C Reference Card (ANSI)

Input/Output <stdio.h>

Standard I/O
standard input stream stdin
standard output stream stdout
standard error stream stderr
end of file EOF
get a character getchar()
pin a character putchar(ch)
print formatted data print("format", arg1, . . .)
print to string sprintf(s, "format", arg1, . . .)
read formatted data scanf("format", &arg1, . . .)
read from string sscanf(s, "format", &arg1, . . .)
read line to string s (< max chars) gets(s, max)
print string s puts(s)

File I/O
declare file pointer FILE *fp
name to pointed file fopen("name", "mode")
get a character fgets(fp)
write file to file fprintf(fp, "format", arg1, . . .)
read from file fscanf(fp, "format", arg1, . . .)
close file fclose(fp)
non-zero if error ferror(fp)
non-zero if EOF feof(fp)
read line to string s (< max chars) fgets(s, max, fp)
write string s fputs(s, fp)

Codes for Formatted I/O: "i" = int, "f" = float

- left justify
* print with sign
space print space if no sign
0 pad with leading zero
w min field width
p precision
m conversion character:
h short, l long, L long double
c conversion character:
d, i, long i, unsigned
s single char, S char string
f double, e, exponential
o octal, x, X hexadecimal
p pointer, n number of chars written
s, g same as f or e, depending on exponent

Variable Argument Lists <stdarg.h>
declaration of pointer to arguments va_list arg
initialization of argument pointer va_start(arg, lastarg)
lastarg is last named parameter of the function
access next unnamed arg, update pointer va_arg(arg, type)
call before exiting function va_end(arg)

Standard Utility Functions <stdlib.h>

- absolute value of int n abs(n)
- absolute value of long n labs(n)
- quotient and remainder of int dividend, int divisor quotient, remainder div(t) = quotient * divisor + remainder div(t, rem)
- quotient and remainder of long dividend, long divisor quotient, remainder div(t) = quotient * divisor + remainder div(t, rem)
- pseudo-random integer [0, RAND_MAX) rand() = rand()
- terminate program execution exit(status)
- pass string s to system for execution system(s)

Conv's and S's
- convert string a to double atof(s)
- convert string a to integer atoi(s)
- convert string a to long atol(s)
- reallocate prefix of type double realloc(size)
- prefix of size (base b) to long strl60(size, b)
- same, but unsigned long strl60(size, b)

Storage Allocation
- allocate storage malloc(size)
- deallocate space free(ptr)

Array Functions
- search array for key bsearch(key, array, n, size, cmp)
- sort array ascending order qsort(array, n, size, cmp)

Time and Date Functions <time.h>
- processor time used by program clock()
- current calendar time time() = time(time_t)
- time, time_t in seconds [double] diff time(time_t)
- arithmetic types representing times clock_t, time_t
- structure for calendar time comp tm
  - months after minute
  - minutes after hour
  - hour since midnight
  - day of month
  - months since January
  - years since 1900
  - days since Sunday
  - days since January 1
  - Daylight Savings Time flag

- convert local time to calendar time atmtime(tp)
- convert time in tp to string asctime(tp)
- convert calendar time to time tp
- convert calendar time to GMT gmtime(tp)
- convert calendar time to local time localtime(tp)
- format date and time strftime("format", tm, tp)
- tp is a pointer to a structure of type tm

Mathematical Functions <math.h>

Arguments and returned values are double
- trig functions sin(x), cos(x), tan(x)
- inverse trig functions asin(x), acos(x), atan(x)
- hyperbolic functions sinh(x), cosh(x), tanh(x)
- hyperbolic functions exp(x), log(x), log10(x)
- exponential & log functions expm1(x), log1p(x), lgamma(x)
- exponential & log (2 power) modf(x, &yp), modf(x, y)
- power functions pow(x, y), sqrt(x)

- rounding functions ceil(x), floor(x), fab(x)

- Integer Type Limits <limits.h>

- The numbers given in parentheses are typical values for the constants on a 32-bit Unix system,
- CHAR_BIT bits in char (8)
- CHAR_MIN max value of char (-128 or 0)
- INT_MIN max value of int (-2,147,483,647)
- ULONG_MAX max value of long (-2,147,483,647)
- CHAR_MIN max value of signed char (+127)
- CHAR_MIN max value of signed char (-128)
- INT_MIN max value of signed int (-32,767)
- SHORT_MIN max value of short (-32,767)
- INT_MIN max value of signed int (-2,147,483,647)

- Float Type Limits <float.h>

- FLT_RADIX radix of exponent rep (2)
- FLT_DIG decimal digits of precision (6)
- FLT_EPSILON smallest x 1.0 + x 1.0 (10-2)
- FLT_MIN minimum floating point number (10-9)
- FLT_MAX maximum exponent (1023)
- FLT_MIN minimum exponent (10-9)
- DBL_DIG decimal digits of mantissa (10)
- DBL_MIN smallest x 1.0 + x 1.0 (10-9)
- DBL_MAX maximum exponent (1023)
- DBL_MIN min double floating point number (10-9)
- DBL_MAX maximum exponent (1023)

May 1990 v1.3, Copyright © 1989 Joseph H. Silverman
Permission is granted to make and distribute copies of this card provided the copyright notice and this permission notice are preserved on all copies.

Send comments and corrections to J.H. Silverman, Math Dept., Brown University, Providence, RI 02912 USA. (jhs@math.brown.edu)