C programming language has 25 standard header files which are as follows:

1. **#include<stdio.h>** (Standard input-output header)  
   Used to perform input and output operations in C like scanf() and printf().
2. **#include<string.h>** (String header)  
   Perform string manipulation operations like strlen and strcpy.
3. **#include<conio.h>** (Console input-output header)  
   Perform console input and console output operations like clrscr() to clear the screen and getch() to get the character from the keyboard.
4. **#include<stdlib.h>** (Standard library header)  
   Perform standard utility functions like dynamic memory allocation, using functions such as malloc() and calloc().
5. **#include<math.h>** (Math header )  
   Perform mathematical operations like sqrt() and pow(). To obtain the square root and the power of a number respectively.
6. **#include<ctype.h>**(Character type header)  
   Perform character type functions like isaplha() and isdigit(). To find whether the given character is an alphabet or a digit respectively.
7. **#include<time.h>**(Time header)  
   Perform functions related to date and time like setdate() and getdate(). To modify the system date and get the CPU time respectively.
8. **#include<assert.h>** (Assertion header)  
   It is used in program assertion functions like assert(). To get an integer data type in C/C++ as a parameter which prints stderr only if the parameter passed is 0.
9. **#include<locale.h>** (Localization header)  
   Perform localization functions like setlocale() and localeconv(). To set locale and get locale conventions respectively.
10. **#include<signal.h>** (Signal header)  
    Perform signal handling functions like signal() and raise(). To install signal handler and to raise the signal in the program respectively
11. **#include<setjmp.h>** (Jump header) - Perform jump functions.
12. **include<stdarg.h>** (Standard argument header)  
    Perform standard argument functions like va\_start and va\_arg(). To indicate start of the variable-length argument list and to fetch the arguments from the variable-length argument list in the program respectively.
13. **#include<errno.h>** (Error handling header)  
    Used to perform error handling operations like errno(). To indicate errors in the program by initially assigning the value of this function to 0 and then later changing it to indicate errors.
14. **<complex.h>** - Complex number arithmetic
15. **<fenv.h>** - Floating-point environment
16. **<float.h>** - Limits of floating-point types
17. **<inttypes.h>** - Format conversion of integer types
18. **<iso646.h>** - Alternative operator spellings
19. **<limits.h>**- Ranges of integer types
20. **<setjmp.h>**- Nonlocal jumps
21. **<signal.h>** - Signal handling
22. **<stdalign.h>** - alignas and alignof convenience macros
23. **<uchar.h>** - UTF-16 and UTF-32 character utilities
24. **<wchar.h>** - Extended multibyte and wide character utilities
25. **<wctype.h>** - Functions to determine the type contained in wide character data
26. **Important header files and it's functions discussed**
27. **#include<stdio.h>**

This header file contains functions and macros to perform standard i/o operations. When we include the header file has some standard functions that are listed below.

The stdio.h header defines three variable types, several macros, and various functions for performing input and output.

* **printf()** -*Description* -The C library function sends formatted output to stdout, *Declaration* -int printf(const char \*format, ...);
* **scanf()** -*Description* - The C library function reads formatted input, *Declaration* - int scanf(const char \*format, ...);
* **fgetc()** - The function fgetc() is used to read the character from the file. It returns the character pointed by file pointer
* **fputc()** - This function is used to put/display a character into the screen.
* **fgets()** - This function is used to get the string input form the user.
* *fgets(string, sizeof(string) , stdin)*
* **getc()** -The getc function in C reads the next character from any input stream and returns an integer value.
* **putc()** -The putc function in C writes a character to any output stream and returns an integer value
* **vprintf()** -The function works in a similar way to printf(). However, vprintf() uses elements in the variable argument list to replace format specifiers rather than using additional arguments
* **getchar()** -getchar is a function in C programming language that reads a single character from the standard input stream stdin

1. **FILE HANDLING FUNCTION IN STDIO HEADER FILE**

**rewind()**  
the rewind function sets the file position to the beginning of the file for the stream pointed to by stream. It also clears the error and end-of-file indicators for stream

**ferror**  
The ferror() function is used to check for the file error on given stream. return value of zero indicates no error has occurred, and a non-zero value indicates an error occurred.  
**fopen()**  
The fopen() method in C is a library function that is used to open a file to perform various operations which include reading, writing  
**fclose()**  
fclose() function is C library function and it's used to releases the memory stream, opened by fopen() function

1. **string.h**

This header file declares functions to manipulate strings and memory manipulation routines.The function contained in the header file string.h are listed below.

**strlen()**  
This function returns the lenght of the string , the number of characters present in the string excluding the null character.  
variable = strlen(string);  
*a string of length 0 is called null string*

#include <stdio.h>

#include <string.h>

int main()

{

char a[20]="Program";

char b[20]={'P','r','o','g','r','a','m','\0'};

printf("Length of string a = %d \n",strlen(a));

printf("Length of string b = %d \n",strlen(b));

return 0;

}

C

Copy

1. **strcat()**

This function is used to concatenate two strings. The process of combining two strings to form a string is called as Concatenation.  
strcat(string 1 , string2);  
*joins string2 to the end of string1.*

#include <stdio.h>

#include <string.h>

int main() {

char str1[100] = "This is ", str2[] = "opengenus.org";

// concatenates str1 and str2

// the resultant string is stored in str1.

strcat(str1, str2);

puts(str1);

puts(str2);

return 0;

}

C

Copy

1. **strcpy()**

A string cannot be copied to another string by using assignment operator. The function strcpy() is used to copy.  
strcpy(string1, stirng2);  
*it copies all characters of string2 to string1*

#include <stdio.h>

#include <string.h>

int main() {

char str1[20] = "C programming";

char str2[20];

// copying str1 to str2

strcpy(str2, str1);

puts(str2); // C programming

return 0;

}

C

Copy

1. **strcmp()**

This function is used to alphabetically compare a string with another string. this function is case - sensitive (it treats uppercases and lowercases as different).  
strcmp(string1,string2);  
return 1: if string1 > string2  
return 0: if string1 = string2  
return -1:if string1 < string2

#include <stdio.h>

#include <string.h>

int main() {

char str1[] = "abcd", str2[] = "abCd", str3[] = "abcd";

int result;

// comparing strings str1 and str2

result = strcmp(str1, str2);

printf("strcmp(str1, str2) = %d\n", result);

// comparing strings str1 and str3

result = strcmp(str1, str3);

printf("strcmp(str1, str3) = %d\n", result);

return 0;

}

C

Copy

1. **stdlib.h**

This header file is used to declare conversion routines , search . sort routines.  
the functions are listed below:

* \_fullpath \* \_makepath \* \_searchenv \* \_splitpath \* atoi
* atof \* fcvt \* gcvt \* itoa \* ltoa \* abs
* malloc \* realloc \* free \* coreleft  
  check out <https://iq.opengenus.org/stdlib-in-c/>

1. **math.h**

This header file declares prototypes for mathematical functions and error handlers. the functions that are used to perform mathematical calculations and conversions are listed below.

* **acos()** - computes arc cosine
* **acosh** - computes arc hyperbolic cosine
* **asin()** -computes arc sine
* **asinh()** -computes the hyperbolic of arc sine of an argument
* **atan()** -computes the arc tangent of an argument
* **atan2()** -computes the arc tangent of an argument.
* **atanh()** -computes arc hyperbolic tangent
* **cbrt()** -computes cube root of a number
* **ceil()** -computes the nearest integer greater than argument
* **cos()** -computes the cosine of an argument.
* **cosh()** -computer hyperbolic cosine.
* **exp()** -computes the exponential raised to the argument
* **fabs()** -computes absolute value
* **floor()** -calculates the nearest integer less than argument
* **hypot()** -computes hypotenuse
* **log()** -computes natural logarithm of an argument.
* **log10()** -computes the base 10 logarithm of an argument.
* **pow()** -Computes power of a number
* **sin()** -compute sine of a number
* **sinh()** -computes the hyperbolic sine of an argument.
* **sqrt()** -computes square root of a number
* **tan()** -computes tangent
* **tanh()**- computes the hyperbolic tangent of an argument

Check out for details - <https://iq.opengenus.org/math-h-in-c/>

1. **ctype.h**

The C <ctype.h> header file declares a set of functions to classify (and transform) individual characters. For example, isupper() checks whether a character is uppercase or not.

* **isalnum()** -checks alphanumeric character
* **isalpha()** -checks whether a character is an alphabet or not
* **iscntrl()** -checks control character
* **isdigit()** -checks numeric character
* **isgraph()**- checks graphic character
* **islower()**- checks lowercase alphabet
* **isprint()**- checks printable character
* **ispunct()**- checks punctuation
* **isspace()**-check white-space character
* **isupper()**- checks uppercase alphabet
* **isxdigit()** -checks hexadecimal digit character
* **tolower()** -converts alphabet to lowercase
* **toupper()** -converts to uppercase alphabet

1. **Summary**

*In this article at OpenGenus, you have a basic clarity on different header files in c programming language.  
You will be able to use and apply the knowledge of function into the header files.  
Applications of each of the header files.  
Brief on the standard C header files like stdio.h , sting.h, stdlib.h , ctype.h , math.h.  
You can have a idea on all of the standard header files that c programming language offers.*