1133   \else
1134     \expandafter\BIC@@Sqr\expandafter#1%
1135   \fi
1136 }

```

\BIC@@Sqr

```

1137 \def\BIC@@Sqr#1!{%
1138   \BIC@ProcessMul0!#1!#1!%
1139 }

```

2.17 Fac

\bigintcalcFac

```

1140 \def\bigintcalcFac#1{%
1141   \romannumeral0%
1142   \expandafter\expandafter\expandafter\BIC@Fac
1143   \bigintcalcNum{#1}!%
1144 }

```

\BIC@Fac

```

1145 \def\BIC@Fac#1#2!{%
1146   \ifx#1-%
1147     \BIC@AfterFi{ 0\BigIntCalcError:FacNegative}%
1148   \else
1149     \ifnum\BIC@PosCmp#1#2!13!<0 %
1150       \ifcase#1#2 %
1151         \BIC@AfterFiFiFi{ 1}% 0!
1152       \or\BIC@AfterFiFiFi{ 1}% 1!
1153       \or\BIC@AfterFiFiFi{ 2}% 2!
1154       \or\BIC@AfterFiFiFi{ 6}% 3!
1155       \or\BIC@AfterFiFiFi{ 24}% 4!
1156       \or\BIC@AfterFiFiFi{ 120}% 5!
1157       \or\BIC@AfterFiFiFi{ 720}% 6!
1158       \or\BIC@AfterFiFiFi{ 5040}% 7!
1159       \or\BIC@AfterFiFiFi{ 40320}% 8!
1160       \or\BIC@AfterFiFiFi{ 362880}% 9!
1161       \or\BIC@AfterFiFiFi{ 3628800}% 10!
1162       \or\BIC@AfterFiFiFi{ 39916800}% 11!
1163       \or\BIC@AfterFiFiFi{ 479001600}% 12!
1164     ? \else\BigIntCalcError:ThisCannotHappen%
1165     \fi
1166   \else
1167     \BIC@AfterFiFi{%
1168       \BIC@ProcessFac#1#2!479001600!%
1169     }%
1170   \fi
1171   \BIC@Fi
1172 }

```

\BIC@ProcessFac #1: n

#2: result

```

1173 \def\BIC@ProcessFac#1!#2!{%
1174   \ifnum\BIC@PosCmp#1!12!=0 %

```

```

1175     \BIC@AfterFi{ #2}%
1176 \else
1177     \BIC@AfterFi{%
1178         \expandafter\BIC@@ProcessFac
1179         \romannumeral0\BIC@ProcessMul0!#2!#1!%
1180         !#1!%
1181     }%
1182 \BIC@Fi
1183 }

```

```

\BIC@@ProcessFac #1: result
#2:  $n$ 
1184 \def\BIC@@ProcessFac#1!#2!{%
1185     \expandafter\BIC@ProcessFac
1186     \romannumeral0\BIC@Dec#2!{%}%
1187     !#1!%
1188 }

```

2.18 Pow

```

\bigintcalcPow #1: basis
#2: power
1189 \def\bigintcalcPow#1{%
1190     \romannumeral0%
1191     \expandafter\expandafter\expandafter\BIC@Pow
1192     \bigintcalcNum{#1}!%
1193 }

\BIC@Pow #1: basis
#2: power
1194 \def\BIC@Pow#1!#2{%
1195     \expandafter\expandafter\expandafter\BIC@PowSwitch
1196     \bigintcalcNum{#2}!#1!%
1197 }

\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 ($x > 1$)	$\text{Pow}(x, y)$	

```

1198 \def\BIC@PowSwitch#1#2!#3#4!{%
1199     \ifcase\ifx\\#2\\%
1200         \ifx#100 %  $y = 0$ 
1201         \else\ifx#111 %  $y = 1$ 

```

```

1202         \else\ifx#122 % y = 2
1203         \else4 % y > 2
1204         \fi\fi\fi
1205     \else
1206         \ifx#1-3 % y < 0
1207         \else4 % y > 2
1208         \fi
1209     \fi
1210     \BIC@AfterFi{ 1}% y = 0
1211 \or % y = 1
1212     \BIC@AfterFi{ #3#4}%
1213 \or % y = 2
1214     \ifx#3-% x < 0
1215         \BIC@AfterFiFi{%
1216             \BIC@ProcessMul0!#4!#4!%
1217         }%
1218     \else % x >= 0
1219         \BIC@AfterFiFi{%
1220             \BIC@ProcessMul0!#3#4!#3#4!%
1221         }%
1222     \fi
1223 \or % y < 0
1224     \ifcase\ifx\#4\%
1225         \ifx#300 % x = 0
1226         \else\ifx#311 % x = 1
1227         \else3 % x > 1
1228         \fi\fi
1229     \else
1230         \ifcase\BIC@MinusOne#3#4! %
1231             3 % |x| > 1
1232         \or
1233             2 % x = -1
1234 ?         \else\BigIntCalcError:ThisCannotHappen%
1235         \fi
1236     \fi
1237     \BIC@AfterFiFi{ 0\BigIntCalcError:DivisionByZero}% x = 0
1238 \or % x = 1
1239     \BIC@AfterFiFi{ 1}% x = 1
1240 \or % x = -1
1241     \ifcase\BIC@ModTwo#2! % even(y)
1242         \BIC@AfterFiFiFi{ 1}%
1243     \or % odd(y)
1244         \BIC@AfterFiFiFi{ -1}%
1245 ?     \else\BigIntCalcError:ThisCannotHappen%
1246     \fi
1247 \or % |x| > 1
1248     \BIC@AfterFiFi{ 0}%
1249 ?     \else\BigIntCalcError:ThisCannotHappen%
1250     \fi
1251 \or % y > 2
1252     \ifcase\ifx\#4\%
1253         \ifx#300 % x = 0
1254         \else\ifx#311 % x = 1
1255         \else4 % x > 1
1256         \fi\fi
1257     \else
1258         \ifx#3-%
1259             \ifcase\BIC@MinusOne#3#4! %
1260                 3 % x < -1
1261             \else
1262                 2 % x = -1
1263             \fi

```

```

1264         \else
1265         4 % x > 1
1266         \fi
1267     \fi
1268     \BIC@AfterFiFi{ 0}% x = 0
1269 \or % x = 1
1270     \BIC@AfterFiFi{ 1}% x = 1
1271 \or % x = -1
1272     \ifcase\BIC@ModTwo#1#2! % even(y)
1273         \BIC@AfterFiFiFi{ 1}%
1274     \or % odd(y)
1275         \BIC@AfterFiFiFi{ -1}%
1276 ?     \else\BigIntCalcError:ThisCannotHappen%
1277     \fi
1278 \or % x < -1
1279     \ifcase\BIC@ModTwo#1#2! % even(y)
1280         \BIC@AfterFiFiFi{%
1281         \BIC@PowRec#4!#1#2!1!%
1282     }%
1283 \or % odd(y)
1284     \expandafter-\romannumeral0%
1285     \BIC@AfterFiFiFi{%
1286     \BIC@PowRec#4!#1#2!1!%
1287 }%
1288 ?     \else\BigIntCalcError:ThisCannotHappen%
1289     \fi
1290 \or % x > 1
1291     \BIC@AfterFiFi{%
1292     \BIC@PowRec#3#4!#1#2!1!%
1293 }%
1294 ?     \else\BigIntCalcError:ThisCannotHappen%
1295     \fi
1296 ? \else\BigIntCalcError:ThisCannotHappen%
1297 \BIC@Fi
1298 }

```

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.

```

1299 \def\BIC@ModTwo#1#2!{%
1300     \ifx\#2\%
1301         \ifodd#1 %
1302             \BIC@AfterFiFi1%
1303         \else
1304             \BIC@AfterFiFi0%
1305         \fi
1306     \else
1307         \BIC@AfterFi{%
1308         \BIC@ModTwo#2!%
1309     }%
1310 \BIC@Fi
1311 }

```

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

```

1312 \def\BIC@MinusOne#1#2!{%
1313     \ifx#1-%
1314         \BIC@@MinusOne#2!%
1315     \else
1316         0%
1317     \fi

```

```

1318 }

\BIC@@MinusOne

1319 \def\BIC@@MinusOne#1#2!{%
1320   \ifx#11%
1321     \ifx\|#2\|%
1322       1%
1323     \else
1324       0%
1325     \fi
1326   \else
1327     0%
1328   \fi
1329 }

```

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)
1330 \def\BIC@PowRec#1!#2#3!#4!{%
1331   \ifcase\ifx#21\ifx\|#3\|0 \else1 \fi\else1 \fi % y = 1
1332     \ifnum\BIC@PosCmp#1!#4!=1 % x > r
1333       \BIC@AfterFiFi{%
1334         \BIC@ProcessMul0!#1!#4!%
1335       }%
1336     \else
1337       \BIC@AfterFiFi{%
1338         \BIC@ProcessMul0!#4!#1!%
1339       }%
1340     \fi
1341   \or
1342     \ifcase\BIC@ModTwo#2#3! % even(y)
1343       \BIC@AfterFiFi{%
1344         \expandafter\BIC@@PowRec\romannumeral0%
1345         \BIC@@Shr#2#3!%
1346         !#1!#4!%
1347       }%
1348     \or % odd(y)
1349       \ifnum\BIC@PosCmp#1!#4!=1 % x > r
1350         \BIC@AfterFiFiFi{%
1351           \expandafter\BIC@@@PowRec\romannumeral0%
1352           \BIC@ProcessMul0!#1!#4!%
1353           !#1!#2#3!%
1354         }%
1355       \else
1356         \BIC@AfterFiFiFi{%
1357           \expandafter\BIC@@@PowRec\romannumeral0%

```

```

1358         \BIC@ProcessMul0!#1!#4!%
1359         !#1!#2#3!%
1360     }%
1361     \fi
1362 ?   \else\BigIntCalcError:ThisCannotHappen%
1363     \fi
1364 ?   \else\BigIntCalcError:ThisCannotHappen%
1365     \BIC@Fi
1366 }

```

```

\BIC@@PowRec #1:  $y/2$ 
#2:  $x$ 
#3: new  $r$  ( $r$  or  $r * x$ )
1367 \def\BIC@@PowRec#1!#2!#3!{%
1368   \expandafter\BIC@PowRec\romannumeral0%
1369   \BIC@ProcessMul0!#2!#2!%
1370   !#1!#3!%
1371 }

```

```

\BIC@@@PowRec #1:  $r * x$  #2:  $x$  #3:  $y$ 
1372 \def\BIC@@@PowRec#1!#2!#3!{%
1373   \expandafter\BIC@@PowRec\romannumeral0%
1374   \BIC@@Shr#3!%
1375   !#2!#1!%
1376 }

```

2.19 Div

```

\bigintcalcDiv #1:  $x$ 
#2:  $y$  (divisor)
1377 \def\bigintcalcDiv#1{%
1378   \romannumeral0%
1379   \expandafter\expandafter\expandafter\BIC@Div
1380   \bigintcalcNum{#1}!%
1381 }

```

```

\BIC@Div #1:  $x$ 
#2:  $y$ 
1382 \def\BIC@Div#1!#2{%
1383   \expandafter\expandafter\expandafter\BIC@DivSwitchSign
1384   \bigintcalcNum{#2}!#1!%
1385 }

```

```

\BigIntCalcDiv
1386 \def\BigIntCalcDiv#1!#2!{%
1387   \romannumeral0%
1388   \BIC@DivSwitchSign#2!#1!%
1389 }

```

\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:  $x$ 
1390 \def\BIC@DivSwitchSign#1#2!#3#4!{%
1391   \ifcase\BIC@Sgn#1#2! %  $y = 0$ 
1392     \BIC@AfterFi{ 0\BigIntCalcError:DivisionByZero}%
1393   \or %  $y > 0$ 
1394     \ifcase\BIC@Sgn#3#4! %  $x = 0$ 
1395       \BIC@AfterFiFi{ 0}%
1396     \or %  $x > 0$ 
1397       \BIC@AfterFiFi{%
1398         \BIC@DivSwitch{ }#3#4!#1#2!%
1399       }%
1400     \else %  $x < 0$ 
1401       \BIC@AfterFiFi{%
1402         \BIC@DivSwitch-#4!#1#2!%
1403       }%
1404     \fi
1405   \else %  $y < 0$ 
1406     \ifcase\BIC@Sgn#3#4! %  $x = 0$ 
1407       \BIC@AfterFiFi{ 0}%
1408     \or %  $x > 0$ 
1409       \BIC@AfterFiFi{%
1410         \BIC@DivSwitch-#3#4!#2!%
1411       }%
1412     \else %  $x < 0$ 
1413       \BIC@AfterFiFi{%
1414         \BIC@DivSwitch{ }#4!#2!%
1415       }%
1416     \fi
1417   \BIC@Fi
1418 }

```

\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:  $y$  ( $y \neq 0$ )
1419 \def\BIC@DivSwitch#1#2!#3#4!{%
1420   \ifcase\BIC@PosCmp#3#4!#2!%  $y = x$ 
1421     \BIC@AfterFi{ #1}%
1422   \or %  $y > x$ 
1423     \BIC@AfterFi{ 0}%
1424   \else %  $y < x$ 
1425     \ifx\#1\%
1426       \else
1427         \expandafter-\romannumeral0%
1428       \fi
1429     \ifcase\ifx\#4\%
1430       \ifx#310 %  $y = 1$ 
1431       \else\ifx#321 %  $y = 2$ 
1432       \else\ifx#342 %  $y = 4$ 
1433       \else3 %  $y > 2$ 
1434       \fi\fi\fi
1435     \else
1436       3 %  $y > 2$ 

```

```

1437         \fi
1438         \BIC@AfterFiFi{ #2}% y = 1
1439     \or % y = 2
1440         \BIC@AfterFiFi{%
1441             \BIC@Shr#2!%
1442         }%
1443     \or % y = 4
1444         \BIC@AfterFiFi{%
1445             \expandafter\BIC@Shr\romannumeral0%
1446             \BIC@Shr#2!%
1447         }%
1448     \or % y > 2
1449         \BIC@AfterFiFi{%
1450             \BIC@DivStartX#2!#3#4!!!%
1451         }%
1452 ? \else\BigIntCalcError:ThisCannotHappen%
1453     \fi
1454 \BIC@Fi
1455 }

\BIC@ProcessDiv #1#2:  $x$ 
#3#4:  $y$ 
#5: collect first digits of  $x$ 
#6: corresponding digits of  $y$ 
1456 \def\BIC@DivStartX#1#2!#3#4!#5!#6!{%
1457     \ifx\#4\%
1458         \BIC@AfterFi{%
1459             \BIC@DivStartYii#6#3#4!{#5#1}#2=!%
1460         }%
1461     \else
1462         \BIC@AfterFi{%
1463             \BIC@DivStartX#2!#4!#5#1!#6#3!%
1464         }%
1465     \BIC@Fi
1466 }

\BIC@DivStartYii #1:  $y$ 
#2:  $x$ , =
1467 \def\BIC@DivStartYii#1!{%
1468     \expandafter\BIC@DivStartYiv\romannumeral0%
1469     \BIC@Shl#1!%
1470     !#1!%
1471 }

\BIC@DivStartYiv #1:  $2y$ 
#2:  $y$ 
#3:  $x$ , =
1472 \def\BIC@DivStartYiv#1!{%
1473     \expandafter\BIC@DivStartYvi\romannumeral0%
1474     \BIC@Shl#1!%
1475     !#1!%
1476 }

\BIC@DivStartYvi #1:  $4y$ 
#2:  $2y$ 
#3:  $y$ 
#4:  $x$ , =
1477 \def\BIC@DivStartYvi#1!#2!{%
1478     \expandafter\BIC@DivStartYviii\romannumeral0%
1479     \BIC@AddXY#1!#2!!!%
1480     !#1!#2!%
1481 }

```

```

\BIC@DivStartYviii #1: 6y
                   #2: 4y
                   #3: 2y
                   #4: y
                   #5: x, =
1482 \def\BIC@DivStartYviii#1!#2!{%
1483   \expandafter\BIC@DivStart\romannumeral0%
1484   \BIC@Shl#2!%
1485   !#1!#2!%
1486 }

\BIC@DivStart #1: 8y
              #2: 6y
              #3: 4y
              #4: 2y
              #5: y
              #6: x, =
1487 \def\BIC@DivStart#1!#2!#3!#4!#5!#6!{%
1488   \BIC@ProcessDiv#6!!#5!#4!#3!#2!#1!=%
1489 }

\BIC@ProcessDiv #1#2#3: x, =
                #4: result
                #5: y
                #6: 2y
                #7: 4y
                #8: 6y
                #9: 8y
1490 \def\BIC@ProcessDiv#1#2#3!#4!#5!{%
1491   \ifcase\BIC@PosCmp#5!#1!% y = #1
1492     \ifx#2=%
1493       \BIC@AfterFiFi{\BIC@DivCleanup{#41}}%
1494     \else
1495       \BIC@AfterFiFi{%
1496         \BIC@ProcessDiv#2#3!#41!#5!%
1497       }%
1498     \fi
1499   \or % y > #1
1500     \ifx#2=%
1501       \BIC@AfterFiFi{\BIC@DivCleanup{#40}}%
1502     \else
1503       \ifx\#4\%
1504         \BIC@AfterFiFiFi{%
1505           \BIC@ProcessDiv{#1#2}#3!#5!%
1506         }%
1507       \else
1508         \BIC@AfterFiFiFi{%
1509           \BIC@ProcessDiv{#1#2}#3!#40!#5!%
1510         }%
1511       \fi
1512     \fi
1513   \else % y < #1
1514     \BIC@AfterFi{%
1515       \BIC@ProcessDiv{#1}#2#3!#4!#5!%
1516     }%
1517   \BIC@Fi
1518 }

\BIC@DivCleanup #1: result
                #2: garbage
1519 \def\BIC@DivCleanup#1#2={ #1}%

```

\BIC@@ProcessDiv

```

1520 \def\BIC@@ProcessDiv#1#2#3!#4!#5!#6!#7!{%
1521   \ifcase\BIC@PosCmp#7!#1!% 4y = #1
1522     \ifx#2=%
1523       \BIC@AfterFiFi{\BIC@DivCleanup{#44}}}%
1524     \else
1525       \BIC@AfterFiFi{%
1526         \BIC@ProcessDiv#2#3!#44!#5!#6!#7!%
1527       }%
1528     \fi
1529   \or % 4y > #1
1530     \ifcase\BIC@PosCmp#6!#1!% 2y = #1
1531       \ifx#2=%
1532         \BIC@AfterFiFiFi{\BIC@DivCleanup{#42}}}%
1533       \else
1534         \BIC@AfterFiFiFi{%
1535           \BIC@ProcessDiv#2#3!#42!#5!#6!#7!%
1536         }%
1537       \fi
1538     \or % 2y > #1
1539       \ifx#2=%
1540         \BIC@AfterFiFiFi{\BIC@DivCleanup{#41}}}%
1541       \else
1542         \BIC@AfterFiFiFi{%
1543           \BIC@DivSub#1!#5!#2#3!#41!#5!#6!#7!%
1544         }%
1545       \fi
1546     \else % 2y < #1
1547       \BIC@AfterFiFi{%
1548         \expandafter\BIC@ProcessDivII\romannumeral0%
1549         \BIC@SubXY#1!#6!!!%
1550         !#2#3!#4!#5!23%
1551         #6!#7!%
1552       }%
1553     \fi
1554   \else % 4y < #1
1555     \BIC@AfterFiFi{%
1556       \BIC@@@ProcessDiv{#1}#2#3!#4!#5!#6!#7!%
1557     }%
1558   \BIC@Fi
1559 }

```

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

```

1560 \def\BIC@DivSub#1!#2!#3{%
1561   \expandafter\BIC@ProcessDiv\expandafter{%
1562     \romannumeral0%
1563     \BIC@SubXY#1!#2!!!%
1564     #3%
1565   }%
1566 }

```

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

```

1567 \def\BIC@ProcessDivII#1!#2#3!#4!#5!#6#7{%
1568   \ifcase\BIC@PosCmp#5!#1!% y = #1
1569     \ifx#2=%
1570       \BIC@AfterFiFi{\BIC@DivCleanup{#4#7}}}%
1571     \else

```

```

1572     \BIC@AfterFiFi{%
1573     \BIC@ProcessDiv#2#3!#4#7!#5!%
1574     }%
1575     \fi
1576 \or % y > #1
1577     \ifx#2=%
1578     \BIC@AfterFiFi{\BIC@DivCleanup{#4#6}}%
1579     \else
1580     \BIC@AfterFiFi{%
1581     \BIC@ProcessDiv{#1#2}#3!#4#6!#5!%
1582     }%
1583     \fi
1584 \else % y < #1
1585     \ifx#2=%
1586     \BIC@AfterFiFi{\BIC@DivCleanup{#4#7}}%
1587     \else
1588     \BIC@AfterFiFi{%
1589     \BIC@DivSub#1!#5!#2#3!#4#7!#5!%
1590     }%
1591     \fi
1592 \BIC@Fi
1593 }

```

```

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

1594 \def\BIC@@@ProcessDiv#1#2#3!#4!#5!#6!#7!#8!#9!{%
1595     \ifcase\BIC@PosCmp#8!#1!% 6y = #1
1596     \ifx#2=%
1597     \BIC@AfterFiFi{\BIC@DivCleanup{#46}}%
1598     \else
1599     \BIC@AfterFiFi{%
1600     \BIC@ProcessDiv#2#3!#46!#5!#6!#7!#8!#9!%
1601     }%
1602     \fi
1603 \or % 6y > #1
1604     \BIC@AfterFi{%
1605     \expandafter\BIC@ProcessDivII\romannumeral0%
1606     \BIC@SubXY#1!#7!!!%
1607     !#2#3!#4!#5!45%
1608     #6!#7!#8!#9!%
1609     }%
1610 \else % 6y < #1
1611     \ifcase\BIC@PosCmp#9!#1!% 8y = #1
1612     \ifx#2=%
1613     \BIC@AfterFiFiFi{\BIC@DivCleanup{#48}}%
1614     \else
1615     \BIC@AfterFiFiFi{%
1616     \BIC@ProcessDiv#2#3!#48!#5!#6!#7!#8!#9!%
1617     }%
1618     \fi
1619 \or % 8y > #1
1620     \BIC@AfterFiFi{%
1621     \expandafter\BIC@ProcessDivII\romannumeral0%
1622     \BIC@SubXY#1!#8!!!%
1623     !#2#3!#4!#5!67%
1624     #6!#7!#8!#9!%
1625     }%

```

```

1626 \else % 8y < #1
1627 \BIC@AfterFiFi{%
1628 \expandafter\BIC@ProcessDivII\romannumeral0%
1629 \BIC@SubXY#1!#9!!!%
1630 !#2#3!#4!#5!89%
1631 #6!#7!#8!#9!%
1632 }%
1633 \fi
1634 \BIC@Fi
1635 }

```

2.20 Mod

```

\bigintcalcMod #1: x
#2: y
1636 \def\bigintcalcMod#1{%
1637 \romannumeral0%
1638 \expandafter\expandafter\expandafter\BIC@Mod
1639 \bigintcalcNum{#1}!%
1640 }

\BIC@Mod #1: x
#2: y
1641 \def\BIC@Mod#1!#2{%
1642 \expandafter\expandafter\expandafter\BIC@ModSwitchSign
1643 \bigintcalcNum{#2}!#1!%
1644 }

\BigIntCalcMod
1645 \def\BigIntCalcMod#1!#2!{%
1646 \romannumeral0%
1647 \BIC@ModSwitchSign#2!#1!%
1648 }

```

\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
1649 \def\BIC@ModSwitchSign#1#2!#3#4!{%
1650 \ifcase\ifx\#2\%
1651 \ifx#100 % y = 0
1652 \else1 % y > 0
1653 \fi
1654 \else
1655 \ifx#1-2 % y < 0
1656 \else1 % y > 0
1657 \fi
1658 \fi
1659 \BIC@AfterFi{ 0\BigIntCalcError:DivisionByZero}%
1660 \or % y > 0
1661 \ifcase\ifx\#4\%
1662 \BIC@AfterFiFi{ 0}%
1663 \else
1664 \BIC@AfterFiFi{%
1665 \BIC@ModSwitch{ }#3#4!#1#2!%
1666 }%

```

```

1667 \fi
1668 \else % y < 0
1669 \ifcase\ifx\#4\%
1670 \ifx#300 % x = 0
1671 \else1 % x > 0
1672 \fi
1673 \else
1674 \ifx#3-2 % x < 0
1675 \else1 % x > 0
1676 \fi
1677 \fi
1678 \BIC@AfterFiFi{ 0}%
1679 \or % x > 0
1680 \BIC@AfterFiFi{%
1681 \BIC@ModSwitch--#3#4!#2!%
1682 }%
1683 \else % x < 0
1684 \BIC@AfterFiFi{%
1685 \BIC@ModSwitch-#4!#2!%
1686 }%
1687 \fi
1688 \BIC@Fi
1689 }

```

\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

```

1690 \def\BIC@ModSwitch#1#2#3!#4#5!{%
1691 \ifcase\ifx\#5\%
1692 \ifx#410 % y = 1
1693 \else\ifx#421 % y = 2
1694 \else2 % y > 2
1695 \fi\fi
1696 \else2 % y > 2
1697 \fi
1698 \BIC@AfterFi{ 0}% y = 1
1699 \or % y = 2
1700 \ifcase\BIC@ModTwo#2#3! % even(x)
1701 \BIC@AfterFiFi{ 0}%
1702 \or % odd(x)
1703 \BIC@AfterFiFi{ #11}%
1704 ? \else\BigIntCalcError:ThisCannotHappen%
1705 \fi
1706 \or % y > 2
1707 \ifx\#1\%
1708 \else
1709 \expandafter\BIC@Space\romannumeral0%
1710 \expandafter\BIC@ModMinus\romannumeral0%
1711 \fi
1712 \ifx#2-% x < 0
1713 \BIC@AfterFiFi{%
1714 \expandafter\expandafter\expandafter\BIC@ModX
1715 \bigintcalcSub{#2#3}{%
1716 \bigintcalcMul{#4#5}{\bigintcalcDiv{#2#3}{#4#5}}}%

```

```

1717         }!#4#5!%
1718     }%
1719     \else % x > 0
1720         \BIC@AfterFiFi{%
1721             \expandafter\expandafter\expandafter\BIC@Space
1722             \bigintcalcSub{#2#3}{%
1723                 \bigintcalcMul{#4#5}{\bigintcalcDiv{#2#3}{#4#5}}%
1724             }%
1725         }%
1726     \fi
1727 ? \else\BigIntCalcError:ThisCannotHappen%
1728     \BIC@Fi
1729 }

\BIC@ModMinus

1730 \def\BIC@ModMinus#1{%
1731     \ifx#10%
1732         \BIC@AfterFi{ 0}%
1733     \else
1734         \BIC@AfterFi{ -#1}%
1735     \BIC@Fi
1736 }

\BIC@ModX #1#2: z
#3: x
1737 \def\BIC@ModX#1#2!#3!{%
1738     \ifx#1-% z < 0
1739         \BIC@AfterFi{%
1740             \expandafter\BIC@Space\romannumeral0%
1741             \BIC@SubXY#3!#2!!!%
1742         }%
1743     \else % z >= 0
1744         \BIC@AfterFi{ #1#2}%
1745     \BIC@Fi
1746 }

1747 \BIC@AtEnd
1748 \</package>

```

3 Test

3.1 Catcode checks for loading

```

1749 \<test1>

1750 \catcode'\{=1 %
1751 \catcode'\}=2 %
1752 \catcode'\#=6 %
1753 \catcode'\@=11 %
1754 \expandafter\ifx\csname count@\endcsname\relax
1755     \countdef\count@=255 %
1756 \fi
1757 \expandafter\ifx\csname @gobble\endcsname\relax
1758     \long\def\@gobble#1{}%
1759 \fi
1760 \expandafter\ifx\csname @firstofone\endcsname\relax
1761     \long\def\@firstofone#1{#1}%
1762 \fi
1763 \expandafter\ifx\csname loop\endcsname\relax
1764     \expandafter\@firstofone
1765 \else
1766     \expandafter\@gobble

```

```

1767 \fi
1768 {%
1769   \def\loop#1\repeat{%
1770     \def\body{#1}%
1771     \iterate
1772   }%
1773   \def\iterate{%
1774     \body
1775     \let\next\iterate
1776   \else
1777     \let\next\relax
1778   \fi
1779   \next
1780 }%
1781 \let\repeat=\fi
1782 }%
1783 \def\RestoreCatcodes{}
1784 \count@=0 %
1785 \loop
1786   \edef\RestoreCatcodes{%
1787     \RestoreCatcodes
1788     \catcode\the\count@=\the\catcode\count@\relax
1789   }%
1790 \ifnum\count@<255 %
1791   \advance\count@ 1 %
1792 \repeat
1793
1794 \def\RangeCatcodeInvalid#1#2{%
1795   \count@=#1\relax
1796   \loop
1797     \catcode\count@=15 %
1798   \ifnum\count@<#2\relax
1799     \advance\count@ 1 %
1800   \repeat
1801 }
1802 \expandafter\ifx\csname LoadCommand\endcsname\relax
1803   \def\LoadCommand{\input bigintcalc.sty\relax}%
1804 \fi
1805 \def\Test{%
1806   \RangeCatcodeInvalid{0}{47}%
1807   \RangeCatcodeInvalid{58}{64}%
1808   \RangeCatcodeInvalid{91}{96}%
1809   \RangeCatcodeInvalid{123}{255}%
1810   \catcode'\@=12 %
1811   \catcode'\@=0 %
1812   \catcode'\{=1 %
1813   \catcode'\}=2 %
1814   \catcode'\#=6 %
1815   \catcode'\[=12 %
1816   \catcode'\]=12 %
1817   \catcode'\%=14 %
1818   \catcode'\ =10 %
1819   \catcode13=5 %
1820   \LoadCommand
1821   \RestoreCatcodes
1822 }
1823 \Test
1824 \csname @@end\endcsname
1825 \end
1826 </test1>

```

3.2 Macro tests

3.2.1 Preamble with test macro definitions

```
1827 <*test2>
1828 \NeedsTeXFormat{LaTeX2e}
1829 \nofiles
1830 \documentclass{article}
1831 <noetex> \let \SavedNumexpr \numexpr
1832 <noetex> \let \numexpr \UNDEFINED
1833 \makeatletter
1834 \chardef \BIC@TestMode=1 %
1835 \makeatother
1836 \usepackage{bigintcalc}[2007/11/11]
1837 <noetex> \let \numexpr \SavedNumexpr
1838 \usepackage{qstest}
1839 \IncludeTests{*}
1840 \LogTests{log}{*}{*}
1841 \newcommand*{\TestSpaceAtEnd}[1]{%
1842 <noetex> \let \SavedNumexpr \numexpr
1843 <noetex> \let \numexpr \UNDEFINED
1844 \edef\resultA{#1}%
1845 \edef\resultB{#1 }%
1846 <noetex> \let \numexpr \SavedNumexpr
1847 \Expect*{\resultA\space}*{\resultB}%
1848 }
1849 \newcommand*{\TestResult}[2]{%
1850 <noetex> \let \SavedNumexpr \numexpr
1851 <noetex> \let \numexpr \UNDEFINED
1852 \edef\result{#1}%
1853 <noetex> \let \numexpr \SavedNumexpr
1854 \Expect*{\result}{#2}%
1855 }
1856 \newcommand*{\TestResultTwoExpansions}[2]{%
1857 <*noetex>
1858 \begingroup
1859 \let \numexpr \UNDEFINED
1860 \expandafter\expandafter\expandafter
1861 \endgroup
1862 </noetex>
1863 \expandafter\expandafter\expandafter\Expect
1864 \expandafter\expandafter\expandafter{#1}{#2}%
1865 }
1866 \newcount\TestCount
1867 <etex> \newcommand*{\TestArg}[1]{\numexpr#1\relax}
1868 <noetex> \newcommand*{\TestArg}[1]{#1}
1869 \newcommand*{\TestTeXDivide}[2]{%
1870 \TestCount=\TestArg{#1}\relax
1871 \divide\TestCount by \TestArg{#2}\relax
1872 \Expect*{\bigintcalcDiv{#1}{#2}}*{\the\TestCount}%
1873 }
1874 \newcommand*{\Test}[2]{%
1875 \TestResult{#1}{#2}%
1876 \TestResultTwoExpansions{#1}{#2}%
1877 \TestSpaceAtEnd{#1}%
1878 }
1879 \newcommand*{\TestExch}[2]{\Test{#2}{#1}}
1880 \newcommand*{\TestInv}[2]{%
1881 \Test{\bigintcalcInv{#1}}{#2}%
1882 }
1883 \newcommand*{\TestAbs}[2]{%
1884 \Test{\bigintcalcAbs{#1}}{#2}%
1885 }
```

```

1886 \newcommand*{\TestSgn}[2]{%
1887   \Test{\bigintcalcSgn{#1}}{#2}%
1888 }
1889 \newcommand*{\TestMin}[3]{%
1890   \Test{\bigintcalcMin{#1}{#2}}{#3}%
1891 }
1892 \newcommand*{\TestMax}[3]{%
1893   \Test{\bigintcalcMax{#1}{#2}}{#3}%
1894 }
1895 \newcommand*{\TestCmp}[3]{%
1896   \Test{\bigintcalcCmp{#1}{#2}}{#3}%
1897 }
1898 \newcommand*{\TestOdd}[2]{%
1899   \Test{\bigintcalcOdd{#1}}{#2}%
1900   \edef\x{%
1901     \noexpand\Test{%
1902       \noexpand\BigIntCalcOdd
1903       \bigintcalcAbs{#1}!%
1904     }{#2}%
1905   }%
1906   \x
1907 }
1908 \newcommand*{\TestInc}[2]{%
1909   \Test{\bigintcalcInc{#1}}{#2}%
1910   \ifnum\bigintcalcSgn{#1}>-1 %
1911     \edef\x{%
1912       \noexpand\Test{%
1913         \noexpand\BigIntCalcInc\bigintcalcNum{#1}!%
1914       }{#2}%
1915     }%
1916     \x
1917   \fi
1918 }
1919 \newcommand*{\TestDec}[2]{%
1920   \Test{\bigintcalcDec{#1}}{#2}%
1921   \ifnum\bigintcalcSgn{#1}>0 %
1922     \edef\x{%
1923       \noexpand\Test{%
1924         \noexpand\BigIntCalcDec\bigintcalcNum{#1}!%
1925       }{#2}%
1926     }%
1927     \x
1928   \fi
1929 }
1930 \newcommand*{\TestAdd}[3]{%
1931   \Test{\bigintcalcAdd{#1}{#2}}{#3}%
1932   \ifnum\bigintcalcSgn{#1}>0 %
1933     \ifnum\bigintcalcSgn{#2}> 0 %
1934       \ifnum\bigintcalcCmp{#1}{#2}>0 %
1935         \edef\x{%
1936           \noexpand\Test{%
1937             \noexpand\BigIntCalcAdd
1938             \bigintcalcNum{#1}!\bigintcalcNum{#2}!%
1939           }{#3}%
1940         }%
1941         \x
1942       \else
1943         \edef\x{%
1944           \noexpand\Test{%
1945             \noexpand\BigIntCalcAdd
1946             \bigintcalcNum{#2}!\bigintcalcNum{#1}!%
1947           }{#3}%

```

```

1948     }%
1949     \x
1950     \fi
1951     \fi
1952     \fi
1953 }
1954 \newcommand*{\TestSub}[3]{%
1955   \Test{\bigintcalcSub{#1}{#2}}{#3}%
1956   \ifnum\bigintcalcSgn{#1}>0 %
1957     \ifnum\bigintcalcSgn{#2}> 0 %
1958       \ifnum\bigintcalcCmp{#1}{#2}>0 %
1959         \edef\x{%
1960           \noexpand\Test{%
1961             \noexpand\BigIntCalcSub
1962             \bigintcalcNum{#1}!\bigintcalcNum{#2}!%
1963           }{#3}%
1964         }%
1965         \x
1966       \fi
1967     \fi
1968   \fi
1969 }
1970 \newcommand*{\TestShl}[2]{%
1971   \Test{\bigintcalcShl{#1}}{#2}%
1972   \edef\x{%
1973     \noexpand\Test{%
1974       \noexpand\BigIntCalcShl\bigintcalcAbs{#1}!%
1975     }{\bigintcalcAbs{#2}}}%
1976   }%
1977   \x
1978 }
1979 \newcommand*{\TestShr}[2]{%
1980   \Test{\bigintcalcShr{#1}}{#2}%
1981   \edef\x{%
1982     \noexpand\Test{%
1983       \noexpand\BigIntCalcShr\bigintcalcAbs{#1}!%
1984     }{\bigintcalcAbs{#2}}}%
1985   }%
1986   \x
1987 }
1988 \newcommand*{\TestMul}[3]{%
1989   \Test{\bigintcalcMul{#1}{#2}}{#3}%
1990   \edef\x{%
1991     \noexpand\Test{%
1992       \noexpand\BigIntCalcMul
1993       \bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
1994     }{\bigintcalcAbs{#3}}}%
1995   }%
1996   \x
1997 }
1998 \newcommand*{\TestSqr}[2]{%
1999   \Test{\bigintcalcSqr{#1}}{#2}%
2000 }
2001 \newcommand*{\TestFac}[2]{%
2002   \expandafter\TestExch\expandafter{%
2003     \the\numexpr#2%
2004   }{\bigintcalcFac{#1}}%
2005 }
2006 \newcommand*{\TestFacBig}[2]{%
2007   \Test{\bigintcalcFac{#1}}{#2}%
2008 }
2009 \newcommand*{\TestPow}[3]{%

```

```

2010 \Test{\bigintcalcPow{#1}{#2}}{#3}%
2011 }
2012 \newcommand*{\TestDiv}[3]{%
2013 \Test{\bigintcalcDiv{#1}{#2}}{#3}%
2014 \TestTeXDivide{#1}{#2}%
2015 }
2016 \newcommand*{\TestDivBig}[3]{%
2017 \Test{\bigintcalcDiv{#1}{#2}}{#3}%
2018 \edef\x{%
2019 \noexpand\Test{%
2020 \noexpand\BigIntCalcDiv\bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
2021 }\bigintcalcAbs{#3}}%
2022 }%
2023 }
2024 \newcommand*{\TestMod}[3]{%
2025 \Test{\bigintcalcMod{#1}{#2}}{#3}%
2026 \ifcase\ifcase\bigintcalcSgn{#1} 0%
2027 \or
2028 \ifcase\bigintcalcSgn{#2} 1%
2029 \or 0%
2030 \else 1%
2031 \fi
2032 \else
2033 \ifcase\bigintcalcSgn{#2} 1%
2034 \or 1%
2035 \else 0%
2036 \fi
2037 \fi\relax
2038 \edef\x{%
2039 \noexpand\Test{%
2040 \noexpand\BigIntCalcMod
2041 \bigintcalcAbs{#1}!\bigintcalcAbs{#2}!%
2042 }\bigintcalcAbs{#3}}%
2043 }%
2044 \x
2045 \fi
2046 }

```

3.2.2 Time

```

2047 \beginingroup\expandafter\expandafter\expandafter\endgroup
2048 \expandafter\ifx\csname pdfresettimer\endcsname\relax
2049 \else
2050 \makeatletter
2051 \newcount\SummaryTime
2052 \newcount\TestTime
2053 \SummaryTime=\z@
2054 \newcommand*{\PrintTime}[2]{%
2055 \typeout{%
2056 [Time #1: \strip@pt\dimexpr\number#2sp\relax\space s]%
2057 }%
2058 }%
2059 \newcommand*{\StartTime}[1]{%
2060 \renewcommand*{\TimeDescription}{#1}%
2061 \pdfresettimer
2062 }%
2063 \newcommand*{\TimeDescription}{}%
2064 \newcommand*{\StopTime}{%
2065 \TestTime=\pdfelapsedtime
2066 \global\advance\SummaryTime\TestTime
2067 \PrintTime\TimeDescription\TestTime
2068 }%
2069 \let\saved@qstest\qstest
2070 \let\saved@endqstest\endqstest

```

```

2071 \def\qstest#1#2{%
2072   \saved@qstest{#1}{#2}%
2073   \StartTime{#1}%
2074 }%
2075 \def\endqstest{%
2076   \StopTime
2077   \saved@endqstest
2078 }%
2079 \AtEndDocument{%
2080   \PrintTime{summary}\SummaryTime
2081 }%
2082 \makeatother
2083 \fi

```

3.2.3 Test sets

```

2084 \makeatletter
2085
2086 \begin{qstest}{inv}{inv}%
2087   \TestInv{0}{0}%
2088   \TestInv{1}{-1}%
2089   \TestInv{-1}{1}%
2090   \TestInv{10}{-10}%
2091   \TestInv{-10}{10}%
2092   \TestInv{2147483647}{-2147483647}%
2093   \TestInv{-2147483647}{2147483647}%
2094   \TestInv{12345678901234567890}{-12345678901234567890}%
2095   \TestInv{-12345678901234567890}{12345678901234567890}%
2096   \TestInv{ 0 }{0}%
2097   \TestInv{ 1 }{-1}%
2098   \TestInv{--1}{-1}%
2099   \TestInv{\number\z@}{0}%
2100   \TestInv{\ifx\relax\relax1\fi}{-1}%
2101   \TestInv{\ifx\relax\relax-\fi\ifx234\else1\fi}{1}%
2102 \end{qstest}
2103
2104 \begin{qstest}{abs}{abs}%
2105   \TestAbs{0}{0}%
2106   \TestAbs{1}{1}%
2107   \TestAbs{-1}{1}%
2108   \TestAbs{10}{10}%
2109   \TestAbs{-10}{10}%
2110   \TestAbs{2147483647}{2147483647}%
2111   \TestAbs{-2147483647}{2147483647}%
2112   \TestAbs{12345678901234567890}{12345678901234567890}%
2113   \TestAbs{-12345678901234567890}{12345678901234567890}%
2114   \TestAbs{ 0 }{0}%
2115   \TestAbs{ 1 }{1}%
2116   \TestAbs{--1}{1}%
2117   \TestAbs{-++1}{1}%
2118   \TestAbs{00000000000}{0}%
2119   \TestAbs{00000001000}{1000}%
2120   \TestAbs{\ifx\relax\relax 0\else 1\fi}{0}%
2121 \end{qstest}
2122
2123 \begin{qstest}{sign}{sign}%
2124   \TestSgn{0}{0}%
2125   \TestSgn{1}{1}%
2126   \TestSgn{-1}{-1}%
2127   \TestSgn{10}{1}%
2128   \TestSgn{-10}{-1}%
2129   \TestSgn{2147483647}{1}%
2130   \TestSgn{-2147483647}{-1}%
2131   \TestSgn{12345678901234567890}{1}%

```

```

2132 \TestSgn{-12345678901234567890}{-1}%
2133 \TestSgn{ 0 }{0}%
2134 \TestSgn{ 2 }{1}%
2135 \TestSgn{ -2 }{-1}%
2136 \TestSgn{--2}{1}%
2137 \TestSgn{\number\z@}{0}%
2138 \TestSgn{\number\@ne}{1}%
2139 \TestSgn{\number\m@ne}{-1}%
2140 \TestSgn{%
2141   -++\number\z@\number\z@
2142   \iftrue1\fi\iftrue2\fi\iftrue3\fi
2143 }{1}%
2144 \end{qstest}
2145
2146 \begin{qstest}{min}{min}%
2147 \TestMin{0}{1}{0}%
2148 \TestMin{1}{0}{0}%
2149 \TestMin{-10}{-20}{-20}%
2150 \TestMin{ 1 }{ 2 }{1}%
2151 \TestMin{ 2 }{ 1 }{1}%
2152 \TestMin{1}{1}{1}%
2153 \TestMin{\number\z@}{\number\@ne}{0}%
2154 \TestMin{\number\@ne}{\number\m@ne}{-1}%
2155 \end{qstest}
2156
2157 \begin{qstest}{max}{max}%
2158 \TestMax{0}{1}{1}%
2159 \TestMax{1}{0}{1}%
2160 \TestMax{-10}{-20}{-10}%
2161 \TestMax{ 1 }{ 2 }{2}%
2162 \TestMax{ 2 }{ 1 }{2}%
2163 \TestMax{1}{1}{1}%
2164 \TestMax{\number\z@}{\number\@ne}{1}%
2165 \TestMax{\number\@ne}{\number\m@ne}{1}%
2166 \end{qstest}
2167
2168 \begin{qstest}{cmp}{cmp}%
2169 \TestCmp{0}{0}{0}%
2170 \TestCmp{-21}{17}{-1}%
2171 \TestCmp{3}{4}{-1}%
2172 \TestCmp{-10}{-10}{0}%
2173 \TestCmp{-10}{-11}{1}%
2174 \TestCmp{100}{5}{1}%
2175 \TestCmp{9}{10}{-1}%
2176 \TestCmp{10}{9}{1}%
2177 \TestCmp{ 3 }{ 3 }{0}%
2178 \TestCmp{-9}{-10}{1}%
2179 \TestCmp{-10}{-9}{-1}%
2180 \TestCmp{-3}{-3}{0}%
2181 \TestCmp{0}{-2}{1}%
2182 \TestCmp{0}{2}{-1}%
2183 \TestCmp{2}{0}{1}%
2184 \TestCmp{-2}{0}{-1}%
2185 \TestCmp{12}{11}{1}%
2186 \TestCmp{11}{12}{-1}%
2187 \TestCmp{2147483647}{-2147483647}{1}%
2188 \TestCmp{-2147483647}{2147483647}{-1}%
2189 \TestCmp{2147483647}{2147483647}{0}%
2190 \TestCmp{\number\z@}{\number\@ne}{-1}%
2191 \TestCmp{\number\@ne}{\number\m@ne}{1}%
2192 \TestCmp{ 4 }{ 5 }{-1}%
2193 \TestCmp{ -3 }{ -7 }{1}%

```

```

2194 \end{qstest}
2195
2196 \begin{qstest}{odd}{odd}
2197 \tracingmacros=1
2198 \TestOdd{0}{0}%
2199 \TestOdd{1}{1}%
2200 \TestOdd{2}{0}%
2201 \TestOdd{3}{1}%
2202 \TestOdd{14}{0}%
2203 \TestOdd{15}{1}%
2204 \TestOdd{12345678901234567896}{0}%
2205 \TestOdd{12345678901234567897}{1}%
2206 \end{qstest}
2207
2208 \begin{qstest}{inc}{inc}%
2209 \TestInc{0}{1}%
2210 \TestInc{1}{2}%
2211 \TestInc{-1}{0}%
2212 \TestInc{10}{11}%
2213 \TestInc{-10}{-9}%
2214 \TestInc{ 3 }{4}%
2215 \TestInc{999}{1000}%
2216 \TestInc{-1000}{-999}%
2217 \TestInc{129}{130}%
2218 \TestInc{2147483646}{2147483647}%
2219 \TestInc{-2147483647}{-2147483646}%
2220 \TestInc{12345678901234567890}{12345678901234567891}%
2221 \TestInc{9999999999999999999}{10000000000000000000}%
2222 \TestInc{-12345678901234567891}{-12345678901234567890}%
2223 \TestInc{-10000000000000000000}{-9999999999999999999}%
2224 \end{qstest}
2225
2226 \begin{qstest}{dec}{dec}%
2227 \TestDec{0}{-1}%
2228 \TestDec{1}{0}%
2229 \TestDec{-1}{-2}%
2230 \TestDec{10}{9}%
2231 \TestDec{-10}{-11}%
2232 \TestDec{1000}{999}%
2233 \TestDec{-999}{-1000}%
2234 \TestDec{130}{129}%
2235 \TestDec{2147483647}{2147483646}%
2236 \TestDec{-2147483646}{-2147483647}%
2237 \TestDec{12345678901234567891}{12345678901234567890}%
2238 \TestDec{10000000000000000000}{9999999999999999999}%
2239 \TestDec{-12345678901234567890}{-12345678901234567891}%
2240 \TestDec{-9999999999999999999}{-10000000000000000000}%
2241 \end{qstest}
2242
2243 \begin{qstest}{add}{add}%
2244 \TestAdd{0}{0}{0}%
2245 \TestAdd{1}{0}{1}%
2246 \TestAdd{0}{1}{1}%
2247 \TestAdd{1}{2}{3}%
2248 \TestAdd{-1}{-1}{-2}%
2249 \TestAdd{2147483646}{1}{2147483647}%
2250 \TestAdd{-2147483647}{2147483647}{0}%
2251 \TestAdd{20}{-5}{15}%
2252 \TestAdd{-4}{-1}{-5}%
2253 \TestAdd{-1}{-4}{-5}%
2254 \TestAdd{-4}{1}{-3}%
2255 \TestAdd{-1}{4}{3}%

```

```

2256 \TestAdd{4}{-1}{3}%
2257 \TestAdd{1}{-4}{-3}%
2258 \TestAdd{-4}{-1}{-5}%
2259 \TestAdd{-1}{-4}{-5}%
2260 \TestAdd{ -4 }{ -1 }{-5}%
2261 \TestAdd{ -1 }{ -4 }{-5}%
2262 \TestAdd{ -4 }{ 1 }{-3}%
2263 \TestAdd{ -1 }{ 4 }{3}%
2264 \TestAdd{ 4 }{ -1 }{3}%
2265 \TestAdd{ 1 }{ -4 }{-3}%
2266 \TestAdd{ -4 }{ -1 }{-5}%
2267 \TestAdd{ -1 }{ -4 }{-5}%
2268 \TestAdd{876543210}{111111111}{987654321}%
2269 \TestAdd{999999999}{2}{1000000001}%
2270 \end{qstest}
2271
2272 \begin{qstest}{sub}{sub}
2273 \TestSub{0}{0}{0}%
2274 \TestSub{1}{0}{1}%
2275 \TestSub{1}{2}{-1}%
2276 \TestSub{-1}{-1}{0}%
2277 \TestSub{2147483646}{-1}{2147483647}%
2278 \TestSub{-2147483647}{-2147483647}{0}%
2279 \TestSub{-4}{-1}{-3}%
2280 \TestSub{-1}{-4}{3}%
2281 \TestSub{-4}{1}{-5}%
2282 \TestSub{-1}{4}{-5}%
2283 \TestSub{4}{-1}{5}%
2284 \TestSub{1}{-4}{5}%
2285 \TestSub{-4}{-1}{-3}%
2286 \TestSub{-1}{-4}{3}%
2287 \TestSub{ -4 }{ -1 }{-3}%
2288 \TestSub{ -1 }{ -4 }{3}%
2289 \TestSub{ -4 }{ 1 }{-5}%
2290 \TestSub{ -1 }{ 4 }{-5}%
2291 \TestSub{ 4 }{ -1 }{5}%
2292 \TestSub{ 1 }{ -4 }{5}%
2293 \TestSub{ -4 }{ -1 }{-3}%
2294 \TestSub{ -1 }{ -4 }{3}%
2295 \TestSub{1000000000}{2}{999999998}%
2296 \TestSub{987654321}{111111111}{876543210}%
2297 \end{qstest}
2298
2299 \begin{qstest}{shl}{shl}
2300 \TestShl{0}{0}%
2301 \TestShl{1}{2}%
2302 \TestShl{2}{4}%
2303 \TestShl{5621}{11242}%
2304 \TestShl{1073741823}{2147483646}%
2305 \end{qstest}
2306
2307 \begin{qstest}{shr}{shr}
2308 \TestShr{0}{0}%
2309 \TestShr{1}{0}%
2310 \TestShr{2}{1}%
2311 \TestShr{3}{1}%
2312 \TestShr{4}{2}%
2313 \TestShr{5}{2}%
2314 \TestShr{6}{3}%
2315 \TestShr{7}{3}%
2316 \TestShr{8}{4}%
2317 \TestShr{9}{4}%

```

```

2318 \TestShr{10}{5}%
2319 \TestShr{11}{5}%
2320 \TestShr{12}{6}%
2321 \TestShr{13}{6}%
2322 \TestShr{14}{7}%
2323 \TestShr{15}{7}%
2324 \TestShr{16}{8}%
2325 \TestShr{17}{8}%
2326 \TestShr{18}{9}%
2327 \TestShr{19}{9}%
2328 \TestShr{20}{10}%
2329 \TestShr{21}{10}%
2330 \TestShr{22}{11}%
2331 \TestShr{11241}{5620}%
2332 \TestShr{73054202}{36527101}%
2333 \TestShr{2147483646}{1073741823}%
2334 \end{qstest}
2335
2336 \begin{qstest}{mul}{mul}
2337 \TestMul{0}{0}{0}%
2338 \TestMul{1}{0}{0}%
2339 \TestMul{0}{1}{0}%
2340 \TestMul{1}{1}{1}%
2341 \TestMul{3}{1}{3}%
2342 \TestMul{1}{-3}{-3}%
2343 \TestMul{-4}{-5}{20}%
2344 \TestMul{3}{7}{21}%
2345 \TestMul{7}{3}{21}%
2346 \TestMul{3}{-7}{-21}%
2347 \TestMul{7}{-3}{-21}%
2348 \TestMul{-3}{7}{-21}%
2349 \TestMul{-7}{3}{-21}%
2350 \TestMul{-3}{-7}{21}%
2351 \TestMul{-7}{-3}{21}%
2352 \TestMul{12}{11}{132}%
2353 \TestMul{999}{333}{332667}%
2354 \TestMul{1000}{4321}{4321000}%
2355 \TestMul{12345}{173955}{2147474475}%
2356 \TestMul{1073741823}{2}{2147483646}%
2357 \TestMul{2}{1073741823}{2147483646}%
2358 \TestMul{-1073741823}{2}{-2147483646}%
2359 \TestMul{2}{-1073741823}{-2147483646}%
2360 \TestMul{6706022400}{13}{87178291200}%
2361 \end{qstest}
2362
2363 \begin{qstest}{sqr}{sqr}
2364 \TestSqr{0}{0}%
2365 \TestSqr{1}{1}%
2366 \TestSqr{2}{4}%
2367 \TestSqr{3}{9}%
2368 \TestSqr{4}{16}%
2369 \TestSqr{9}{81}%
2370 \TestSqr{10}{100}%
2371 \TestSqr{46340}{2147395600}%
2372 \TestSqr{-1}{1}%
2373 \TestSqr{-2}{4}%
2374 \TestSqr{-46340}{2147395600}%
2375 \end{qstest}
2376
2377 \begin{qstest}{fac}{fac}
2378 \TestFac{0}{1}%
2379 \TestFac{1}{1}%

```

```

2380 \TestFac{2}{2}%
2381 \TestFac{3}{2*3}%
2382 \TestFac{4}{2*3*4}%
2383 \TestFac{5}{2*3*4*5}%
2384 \TestFac{6}{2*3*4*5*6}%
2385 \TestFac{7}{2*3*4*5*6*7}%
2386 \TestFac{8}{2*3*4*5*6*7*8}%
2387 \TestFac{9}{2*3*4*5*6*7*8*9}%
2388 \TestFac{10}{2*3*4*5*6*7*8*9*10}%
2389 \TestFac{11}{2*3*4*5*6*7*8*9*10*11}%
2390 \TestFac{12}{2*3*4*5*6*7*8*9*10*11*12}%
2391 \TestFacBig{13}{6227020800}%
2392 \TestFacBig{14}{87178291200}%
2393 \TestFacBig{15}{1307674368000}%
2394 \TestFacBig{16}{20922789888000}%
2395 \TestFacBig{17}{355687428096000}%
2396 \TestFacBig{18}{6402373705728000}%
2397 \TestFacBig{19}{121645100408832000}%
2398 \TestFacBig{20}{2432902008176640000}%
2399 \TestFacBig{21}{51090942171709440000}%
2400 \TestFacBig{22}{112400072777607680000}%
2401 \end{qstest}
2402
2403 \begin{qstest}{pow}{pow}
2404 \TestPow{-2}{0}{1}%
2405 \TestPow{-1}{0}{1}%
2406 \TestPow{0}{0}{1}%
2407 \TestPow{1}{0}{1}%
2408 \TestPow{2}{0}{1}%
2409 \TestPow{3}{0}{1}%
2410 \TestPow{-2}{1}{-2}%
2411 \TestPow{-1}{1}{-1}%
2412 \TestPow{1}{1}{1}%
2413 \TestPow{2}{1}{2}%
2414 \TestPow{3}{1}{3}%
2415 \TestPow{-2}{2}{4}%
2416 \TestPow{-1}{2}{1}%
2417 \TestPow{0}{2}{0}%
2418 \TestPow{1}{2}{1}%
2419 \TestPow{2}{2}{4}%
2420 \TestPow{3}{2}{9}%
2421 \TestPow{0}{1}{0}%
2422 \TestPow{1}{-2}{1}%
2423 \TestPow{1}{-1}{1}%
2424 \TestPow{-1}{-2}{1}%
2425 \TestPow{-1}{-1}{-1}%
2426 \TestPow{-1}{3}{-1}%
2427 \TestPow{-1}{4}{1}%
2428 \TestPow{-2}{-1}{0}%
2429 \TestPow{-2}{-2}{0}%
2430 \TestPow{2}{3}{8}%
2431 \TestPow{2}{4}{16}%
2432 \TestPow{2}{5}{32}%
2433 \TestPow{2}{6}{64}%
2434 \TestPow{2}{7}{128}%
2435 \TestPow{2}{8}{256}%
2436 \TestPow{2}{9}{512}%
2437 \TestPow{2}{10}{1024}%
2438 \TestPow{-2}{3}{-8}%
2439 \TestPow{-2}{4}{16}%
2440 \TestPow{-2}{5}{-32}%
2441 \TestPow{-2}{6}{64}%

```

```

2442 \TestPow{-2}{7}{-128}%
2443 \TestPow{-2}{8}{256}%
2444 \TestPow{-2}{9}{-512}%
2445 \TestPow{-2}{10}{1024}%
2446 \TestPow{3}{3}{27}%
2447 \TestPow{3}{4}{81}%
2448 \TestPow{3}{5}{243}%
2449 \TestPow{-3}{3}{-27}%
2450 \TestPow{-3}{4}{81}%
2451 \TestPow{-3}{5}{-243}%
2452 \TestPow{2}{30}{1073741824}%
2453 \TestPow{-3}{19}{-1162261467}%
2454 \TestPow{5}{13}{1220703125}%
2455 \TestPow{-7}{11}{-1977326743}%
2456 \end{qstest}
2457
2458 \begin{qstest}{div}{div}
2459 \TestDiv{1}{1}{1}%
2460 \TestDiv{2}{1}{2}%
2461 \TestDiv{-2}{1}{-2}%
2462 \TestDiv{2}{-1}{-2}%
2463 \TestDiv{-2}{-1}{2}%
2464 \TestDiv{15}{2}{7}%
2465 \TestDiv{-16}{2}{-8}%
2466 \TestDiv{1}{2}{0}%
2467 \TestDiv{1}{3}{0}%
2468 \TestDiv{2}{3}{0}%
2469 \TestDiv{-2}{3}{0}%
2470 \TestDiv{2}{-3}{0}%
2471 \TestDiv{-2}{-3}{0}%
2472 \TestDiv{13}{3}{4}%
2473 \TestDiv{-13}{-3}{4}%
2474 \TestDiv{-13}{3}{-4}%
2475 \TestDiv{-6}{5}{-1}%
2476 \TestDiv{-5}{5}{-1}%
2477 \TestDiv{-4}{5}{0}%
2478 \TestDiv{-3}{5}{0}%
2479 \TestDiv{-2}{5}{0}%
2480 \TestDiv{-1}{5}{0}%
2481 \TestDiv{0}{5}{0}%
2482 \TestDiv{1}{5}{0}%
2483 \TestDiv{2}{5}{0}%
2484 \TestDiv{3}{5}{0}%
2485 \TestDiv{4}{5}{0}%
2486 \TestDiv{5}{5}{1}%
2487 \TestDiv{6}{5}{1}%
2488 \TestDiv{-5}{4}{-1}%
2489 \TestDiv{-4}{4}{-1}%
2490 \TestDiv{-3}{4}{0}%
2491 \TestDiv{-2}{4}{0}%
2492 \TestDiv{-1}{4}{0}%
2493 \TestDiv{0}{4}{0}%
2494 \TestDiv{1}{4}{0}%
2495 \TestDiv{2}{4}{0}%
2496 \TestDiv{3}{4}{0}%
2497 \TestDiv{4}{4}{1}%
2498 \TestDiv{5}{4}{1}%
2499 \TestDiv{12345}{678}{18}%
2500 \TestDiv{32372}{5952}{5}%
2501 \TestDiv{284271294}{18162}{15651}%
2502 \TestDiv{217652429}{12561}{17327}%
2503 \TestDiv{462028434}{5439}{84947}%

```

```

2504 \TestDiv{2147483647}{1000}{2147483}%
2505 \TestDiv{2147483647}{-1000}{-2147483}%
2506 \TestDiv{-2147483647}{1000}{-2147483}%
2507 \TestDiv{-2147483647}{-1000}{2147483}%
2508 \TestDiv{0}{3}{0}%
2509 \TestDiv{1}{3}{0}%
2510 \TestDiv{2}{3}{0}%
2511 \TestDiv{3}{3}{1}%
2512 \TestDiv{4}{3}{1}%
2513 \TestDiv{5}{3}{1}%
2514 \TestDiv{6}{3}{2}%
2515 \TestDiv{7}{3}{2}%
2516 \TestDiv{8}{3}{2}%
2517 \TestDiv{9}{3}{3}%
2518 \TestDiv{10}{3}{3}%
2519 \TestDiv{11}{3}{3}%
2520 \TestDiv{12}{3}{4}%
2521 \TestDiv{13}{3}{4}%
2522 \TestDiv{14}{3}{4}%
2523 \TestDiv{15}{3}{5}%
2524 \TestDiv{16}{3}{5}%
2525 \TestDiv{17}{3}{5}%
2526 \TestDiv{18}{3}{6}%
2527 \TestDiv{19}{3}{6}%
2528 \TestDiv{20}{3}{6}%
2529 \TestDiv{21}{3}{7}%
2530 \TestDiv{22}{3}{7}%
2531 \TestDiv{23}{3}{7}%
2532 \TestDiv{24}{3}{8}%
2533 \TestDiv{25}{3}{8}%
2534 \TestDiv{26}{3}{8}%
2535 \TestDiv{27}{3}{9}%
2536 \TestDiv{28}{3}{9}%
2537 \TestDiv{29}{3}{9}%
2538 \TestDiv{30}{3}{10}%
2539 \TestDiv{31}{3}{10}%
2540 \TestDivBig{17363436332507}{24702}{702916214}%
2541 \end{qstest}
2542
2543 \begin{qstest}{mod}{mod}
2544 \TestMod{-6}{5}{4}%
2545 \TestMod{-5}{5}{0}%
2546 \TestMod{-4}{5}{1}%
2547 \TestMod{-3}{5}{2}%
2548 \TestMod{-2}{5}{3}%
2549 \TestMod{-1}{5}{4}%
2550 \TestMod{0}{5}{0}%
2551 \TestMod{1}{5}{1}%
2552 \TestMod{2}{5}{2}%
2553 \TestMod{3}{5}{3}%
2554 \TestMod{4}{5}{4}%
2555 \TestMod{5}{5}{0}%
2556 \TestMod{6}{5}{1}%
2557 \TestMod{-5}{4}{3}%
2558 \TestMod{-4}{4}{0}%
2559 \TestMod{-3}{4}{1}%
2560 \TestMod{-2}{4}{2}%
2561 \TestMod{-1}{4}{3}%
2562 \TestMod{0}{4}{0}%
2563 \TestMod{1}{4}{1}%
2564 \TestMod{2}{4}{2}%
2565 \TestMod{3}{4}{3}%

```

```

2566 \TestMod{4}{4}{0}%
2567 \TestMod{5}{4}{1}%
2568 \TestMod{-6}{-5}{-1}%
2569 \TestMod{-5}{-5}{0}%
2570 \TestMod{-4}{-5}{-4}%
2571 \TestMod{-3}{-5}{-3}%
2572 \TestMod{-2}{-5}{-2}%
2573 \TestMod{-1}{-5}{-1}%
2574 \TestMod{0}{-5}{0}%
2575 \TestMod{1}{-5}{-4}%
2576 \TestMod{2}{-5}{-3}%
2577 \TestMod{3}{-5}{-2}%
2578 \TestMod{4}{-5}{-1}%
2579 \TestMod{5}{-5}{0}%
2580 \TestMod{6}{-5}{-4}%
2581 \TestMod{-5}{-4}{-1}%
2582 \TestMod{-4}{-4}{0}%
2583 \TestMod{-3}{-4}{-3}%
2584 \TestMod{-2}{-4}{-2}%
2585 \TestMod{-1}{-4}{-1}%
2586 \TestMod{0}{-4}{0}%
2587 \TestMod{1}{-4}{-3}%
2588 \TestMod{2}{-4}{-2}%
2589 \TestMod{3}{-4}{-1}%
2590 \TestMod{4}{-4}{0}%
2591 \TestMod{5}{-4}{-3}%
2592 \TestMod{2147483647}{1000}{647}%
2593 \TestMod{2147483647}{-1000}{-353}%
2594 \TestMod{-2147483647}{1000}{353}%
2595 \TestMod{-2147483647}{-1000}{-647}%
2596 \TestMod{ 0 }{ 4 }{0}%
2597 \TestMod{ 1 }{ 4 }{1}%
2598 \TestMod{ -1 }{ 4 }{3}%
2599 \TestMod{ 0 }{ -4 }{0}%
2600 \TestMod{ 1 }{ -4 }{-3}%
2601 \TestMod{ -1 }{ -4 }{-1}%
2602 \TestMod{18362}{25}{12}%
2603 \end{qstest}
2604
2605 \newcommand*{\TestError}[2]{%
2606   \begingroup
2607     \expandafter\def\csname BigIntCalcError:#1\endcsname{%
2608       \Expect*{#2}{0}%
2609       \expandafter\def\csname BigIntCalcError:#1\endcsname{ERROR}%
2610       \Expect*{#2}{OERROR}%
2611     \endgroup
2612 }
2613 \begin{qstest}{error}{error}
2614   \TestError{FacNegative}{\bigintcalcFac{-1}}%
2615   \TestError{FacNegative}{\bigintcalcFac{-2147483647}}%
2616   \TestError{DivisionByZero}{\bigintcalcPow{0}{-1}}%
2617   \TestError{DivisionByZero}{\bigintcalcDiv{1}{0}}%
2618   \TestError{DivisionByZero}{\bigintcalcMod{1}{0}}%
2619 \end{qstest}
2620
2621 \begin{document}
2622 \end{document}
2623 </test2>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/bigintcalc.dtx](#) The source file.

[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:install/macros/latex/contrib/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, miK_TE_X, ...) relies on file name databases, you must refresh these. For example, teT_EX users run `texhash` or `mktextlsr`.

¹<http://ftp.ctan.org/tex-archive/>

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	
<code>\#</code>	1752 , 1814
<code>\%</code>	1817
<code>\@</code>	1753 , 1810
<code>\@firstofone</code>	1761 , 1764
<code>\@firstoftwo</code>	137 , 145 , 154
<code>\@gobble</code>	1758 , 1766
<code>\@ne</code>	2138 , 2153 , 2154 , 2164 , 2165 , 2190 , 2191
<code>\@nil</code> .	133 , 135 , 142 , 150 , 165 , 168 , 173
<code>\@secondoftwo</code>	139 , 147 , 157
<code>\@undefined</code>	53
<code>\[</code>	1815
<code>\]</code>	170 , 180 , 299 , 300 , 308 , 331 , 429 , 441 , 474 , 487 , 499 , 534 , 536 , 625 , 626 , 636 , 637 , 654 , 716 , 717 , 727 , 728 , 745 , 830 , 841 , 888 , 902 , 965 , 976 , 993 , 1113 ,

	1199, 1224, 1252, 1300, 1321, 1331, 1425, 1429, 1457, 1503, 1650, 1661, 1669, 1691, 1707, 1811	998, 1005, 1007, 1055, 1114, 1119, 1147, 1175, 1177, 1210, 1212, 1307, 1392, 1421, 1423, 1458, 1462, 1514, 1555, 1604, 1659, 1698, 1732, 1734, 1739, 1744
\{	1750, 1812	
\}	1751, 1813	
\]	1816	\BIC@AfterFiFi
		. 116, 181, 185, 191, 212, 216, 275, 279, 285, 289, 301, 303, 309, 313, 327, 376, 378, 442, 444, 450, 467, 476, 478, 500, 502, 508, 527, 542, 627, 631, 647, 718, 722, 738, 889, 891, 904, 911, 920, 927, 1013, 1018, 1058, 1083, 1167, 1215, 1219, 1237, 1239, 1248, 1268, 1270, 1291, 1302, 1304, 1333, 1337, 1343, 1395, 1397, 1401, 1407, 1409, 1413, 1438, 1440, 1444, 1449, 1493, 1495, 1501, 1523, 1525, 1547, 1570, 1572, 1578, 1580, 1586, 1588, 1597, 1599, 1620, 1627, 1662, 1664, 1678, 1680, 1684, 1701, 1703, 1713, 1720
_	1818	\BIC@AfterFiFiFi 117, 196, 200, 332, 336, 537, 539, 569, 573, 579, 582, 586, 594, 596, 601, 607, 611, 638, 642, 729, 733, 1061, 1065, 1072, 1076, 1087, 1091, 1097, 1101, 1151, 1152, 1153, 1154, 1155, 1156, 1157, 1158, 1159, 1160, 1161, 1162, 1163, 1242, 1244, 1273, 1275, 1280, 1285, 1350, 1356, 1504, 1508, 1532, 1534, 1540, 1542, 1613, 1615
A		
\advance	1791, 1799, 2066	\BIC@AtEnd 81, 82, 1747
\AtEndDocument	2079	\BIC@Cmp 265, 268
B		
\begin	2086, 2104, 2123, 2146, 2157, 2168, 2196, 2208, 2226, 2243, 2272, 2299, 2307, 2336, 2363, 2377, 2403, 2458, 2543, 2613, 2621	\BIC@CmpDiff 301, 320
\BIC@@@Shl	848, 852	\BIC@CmpLength 290, 296, 298
\BIC@@@Shr	921, 922, 928, 929, 938	\BIC@CmpResult 304, 310, 319
\BIC@@@Dec	509, 510, 528, 533	\BIC@Dec 399, 413, 426, 486, 1186
\BIC@@@Inc	451, 452, 468, 473	\BIC@DecSwitch 405, 408
\BIC@@@PowRec	1351, 1357, 1372	\BIC@Div 1379, 1382
\BIC@@@ProcessDiv	1556, 1594	\BIC@DivCleanup
\BIC@@@Shl	832, 840, 855, 859	. 1493, 1501, 1519, 1523, 1532, 1540, 1570, 1578, 1586, 1597, 1613
\BIC@@@Shr	892, 897, 901, 939	\BIC@DivStart 1483, 1487
\BIC@@AddDigit	677, 678, 688, 767	\BIC@DivStartX 1450, 1456, 1463
\BIC@@Cmp	269, 272, 358, 390	\BIC@DivStartYii 1459, 1467
\BIC@@Dec	489, 497, 546	\BIC@DivStartYiv 1468, 1472
\BIC@@Expand	133, 135	\BIC@DivStartYvi 1473, 1477
\BIC@@Inc	431, 439, 482	\BIC@DivStartYviii 1478, 1482
\BIC@@MinMax	354, 357	\BIC@DivSub 1543, 1560, 1589
\BIC@@MinusOne	1314, 1319	\BIC@DivSwitch
\BIC@@PowRec	1344, 1367, 1373 1398, 1402, 1410, 1414, 1419
\BIC@@ProcessDiv	1515, 1520	\BIC@DivSwitchSign . . 1383, 1388, 1390
\BIC@@ProcessFac	1178, 1184	\BIC@DoAdd 628, 632, 653
\BIC@@ProcessTim	985, 992	\BIC@DoSub 719, 723, 744
\BIC@@Shl	814, 819, 826, 829	\BIC@Expand 131, 161, 228
\BIC@@Shr	872, 876, 882, 884, 1345, 1374, 1441, 1445, 1446	\BIC@Fac 1142, 1145
\BIC@@Sqr	1132, 1134, 1137	\BIC@Fi . . . 114, 115, 116, 117, 175, 205, 220, 293, 317, 341, 362, 382, 401, 420, 437, 471, 484,
\BIC@@SubDigit	766, 777	
\BIC@@TestMode	107	
\BIC@@Tim	964	
\BIC@Abs	236, 239	
\BIC@Add	552, 555, 561	
\BIC@AddCarry0	695	
\BIC@AddCarry10	696	
\BIC@AddCarry[1-9]	697	
\BIC@AddDigit	659, 664, 675	
\BIC@AddResult	658, 668	
\BIC@AddSwitch	557, 564	
\BIC@AddXY	570, 574, 608, 612, 619, 624, 815, 820, 827, 1021, 1479	
\BIC@AfterFi	115, 172, 209, 322, 359, 361, 391, 393, 397, 410, 412, 416, 430, 434, 481, 488, 492, 545, 655, 662, 690, 692, 746, 751, 779, 784, 812, 818, 831, 835, 842, 847, 854, 858, 871, 875, 886, 896, 966, 970, 977, 984, 994,	

495, 531, 548, 616, 651, 666,
 693, 742, 755, 785, 822, 838,
 850, 861, 878, 899, 934, 973,
 990, 1001, 1026, 1106, 1123,
 1171, 1182, 1297, 1310, 1365,
 1417, 1454, 1465, 1517, 1558,
 1592, 1634, 1688, 1728, 1735, 1745
 \BIC@Inc 394, 418, 423, 428
 \BIC@IncSwitch 386, 389
 \BIC@MinMax 345, 350, 353
 \BIC@MinusOne 1230, 1259, 1312
 \BIC@Mod 1638, 1641
 \BIC@ModMinus 1710, 1730
 \BIC@ModSwitch . 1665, 1681, 1685, 1690
 \BIC@ModSwitchSign .. 1642, 1647, 1649
 \BIC@ModTwo
 1241, 1272, 1279, 1299, 1342, 1700
 \BIC@ModX 1714, 1737
 \BIC@Mul 1046, 1049
 \BIC@MulDigit[3-9] 1028
 \BIC@MulSwitch 1050, 1053
 \BIC@Normalize 178, 225
 \BIC@NormalizeDigits .. 201, 217, 222
 \BIC@NormalizeZero 197, 207
 \BIC@Odd 366, 371, 373
 \BIC@PosCmp 295, 568, 578, 593,
 606, 1060, 1071, 1086, 1096,
 1149, 1174, 1332, 1349, 1420,
 1491, 1521, 1530, 1568, 1595, 1611
 \BIC@Pow 1191, 1194
 \BIC@PowRec 1281, 1286, 1292, 1330, 1368
 \BIC@PowSwitch 1195, 1198
 \BIC@ProcessDiv
 1456, 1488, 1490, 1526,
 1535, 1561, 1573, 1581, 1600, 1616
 \BIC@ProcessDivII
 1548, 1567, 1605, 1621, 1628
 \BIC@ProcessDivIV 1594
 \BIC@ProcessFac 1168, 1173, 1185
 \BIC@ProcessMul . 1062, 1066, 1073,
 1077, 1088, 1092, 1098, 1102,
 1110, 1112, 1138, 1179, 1216,
 1220, 1334, 1338, 1352, 1358, 1369
 \BIC@ProcessTim ... 967, 975, 995, 999
 \BIC@Sgn 249, 252, 409,
 1054, 1057, 1082, 1391, 1394, 1406
 \BIC@Shl 807, 810, 1469, 1474, 1484
 \BIC@Shr 865, 868
 \BIC@ShrDigit[00-19] 941
 \BIC@ShrResult 905, 906, 912, 913, 936
 \BIC@Space 118,
 162, 231, 241, 243, 656, 670,
 672, 761, 780, 843, 978, 1008,
 1014, 1019, 1115, 1709, 1721, 1740
 \BIC@Sqr 1127, 1130
 \BIC@StripHexSpace 165, 168
 \BIC@SubCarry0 787
 \BIC@SubCarry10 788
 \BIC@SubCarry[1-9] 789
 \BIC@SubDigit 748, 753, 764
 \BIC@SubResult 747, 757
 \BIC@SubXY
 . 583, 587, 597, 602, 622, 715,
 1549, 1563, 1606, 1622, 1629, 1741
 \BIC@Temp 697, 705,
 706, 707, 708, 709, 710, 711,
 712, 713, 714, 789, 796, 797,
 798, 799, 800, 801, 802, 803,
 804, 941, 944, 945, 946, 947,
 948, 949, 950, 951, 952, 953,
 954, 955, 956, 957, 958, 959,
 960, 961, 962, 963, 1028, 1037,
 1038, 1039, 1040, 1041, 1042, 1043
 \BIC@TestMode 107, 1834
 \BIC@Tim 964, 1116, 1121
 \BIC@TimDigit 980, 987, 1003
 \bigintcalcAbs .. 4, 234, 367, 1884,
 1903, 1974, 1975, 1983, 1984,
 1993, 1994, 2020, 2021, 2041, 2042
 \BigIntCalcAdd 7, 618, 1937, 1945
 \bigintcalcAdd 5, 550, 1116, 1121, 1931
 \bigintcalcCmp 4, 263, 1896, 1934, 1958
 \BigIntCalcDec 7, 425, 1924
 \bigintcalcDec 5, 403, 1920
 \BigIntCalcDiv 7, 1386, 2020
 \bigintcalcDiv 6, 1377,
 1716, 1723, 1872, 2013, 2017, 2617
 \BigIntCalcError 462,
 512, 522, 701, 792, 1033, 1147,
 1164, 1234, 1237, 1245, 1249,
 1276, 1288, 1294, 1296, 1362,
 1364, 1392, 1452, 1659, 1704, 1727
 \bigintcalcFac
 .. 6, 1140, 2004, 2007, 2614, 2615
 \BigIntCalcInc 7, 422, 1913
 \bigintcalcInc 5, 384, 1909
 \bigintcalcInv 3, 230, 1881
 \bigintcalcMax 4, 348, 1893
 \bigintcalcMin 4, 343, 1890
 \BigIntCalcMod 7, 1645, 2040
 \bigintcalcMod ... 6, 1636, 2025, 2618
 \BigIntCalcMul 7, 1108, 1992
 \bigintcalcMul 6, 1044, 1716, 1723, 1989
 \bigintcalcNum .. 3, 223, 232, 237,
 250, 266, 270, 346, 351, 355,
 387, 406, 553, 557, 562, 808,
 866, 1047, 1051, 1128, 1143,
 1192, 1196, 1380, 1384, 1639,
 1643, 1913, 1924, 1938, 1946, 1962
 \BigIntCalcOdd 7, 369, 1902
 \bigintcalcOdd 5, 364, 1899
 \bigintcalcPow ... 6, 1189, 2010, 2616
 \bigintcalcSgn
 4, 247, 1887, 1910, 1921, 1932,
 1933, 1956, 1957, 2026, 2028, 2033
 \BigIntCalcShl 7, 824, 1974
 \bigintcalcShl 5, 805, 1971
 \BigIntCalcShr 7, 880, 1983
 \bigintcalcShr 5, 863, 1980
 \bigintcalcSqr 6, 1125, 1999
 \BigIntCalcSub 7, 621, 1961
 \bigintcalcSub 5, 559, 1715, 1722, 1955
 \body 1770, 1774

C		
<code>\catcode</code>	3, 4, 5, 6, 7, 18, 19, 20, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 65, 66, 69, 70, 71, 72, 76, 77, 78, 79, 83, 85, 105, 110, 112, 1750, 1751, 1752, 1753, 1788, 1797, 1810, 1811, 1812, 1813, 1814, 1815, 1816, 1817, 1818, 1819	
<code>\chardef</code>	1834	
<code>\count@</code>	1755, 1784, 1788, 1790, 1791, 1795, 1797, 1798, 1799	
<code>\countdef</code>	1755	
<code>\csname</code>	8, 21, 45, 61, 68, 103, 109, 124, 130, 154, 157, 679, 695, 696, 698, 768, 787, 788, 790, 907, 914, 923, 930, 942, 1009, 1015, 1022, 1029, 1754, 1757, 1760, 1763, 1802, 1824, 2048, 2607, 2609	
D		
<code>\dimexpr</code>	2056	
<code>\divide</code>	1871	
<code>\documentclass</code>	1830	
E		
<code>\empty</code>	12	
<code>\end</code>	1825, 2102, 2121, 2144, 2155, 2166, 2194, 2206, 2224, 2241, 2270, 2297, 2305, 2334, 2361, 2375, 2401, 2456, 2541, 2603, 2619, 2622	
<code>\endcsname</code>	8, 21, 45, 61, 68, 103, 109, 124, 130, 154, 157, 686, 695, 696, 698, 775, 787, 788, 790, 907, 914, 923, 930, 942, 1009, 1015, 1022, 1029, 1754, 1757, 1760, 1763, 1802, 1824, 2048, 2607, 2609	
<code>\endinput</code>	30	
<code>\endqstest</code>	2070, 2075	
<code>\Expect</code>	1847, 1854, 1863, 1872, 2608, 2610	
I		
<code>\ifcase</code>	9, 390, 409, 440, 453, 498, 511, 535, 578, 593, 681, 684, 699, 770, 773, 791, 885, 1004, 1012, 1030, 1054, 1057, 1082, 1150, 1199, 1224, 1230, 1241, 1252, 1259, 1272, 1279, 1331, 1342, 1391, 1394, 1406, 1420, 1429, 1491, 1521, 1530, 1568, 1595, 1611, 1650, 1661, 1669, 1691, 1700, 2026, 2028, 2033	
<code>\ifcat</code>	136	
<code>\ifnum</code>	321, 326, 358, 449, 475, 507, 568, 606, 689, 778, 853, 1060, 1071, 1086, 1096, 1149, 1174, 1332, 1349, 1790, 1798, 1910, 1921, 1932, 1933, 1934, 1956, 1957, 1958	
<code>\ifodd</code>	375, 903, 919, 1301	
<code>\iftrue</code>	2142	
<code>\ifx</code>	10, 12, 21, 45, 53, 56, 103, 109, 124, 130, 144, 154, 157, 170, 179, 180, 190, 195, 208, 211, 240, 253, 256, 273, 274, 284, 299, 300, 308, 331, 374, 429, 441, 474, 487, 499, 534, 536, 565, 566, 592, 625, 626, 636, 637, 654, 669, 716, 717, 727, 728, 745, 758, 811, 830, 841, 869, 888, 902, 965, 976, 993, 1113, 1131, 1146, 1199, 1200, 1201, 1202, 1206, 1214, 1224, 1225, 1226, 1252, 1253, 1254, 1258, 1300, 1313, 1320, 1321, 1331, 1425, 1429, 1430, 1431, 1432, 1457, 1492, 1500, 1503, 1522, 1531, 1539, 1569, 1577, 1585, 1596, 1612, 1650, 1651, 1655, 1661, 1669, 1670, 1674, 1691, 1692, 1693, 1707, 1712, 1731, 1738, 1754, 1757, 1760, 1763, 1802, 2048, 2100, 2101, 2120	
<code>\immediate</code>	23, 47	
<code>\IncludeTests</code>	1839	
<code>\input</code>	125, 1803	
<code>\iterate</code>	1771, 1773, 1775	
L		
<code>\LoadCommand</code>	1803, 1820	
<code>\LogTests</code>	1840	
<code>\loop</code>	1769, 1785, 1796	
M		
<code>\m@ne</code>	2139, 2154, 2165, 2191	
<code>\makeatletter</code>	1833, 2050, 2084	
<code>\makeatother</code>	1835, 2082	
N		
<code>\NeedsTeXFormat</code>	1828	
<code>\newcommand</code>	1841, 1849, 1856, 1867, 1868, 1869, 1874, 1879, 1880, 1883, 1886, 1889, 1892, 1895, 1898, 1908, 1919, 1930, 1954, 1970, 1979, 1988, 1998, 2001, 2006, 2009, 2012, 2016, 2024, 2054, 2059, 2063, 2064, 2605	
<code>\newcount</code>	1866, 2051, 2052	
<code>\next</code>	1775, 1777, 1779	
<code>\nofiles</code>	1829	
<code>\number</code>	248, 264, 678, 767, 1009, 1015, 1022, 2056, 2099, 2137, 2138, 2139, 2141, 2153, 2154, 2164, 2165, 2190, 2191	
<code>\numexpr</code>	451, 509, 657, 677, 766, 781, 844, 848, 905, 912, 921, 928, 979, 986, 1831, 1832, 1837, 1842, 1843, 1846, 1850, 1851, 1853, 1859, 1867, 2003	
P		
<code>\PackageInfo</code>	26	
<code>\pdf@escapehex</code>	165	
<code>\pdf@unescapehex</code>	163	

\pdfelapsedtime	2065	2178, 2179, 2180, 2181, 2182,
\pdfresettimer	2061	2183, 2184, 2185, 2186, 2187,
\PrintTime	2054, 2067, 2080	2188, 2189, 2190, 2191, 2192, 2193
\ProvidesPackage	62	\TestCount 1866, 1870, 1871, 1872
Q		
\qstest	2069, 2071	\TestDec .. 1919, 2227, 2228, 2229,
R		
\RangeCatcodeInvalid		2230, 2231, 2232, 2233, 2234,
.....	1794, 1806, 1807, 1808, 1809	2235, 2236, 2237, 2238, 2239, 2240
\renewcommand	2060	\TestDiv
\repeat	1769, 1781, 1792, 1800	2012,
\RequirePackage	127	2459, 2460, 2461, 2462, 2463,
\RestoreCatcodes	1783, 1786, 1787, 1821	2464, 2465, 2466, 2467, 2468,
\result	1852, 1854	2469, 2470, 2471, 2472, 2473,
\resultA	1844, 1847	2474, 2475, 2476, 2477, 2478,
\resultB	1845, 1847	2479, 2480, 2481, 2482, 2483,
\romannumeral	132, 162, 224, 231, 235,	2484, 2485, 2486, 2487, 2488,
	344, 349, 365, 370, 385, 398,	2489, 2490, 2491, 2492, 2493,
	404, 417, 423, 426, 551, 560,	2494, 2495, 2496, 2497, 2498,
	567, 581, 600, 619, 622, 676,	2499, 2500, 2501, 2502, 2503,
	765, 806, 813, 825, 864, 870,	2504, 2505, 2506, 2507, 2508,
	881, 980, 987, 1020, 1045, 1070,	2509, 2510, 2511, 2512, 2513,
	1085, 1109, 1126, 1141, 1179,	2514, 2515, 2516, 2517, 2518,
	1186, 1190, 1284, 1344, 1351,	2519, 2520, 2521, 2522, 2523,
	1357, 1368, 1373, 1378, 1387,	2524, 2525, 2526, 2527, 2528,
	1427, 1445, 1468, 1473, 1478,	2529, 2530, 2531, 2532, 2533,
	1483, 1548, 1562, 1605, 1621,	2534, 2535, 2536, 2537, 2538, 2539
	1628, 1637, 1646, 1709, 1710, 1740	\TestDivBig
S		
\saved@endqstest	2070, 2077	2016, 2540
\saved@qstest	2069, 2072	\TestError
\SavedNumexpr		2605, 2614, 2615, 2616, 2617, 2618
	1831, 1837, 1842, 1846, 1850, 1853	\TestExch
\space	1847, 2056	1879, 2002
\StartTime	2059, 2073	\TestFac
\StopTime	2064, 2076	2001, 2378, 2379,
\strip@pt	2056	2380, 2381, 2382, 2383, 2384,
\SummaryTime ...	2051, 2053, 2066, 2080	2385, 2386, 2387, 2388, 2389, 2390
T		
\Test	1805, 1823, 1874, 1879,	\TestFacBig
	1881, 1884, 1887, 1890, 1893,	. 2006, 2391, 2392, 2393, 2394,
	1896, 1899, 1901, 1909, 1912,	2395, 2396, 2397, 2398, 2399, 2400
	1920, 1923, 1931, 1936, 1944,	\TestInc 1908, 2209, 2210, 2211, 2212,
	1955, 1960, 1971, 1973, 1980,	2213, 2214, 2215, 2216, 2217,
	1982, 1989, 1991, 1999, 2007,	2218, 2219, 2220, 2221, 2222, 2223
	2010, 2013, 2017, 2019, 2025, 2039	\TestInv 1880, 2087, 2088, 2089, 2090,
\TestAbs	1883,	2091, 2092, 2093, 2094, 2095,
	2105, 2106, 2107, 2108, 2109,	2096, 2097, 2098, 2099, 2100, 2101
	2110, 2111, 2112, 2113, 2114,	\TestMax
	2115, 2116, 2117, 2118, 2119, 2120	1892, 2158, 2159,
\TestAdd	1930,	2160, 2161, 2162, 2163, 2164, 2165
	2244, 2245, 2246, 2247, 2248,	\TestMin
	2249, 2250, 2251, 2252, 2253,	1889, 2147, 2148,
	2254, 2255, 2256, 2257, 2258,	2149, 2150, 2151, 2152, 2153, 2154
	2259, 2260, 2261, 2262, 2263,	\TestMod .. 2024, 2544, 2545, 2546,
	2264, 2265, 2266, 2267, 2268, 2269	2547, 2548, 2549, 2550, 2551,
\TestArg	1867, 1868, 1870, 1871	2552, 2553, 2554, 2555, 2556,
\TestCmp	1895, 2169, 2170, 2171, 2172,	2557, 2558, 2559, 2560, 2561,
	2173, 2174, 2175, 2176, 2177,	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,
		2597, 2598, 2599, 2600, 2601, 2602
		\TestMul .. 1988, 2337, 2338, 2339,
		2340, 2341, 2342, 2343, 2344,
		2345, 2346, 2347, 2348, 2349,
		2350, 2351, 2352, 2353, 2354,
		2355, 2356, 2357, 2358, 2359, 2360
		\TestOdd
		1898, 2198, 2199,

2200, 2201, 2202, 2203, 2204, 2205	2286, 2287, 2288, 2289, 2290,
\TestPow 2009, 2404,	2291, 2292, 2293, 2294, 2295, 2296
2405, 2406, 2407, 2408, 2409,	\TestTeXDivide 1869, 2014
2410, 2411, 2412, 2413, 2414,	\TestTime 2052, 2065, 2066, 2067
2415, 2416, 2417, 2418, 2419,	\the 69, 70,
2420, 2421, 2422, 2423, 2424,	71, 72, 83, 451, 509, 657, 677,
2425, 2426, 2427, 2428, 2429,	766, 781, 844, 848, 905, 912,
2430, 2431, 2432, 2433, 2434,	921, 928, 979, 986, 1788, 1872, 2003
2435, 2436, 2437, 2438, 2439,	\TimeDescription . . . 2060, 2063, 2067
2440, 2441, 2442, 2443, 2444,	\TMP@EnsureCode
2445, 2446, 2447, 2448, 2449, 80, 87, 88, 89, 90, 91, 92,
2450, 2451, 2452, 2453, 2454, 2455	93, 94, 95, 96, 97, 98, 99, 100, 101
\TestResult 1849, 1875	\tracingmacros 2197
\TestResultTwoExpansions . 1856, 1876	\typeout 2055
\TestSgn 1886, 2124,	
2125, 2126, 2127, 2128, 2129,	U
2130, 2131, 2132, 2133, 2134,	\UNDEFINED 1832, 1843, 1851, 1859
2135, 2136, 2137, 2138, 2139, 2140	\usepackage 1836, 1838
\TestShl	
1970, 2300, 2301, 2302, 2303, 2304	W
\TestShr 1979,	\write 23, 47
2308, 2309, 2310, 2311, 2312,	
2313, 2314, 2315, 2316, 2317,	X
2318, 2319, 2320, 2321, 2322,	\x 8, 10, 12, 22, 26,
2323, 2324, 2325, 2326, 2327,	28, 46, 51, 61, 67, 75, 119, 122,
2328, 2329, 2330, 2331, 2332, 2333	1900, 1906, 1911, 1916, 1922,
\TestSpaceAtEnd 1841, 1877	1927, 1935, 1941, 1943, 1949,
\TestSqr 1998,	1959, 1965, 1972, 1977, 1981,
2364, 2365, 2366, 2367, 2368,	1986, 1990, 1996, 2018, 2038, 2044
2369, 2370, 2371, 2372, 2373, 2374	
\TestSub . . 1954, 2273, 2274, 2275,	Z
2276, 2277, 2278, 2279, 2280,	\z@ 2053,
2281, 2282, 2283, 2284, 2285,	2099, 2137, 2141, 2153, 2164, 2190