Pari-GP reference card

Pari-GP reference card

Note: optional arguments are surrounded by {}.
To start the calculator, type its name in the terminal: gp
To exit gp, type quit, q, or C-D at prompt.

Help
describe function
extended description
list of relevant help topics
???pattern
name of GP-1.39 function in GP-2.*

Input/Output
previous result, the result before n-th result since startup
separate multiple statements on line
extend statement on additional lines
extend statements on several lines
comment
one-line comment, rest of line ignored

Metacommands & Defaults
set default to s? call toggle timer on/off
print time for last result
print defaults
set debug level to n
set memory debug level to n
set n significant digits / bits
set n terms in series
quit gp
print the list of PARI types
print the list of user-defined functions
read file into GP

Debugger / break loop
go out of break loop
examine object o
set break point
set current error data
number of objects on heap and their size
total size of objects on PARI stack

PARI Types & Input Formats

\texttt{t_INT}. Integers; hex, binary
\texttt{t_REAL}. Reals
\texttt{t_INTMOD}. Integers modulo m
\texttt{t_FRAC}. Rational Numbers
\texttt{t_FFELT}. Elt in finite field \( \mathbb{F}_q \)
\texttt{t_COMPLEXP}. Complex Numbers
\texttt{t_PADIC}. p-adic Numbers
\texttt{t_QUAD}. Quadratic Numbers
\texttt{t_POLMOD}. Polynomials modulo g
\texttt{t_POL}. Polynomials
\texttt{t_SEM}. Power Series
\texttt{t_PFRAC}. Rational Functions
\texttt{t_QF}/\texttt{t_QFR}. Imag/Real binary quad. form
\texttt{t_VEC}/\texttt{t_COL}. Row/Column Vectors
\texttt{t_VEC} integer range

Reserved Variable Names
\( \pi = 3.14 \ldots, \gamma = 0.57 \ldots, C = 0.91 \ldots \)
square root of \(-1\)
Landau's big-oh notation

Information about an Object
\( \text{PARI type of object x} \)
\( \text{length of x / size of x in memory} \)
\( \text{real precision / bit precision of x} \)
p-adic, series prec. of x

Operators
\text{basic operations}
\text{comparisons}
\text{boolean operators (or, and, not)}
\text{bit operations}
\text{derivative of f}
\text{differential operator}
\text{quote operator (formal variable)}
\text{assignment}

Select Components
\text{n-th component of x}
\text{n-th component of vector/list x}
\text{n-th component of multivariate polynomial in \( x \)}
\text{row m or column n of matrix x}
\text{numerator/denominator of fraction x}

Random Numbers
\text{random integer in } [0, N]
\text{set random seed}

Conversions
\text{to vector, matrix, vec. of small ints}
\text{to list, set, map, string}
\text{create PARI object (x mod y)}
\text{make a polynomial of degree 2}
\text{convert x to simplest possible type}
\text{object x with real precision n}
\text{object x with bit precision n}
\text{set precision to } p \text{ digits in dynamic scope}
\text{set precision to } p \text{ bits in dynamic scope}

Conjugates and Lifts
\text{conjugate of a number x}
\text{norm of x, product with conjugate}
\text{L^p norm of x} (\text{if p no power})
\text{square of L^\infty norm of x}
\text{lift of x from Mods and p-adics}
\text{recursive lift}
\text{lift all \text{t_INT} and \text{t_PADIC} → \text{t_INFINITY}}
\text{lift all \text{t_POLMOD} → \text{t_POL}}

Lists, Sets & Maps
\text{sets (= row vector with strictly increasing entries w.r.t. cmp)}
\text{intersection of sets x and y}
\text{set of elements in x not belonging to y}
\text{union of sets x and y}
\text{does y belong to the set x}
\text{set of all \( f(x,y) \), } x \in X, y \in Y
\text{is x a set?}
\text{lists. create empty list: } L = \text{List()}
\text{append x to list L}
\text{remove n-th component from list L}
\text{insert x in list L at position n}
\text{sort the list L in place}
\text{Maps. create empty dictionary: } M = \text{Map()}
\text{attach value to key k}
\text{recover value attached to key k or error}
\text{is key k in the dict ? (set v to \text{Map()})}
\text{remove k from map domain}

GP Programming

User functions and closures
\( x, y \) are formal parameters; \( y \) defaults to \( \Pi \) if parameter oopted;
\( x, t \) are local variables (lexical scope), \text{initialized to 1.}

\text{fun(x, y=\Pi) = my(z=1, t);}
\text{fun(x, y=Pi) → my(z=1; t);}
\text{seq a set ?}
\text{attach a help message to } f
\text{undeline symbol x (also kills help)}

Control Statements
\text{X: formal parameter in expression seq}
\text{if a ≠ 0, evaluate seq 1, else seq2}
\text{eval. seq for } a < X \leq b
\text{for primes } a < X \leq b
\text{for primes } a \equiv (mod q)
\text{for composites } a < X \leq b
\text{for } a < X \leq b \text{ stepping s}
\text{for } X \text{ dividing } n
\text{X = [n, factor(n)], a < n \leq b}
\text{as above, n squarefree}
\text{X = [d, factor(d)], d | n}
\text{multivariable for, lex ordering}
\text{loop over partitions of n}
\text{permutations of S}
\text{subsets of } \{1, \ldots, n\}
\text{``k-subsets of } \{1, \ldots, n\}
\text{vectors } v, q(v) \leq B, q(v) > 0
\text{H \leq F \text{ finite abelian group}}
\text{evaluate seq until a < 0}
\text{while a < 0, evaluate seq}
\text{exit n innermost enclosing loops}
\text{start new iteration of n-th enclosing loop}
\text{set breakout of } n \text{-th enclosing loop}

\text{setbreak(x, y)}
\text{gcd(x,y)}
\text{norm(x)}
\text{numlpx(x, p)}
\text{norml2(x)}
\text{liften, centerlift(x, y)}
\text{liftall}
\text{liftftol}

listput(L, x, \{\})
listpop(L, \{\})
listinsert(L, x, \{\})
listsort(L, \{\})
Exceptions, warnings: raise an exception / warn type of error message E
try seq., evaluate seq. on error

Functions with closure arguments / results:
select(f, v) apply(f, to all entries in v)
evaluate f(a1, ..., a3) evaluate f{(f(a2, a2), a3) | a2, a3}
calling function as closure

Sums & Products:
sum x = a to b, initialized at x sum entries of vector x product of all vector entries
sum expr over divisors of n ... assuming expr multiplicative product a ≤ b, initialized at x product over primes a ≤ b ≤ c

Sorting:
sort x by kth component
min. m of x (m = i | max. does y belong to x, sorted wrt. f.

Input/Output:
print with/without ln, TeX format pretty print matrix print fields with separator formatted printing write args to file
write x in binary format read file into GP ...

Files and file descriptors:
File descriptors allow efficient small consecutive reads or writes from or to a given file. The argument n below is always a descriptor, attached to a file in r(ead), w(e)rite or a(pend) mode. get descriptor n for file path in given mode fileopen(path, mode)
... from shell cmd output (pipe)
close descriptor
commit pending write operations
read logical line from file...
write stream to file

Timers:
CPU time in ms and reset timer CPU time in ms since gp startup time in ms since UNIX Epoch timeout command after s seconds

Interface with system:
allocates a new stack of s bytes alias old to new install function from library execute system command a ...
and feed result to GP ...
returning GP string

error(), warning() errorname(E)
iferr(seq, E, seq)

Parallel evaluation:
These functions evaluate their arguments in parallel (threads or MPI); args. must not access global variables and must be free of side effects. Enabled if threading engine is not single in gp header.
evaluate f on [x1, ..., xn] evaluate closures f[, f[, f]]

as select as sum
(parity select, A, {flag}) parsum(i = a, b, expr{x})
eval f for i = a, ..., b 
(parfor[i = a, b], f, {r}, {j}) 
... for p prime in [a, b] 
(parforprime(p = a, b, f, {r}, {j})
... for v in [a, b]
(parforvec(x = a, b, expr{f}, {r}, {j})

decide x as inline (allows to use as global)

Linear Algebra:
dimensions of matrix x

... assume is square and diagonal... 
adjoint of the matrix
basis_vectors/values of matrix x

characteristic/minimal polynomial of x

determine of matrix x

diagonal form of x

QR decomposition

apply matqrs transform to v

Constructors & Special Matrices:

[g(x); x ∈ v s.t. f(x)] [g(x); x ∈ v s.t. f(x)]

row vec. of expr eval'd at 1 ≤ n

col. vec. of expr eval'd at 1 ≤ n

diagonal matrix with diagonal x

is x diagonal?

-polynomials & Rational Functions:

all defined polynomial variables
get var. of highest priority (higher than v)
... of lowest priority (lower than v)

Polynomials & Rational Functions:

all defined polynomial variables
get var. of highest priority (higher than v)
... of lowest priority (lower than v)

Copyright and this permission notice are preserved on all copies. Copyright and this permission notice are preserved on all copies.
Finite Fields

A finite field is encoded by any element \( (t, \text{FFLT}) \).
find irreducible \( T \in F_q[x] \), \( \deg T = n \)
Create \( t \in F_q \geq F_q[t]/(T) \)
\( \ldots \) indirectly, with implicit \( T \)
map \( m = F_q^{\geq a} \rightarrow F_q^{\geq b} \)
build \( K = F_q[x]/(P) \) extending \( F_q^{\geq a} \).
\( \ldots \) evaluate map \( m \) on \( x \)
\( \ldots \) inverse map of \( m \)
\( \ldots \) compose maps \( m \circ n \)
\( \ldots \) \( F^n \) over \( F_q \geq a \)
\# (monic irred. \( T \in F_q[x], \deg T = n \))

Formal & \( p \)-adic Series

truncate power series or \( p \)-adic number
valuation of \( x \) at \( p \)

Dirichlet and Power Series

Taylor expansion around \( 0 \) of \( f(x) \).
Laurent series expansion around \( 0 \) up to \( x^k \)
\( \sum a_k \cdot x^k \) from \( \sum a_k \cdot x^k \) \( b_k \cdot x^k \)
reverse power series \( F \) so \( F(x) = z \)
removes terms of degree \( < n \) in \( f \)
Dirichlet series multiplication / division
Dirichlet Euler product (\( b \) terms)

Transcendental and \( p \)-adic Functions

real, imaginary part of \( x \)
absolute value, argument of \( x \)
| square root of \( x \) | trig functions
\( \sin \), \( \cos \), \( \tan \), \( \cotan \)
\( \sec \), \( \csc \), \( \cosh \), \( \sinh \)
\( \tan \), \( \cot \), \( \csc \), \( \sec \)

Iterations, Sums & Products

Numerical integration for meromorphic functions
Behaviour at endpoint for Double Exponential (DE) methods: either a scalar \((a \in \mathbb{C}, \text{regular}) \) or \( \pm \infty \) (decreasing at least as \( x^{-2} \))
\( (x-a)^{-\gamma} \) singularity
\( [a, a] \)
exponential decrease \( e^{-a|x|} \)
\( [\pm \infty, a], a > 0 \)
slow decrease \( |x|^\alpha \)
\( \ldots a < -1 \)
oscillating as \( \cos(kx) \)
oscillating as \( \sin(kx) \)
numerical integration
\( \int_0^\infty \frac{1}{x} \) on circle \([z-a] = R \) \( \int_{|x| = R} (a, R, f(T)) \)

Other integration methods

\( \pm \infty \text{Gauss-Legendre} \)
\( \pm \infty \text{Legendre} \)
\( \pm \infty \text{Gauss-Legendre} \)
\( \pm \infty \text{Romberg} \) (low accuracy)

Numerical summation

sum of series \( f(n) \), \( n \geq a \) (low accuracy)
\( \sum \) of alternating/positive series
\( \sum \) of series using Euler-Maclaurin
\( \sum \) of series using Dirichlet
\( \sum \) of series using Abel-Plane
\( \sum \) of series using Lagrange
\( \sum \) of series using Lagrangian

Products

\( \prod \) product over \( a \leq x < b \), initialized at \( x \)
\( \prod \) product over \( a < x \leq b \)
\( \prod \) product over \( a < x \leq b \)
\( \prod \) product over \( a < 0 \)

Other numerical methods

real root of \( f(x) \) in \([a, b]\), bracketed root
\( \ldots \) by interval splitting
\( \lfloor f(t) \rfloor, t \to \infty \)
limit of \( f(t) \), \( t \to \infty \)
\( a \) asymptotic expansion of \( f \) at \( x = 0 \)
numerical derivation \( \text{w.r.t. } x \) \( x^4(a) \)
\( x^4(a) \)
evaluate continued fraction \( F \) at \( \text{confrac}
\) \( 1 \}
\( \text{power series to cont. fraction (L terms) } \)
\( \text{padé approximant (deg, denom.} \leq B \)
\( \text{bestappPade(S, B)} \)
Elementary Arithmetic Functions

- `binary(x)`
- `bittest(x, n)`
- `hammingweight(x)`
- `hammingweight(x)`
- `digits(x, {B = 10})`
- `sundigits(x, {B = 10})`
- `fromdigits(v, {B = 10})`
- `ceil(x)`
- `floor(x)`
- `fract(round(x, {k}))`
- `truncate(x, {k})`

Primes and Factorization

- `gcdLCM(x and y)`
- `gcd of entries of a vector/matrix`
- `Chebyshev(n, ≤, t)`
- `factorization of x`
- `primecertexport(x)`
- `convert certificate to Magma/PRIMO`
- `N∤d`
- `divisors of N`
- `M"obius(f, x)`
- `sum of (x)`
- `Ramanujan's τ(n)`
- `multiplicative order of x`
- `discrete logarithm of x`
- `primitive root mod q`
- `Kronecker symbol (x, g, n)`
- `High-resolution plot (immediate plot)`
- `plot high-resolution graph`