Asymptote Reference Card

Program structure/functions

import "filename"  
import "filename" as name  
include "filename"  


function definition


Data types/declarations

boolean (true or false)  
tristate boolean (true, default, or false)  
integer  
float (double precision)  
ordered pair (complex number)  
character string  
fixed piecewise cubic Bezier spline  
unresolved piecewise cubic Bezier spline  
color, line type/width/cap, font, fill rule  
label with position, alignment, pen attributes  
drawing canvas  
affine transform  
constant (unchanging) value  
allocate in higher scope  
no value  
inhibit implicit argument casting  
structure  
create name by data type  

3D data types (import three;)

ordered triple  
3D path  
3D guide  
3D affine transform  

Constants

exponential form  
TEX string constant  
TEX strings: special characters  
C strings: constant  
C strings: special characters  
C strings: newline, cr, tab, backspace  
C strings: octal, hexadecimal bytes  

Operators

arithmetic operations  
modulo (remainder)  
comparisons  
not and or (conditional evaluation of RHS)  
and or xor  
cast expression to type  
increment decrement prefix operators  
assignment operators  
conditional expression  
structure member operator  
expression evaluation separator  

Flow control

statement terminator  
block delimiters  
comment delimiters  
exit from while/do/for  
next iteration of while/do/for  
return value from function  
terminate execution  
abort execution with error message  

Flow constructions (if/while/for/do)

if(expr) statement  
else if(expr) statement  
else statement  
while(expr) statement  
for(expr1; expr2; expr3) statement  
do statement  
while(expr);
**Arrays**

array
array element i
array indexed by elements of int array A
anonymous array
array containing n deep copies of x
length
cyclic flag
pop element x
push element x
append array a
insert rest arguments at index i
delete element at index i
delete elements with indices in [i,j]
delete all elements
test whether element n is initialized
array of indices of initialized elements
complement of int array in (0,...,n-1)
deep copy of array a
array (0,1,...,n-1)
array (n,n+1,...,m)
array (n-1,n-2,...,0)
array (f(0),f(1),...,f(n-1))
array obtained by applying f to array a
uniform partition of [a,b] into n intervals
concat specified 1D arrays
return sorted array
return array sorted using ordering less
search sorted array a for key
index of first true value of bool array a
index of nth true value of bool array a

**Initialization**

initialize variable
initialize array

**path connectors**

straight segment
Bezier segment with implicit control points
concatenate
lift pen
..tension atleast 1...
..tension atleast infinity..

**Labels**

implicit cast of string s to Label
Label s with relative position and alignment
Label s with specified pen

draw commands

draw path with current pen
draw path with pen
draw labeled path
draw arrow with pen
draw path on picture
draw visible portion of line through two pairs

**fill commands**

fill path with current pen
fill path with pen
fill path on picture

**label commands**

label a pair with optional alignment z
label a path with optional alignment z
add label to picture

**clip commands**

clip to path
clip to path with fill rule
clip picture to path

**pens**

Grayscale pen from value in [0,1]
RGB pen from values in [0,1]
CMYK pen from values in [0,1]
RGB pen from hexadecimal string
hexadecimal string from rgb pen
hex pen from values in [0,1]
invisible pen
default pen
current pen
solid pen
dotted pen
wide dotted current pen
wide dotted pen
dashed pen
long dashed pen
dash dotted pen
long dash dotted pen
PostScript butt line cap
PostScript round line cap
PostScript projecting square line cap
miter join
round join
even-odd fill rule
zero-winding fill rule
join to character bounding box (default)
align to TeX baseline
pen with font size (pt)
LaTeX pen from encoding,family,series,shape
TeX pen
scaled TeX pen
PostScript font from strings
pen with opacity in [0,1]
construct pen nib from polygonal path
pen mixing operator

**type**

type [] name;
name[1]
name[A]
new type[1:dim]
array(n,x)
name.length
name.cyclic
name.pop()
name.push(x)
name.append(a)
name.insert(i,...)
name.delete(i)
name.delete(i,j)
name.delete()
name.initialize(n)
name.keys
complement(a,n)
copy(a)
sequence(n)
sequence(n,m)
reverse(n)
sequence(f,n)
map(f,a)
uniform(a,b,n)
concat(a,b,...)
sort(a)
sort(a,less)
search(a,Keys)
find(a)
find(a,n)

**fill commands**

fill(path)
fill(path,pen)
fill(picture,path)

**label commands**

label(Label,pair,z)
label(Label,path,z)
label(Label,picture,Label)

**clip commands**

clip(path)
clip(path,pen)
clip(picture,path)

**pens**

gray(g)
RGB(r,g,b)
CMYK(r,g,b)
RGB(string)
hex(pen)
hsv(h,s,v)
invisible
defaultpen
currentpen
solid
dotted
dotted
wide dotted pen
long dash dotted pen
dashed
long dashed
dash dotted
long dash dotted
squarecap
roundcap
extendcap
miterjoint
roundjoin
beveljoint
miterlimit(real)
zerowinding
evenodd
basetalign
fontsize(real)
font(strings)
font(string)
font(string,real)
Courier(series,shape)
opacity(real)
makepen(path)
+
path operations

number of segments in path p
number of nodes in path p
is path p cyclic?
is segment i of path p straight?
coordinates of path p at time t
direction of path p at time t
direction of path p at length(p)
unit(dir(p)+dir(q))
acceleration of path p at time t
radius of curvature of path p at time t
precontrol point of path p at time t
postcontrol point of path p at time t
arclength of path p
time at which arclength(p)=L
point on path p at arclength L
first value t at which dir(p,t)=z
time t at relative fraction i of arclength(p)
point at relative fraction i of arclength(p)
point midway along arclength of p
path running backwards along p
subpath of p between times a and b
times for one intersection of paths p and q
points at which p reaches minimal extents
times at which p reaches maximal extents
intersection times of paths p and q
intersection times of path p with "--a--b--"
intersection times of path p crossing x =z
intersection times of path p crossing y =z,y
intersection point of paths p and q
intersection points of p and q
intersection of extension of P--Q and p--q
lower left point of bounding box of path p
upper right point of bounding box of path p
subpaths of p split by nth cut of knife
winding number of path p about pair z
pair z lies within path p?
pair z lies within or on path p?
path surrounding region bounded by paths
path filled by draw(g,p)
unit square with lower-left vertex at origin
circle of radius r about c
arc of radius r about c from angle a to b
unit n-sided polygon
unit n-point cyclic cross

pictures

add picture pic to currentpicture
add picture pic about pair z

length(p)
size(p)
cyclic(p)
straight(p,i)
piecewisestraight(p)
point(p,t)
dir(p,t)
dir(p)
accel(p,t)
radius(p,t)
precontrol(p,t)
postcontrol(p,t)
arclength(p)
arctime(p,L)
arcinterval(p,L)
arcpoint(p,L)
relpoint(p,l)
reltime(p,l)
reltime(p)
interpoint(p,q)
interinterval(p,q)
interpoint(p)
time(p)
pretime(p,t)
posttime(p,t)
times(p)
maxtimes(p)
mintimes(p)
maxtimes(p)
mintimes(p)
intersections(p,q)
times(p,a,b)
times(p,x)
times(p,z)
intersections(p,a,b)
intersections(p,a,b)
times(p,x)
times(p,z)
times(p,z)
times(p,0)
times(p,1)
times(p,t)
times(p,t)
times(p,t)
interpoint(p,q)
interpoints(p,q)
interpoint(p,q)
interpoints(p,q)
interpoint(p,q)
interpoints(p,q)
interpoint(p,q)
interpoints(p,q)

string operations

concatenate operator
string length
position \geq pos of first occurrence of t in s
position \leq pos of last occurrence of t in s
string with t inserted in s at pos
string s with n characters at pos erased
substring of string s of length n at pos
string s reversed
string s with before changed to after
string s translated via \((before,after)\)
string s in format using \(format\)
current time formatted by format
positions in seconds of string t using format
string corresponding to seconds using format
split s into strings separated by delimiter

affine transforms

identity transform
shift by values
shift by pair
scale by x in the x direction
scale by y in the y direction
scale by x in both directions
scale by real values x and y
map \((x,y)\) \rightarrow \((x+ny, y)\)
rotate by real angle in degrees about pair z
reflect about line from P--Q

May 2014 v1.1. Copyright © 2014 John C. Bowman

Permission is granted to make and distribute copies of this card, with or
without modifications, provided the copyright notice and this permission
notice are preserved on all copies.