Asymptote Reference Card

Program structure/functions

import "filename"
import module
import "filename" as name
include "filename"
include verbatim text from file
type f(type,...); optional function declaration
type name; variable declaration
type f(type arg,...) { function definition
  statements
  return value;
}

Data types/declarations

boolean (true or false) bool
tri-state boolean (true, default, or false) bool3
integer int
float (double precision) real
ordered pair (complex number) pair
character string string
fixed piecewise cubic Bezier spline path
unresolved piecewise cubic Bezier spline guide
color, line type/width/cap, font, fill rule pen
label with position, alignment, pen attributes Label
drawing canvas picture
affine transform transform
constant (unchanging) value const
allocate in higher scope static
no value void
inhibit implicit argument casting explicit
structure typedef type name

3D data types (import three)

ordered triple triple
3D path path3
3D guide guide3
3D affine transform transform3

Constants

exponential form 6.02e23
"abc...de"
\", \"
'abc...de'
\", \" \'
\n \r \t \b
\0-\377 \x0-\xFF

Operators

arithmetic operations + - * /
modulo (remainder) %
comparisons == != > >= < <=
and or (conditional evaluation of RHS) !
& & | (type) expr & | -
++ -- += -= *= /= %=
expr1 ? expr2 : expr3
name.member

Flow control

statement terminator ;
block delimiters {
comment delimiters /* */
comment to end of line delimiter //
exit from while/do/for break;
continue;
return expr;
exit();
abort(string);

Flow constructions (if/while/for/do)

if(expr) statement
  else if(expr) statement
  else statement
while(expr) statement
for(expr1; expr2; expr3) statement
for(type var : array) statement
  do statement
  while(expr);

1
**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 [1, 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(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 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 at least 1...
.. tension at least infinity...

**Labels**

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

type [] name;
name[i];
name[A];
new type [dim]
array(n,x);
namex;
namedim;
name.length;
namedim.length;
name.pop();
namedim.append(a);
namedim.insert(i,...);
namedim.delete(i);
namedim.delete(1,j);
namedim.delete();
namedim.initialized(n);
namedim.keys;
complement(a,n);
copy(a);
sequence(a);
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,key);
find(a);
find(a,n);
find(a,n);

**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 heximdecimal string
heximdecimal string from rgb pen
hsv 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
bevel join
pen with miter limit
zero-winding fill rule
even-odd fill rule
align to character bounding box (default)
align to \TeXx\ baseline
pen with font size (pt)
\TeXx\ pen from encoding,family,series,shape
\TeXx\ pen
scaled \TeXx\ pen
PostScript font from strings
font (strings)
font (string, real)
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 segment i of path p straight?
is path p cyclic?
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
first value t at which dir(p,t)=z
time t at relative fraction l of arclength(p)
point at relative fraction l 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
times at which p reaches minimal extents	imes at which p reaches maximal extents
intersection times of paths p and q
intersection times of path p with "x=a--b--x"
intersection times of path p crossing x=x
intersection times of path p crossing y=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
pair of points 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
unit circle centered 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 p

length(p)
size(p)
cyclic(p)
straight(p,i)
piecewisestraight(p)
point(p,t)
dir(p,t)
dir(p,q)
accel(p,q)
radius(p,t)
precontrol(p,t)
postcontrol(p,t)
arcpath(p,t)
arcpath(p,L)
ardpoint(p,L)
arcpoint(p,L)
arctime(p,L)
arclength(p)
postcontrol(p,t)
precontrol(p,t)
postcontrol(p,t)
precontrol(p,t)
postcontrol(p,t)
precontrol(p,t)
postcontrol(p,t)

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) → (x+ay,y)
rotate by real angle in degrees about pair p
reflect about line from P--Q

string operations

concatenate operator
string length
position ≥ pos of first occurrence of t in s
position ≤ pos of last occurrence of t in s
string with t inserted in s at pos
substring of string s of length n at pos
string s reversed
string s with before changed to after
string s translated via \{\text{before,after}\},…
format x using C-style format string s
casts hexadecimal string to an integer
casts x to string using precision digits
current time formatted by format
time in seconds of string t using format
string corresponding to seconds using format
split s into strings separated by delimiter

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.