

General Knowledge

Syntax in C All Statements in C must end with a semicolon ; keywords and other code elements are case-sensitive

Escape Sequences In C, within a string the \n will cut the string and move the part after to the next line. The \t will tab the string after, giving it more space

Single Line Comments //This is a single line comment within c

Block Comments */* Block comments are nice for when you have a lot to say about a particular piece of code */*

Compiling C Code to compile c code, in a terminal you'll need to type in, minimally, gcc fileName.c this will output something call a.out which will allow you to run the program
Alternatively, if you do gcc fileName.c -o fileName you can then run it using the denotation of filename instead

Conditionals

if Statements if(x == 3){
printf(x);
}

else-if Statements if(x == 3){
printf(x);
} else if (x < 3) {
printf(x - 1);
}

else statements if(x == 3){
printf(x);
} else if (x < 3) {
printf(x - 1);
} else {
print(0);
}

Ternary Operators: A condensed if-else statement
if (a < b) {
min = a;
} else {
min = b;
}

min = a < b ? a : b; ((This is the ternary Operator for the above if)
to briefly explain it, it essentially will set min to a if the a < b is true otherwise b, the second option, will be what min is set to

Conditionals (cont)

Switch Statements: A Condensed series of cascading else statements, it tests a value and compares it against multiple cases.

```
switch (grade)
{
case 9:
printf("Freshman\n");
break;
default:
printf("Invalid\n");
break;
}
```

Functions

Functions in C A function is a block of reusable logic that may have a defined set of input and output

Built-In Function in C The C Programming language comes with built-in standard library functions
- printf()
- rand()
**Note to be able to use these make sure to #include <stdio.h>

Calling Functions int myNumber = incrementBy(5, 2);

A function is called by stating the function name followed by parentheses. One or more argument values can be placed in the parentheses as the function requires.



By Randomgirll13

Not published yet.

Last updated 17th August, 2023.

Page 1 of 3.

Sponsored by CrosswordCheats.com

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>

Functions (cont)

Storing a return value
`int myNumber = incrementBy(5, 2);`

A function's return value, or output, can be stored in a variable for future use.

Function Signature

Return Type

Variables

Variable Names in C
 In C, Variable names follow specific Rules:
 Names can only be composed of upper and lower case letters, numbers, and underscores.
 The first character must be a letter (upper or lower case).
 No keywords are allowed as the full name (int is not allowed but int_count would work)

Data Types in C
 The four main data types in c are: int, char, double, and float

Declaring Variables in C
`int int_count = 4;`
`double priceApples;`

Setting Variables in C
 When you declare a variable, you do not need to set it right away, but you can set it right away if you want to

Variables (cont)

Variable Casting in C
 You can implicitly or explicitly cast to variables in C. However, implicitly casting may not have the same effect as explicitly casting

Loops and Errors

While Loops, will iterate until a condition is met
`while (a < 10) {`
`a++;`
`}`

do -While loops, while loops that initially execute the body once before checking conditions
`do {`
`printf("not true!");`
`} (while 2 == 3);`

for loops, iterates a set number of times
`for (int i = 0; i <= 10; i++) {`
`printf("Hello!");`
`}`

Loop Keywords
 All Loops can Utilize Keywords like continue and break. Continue will restart the loop without completing anything within the loop past the continue keyword. break will completely stop the loop and continue on after it within the code.

Pointers and Memory

What is a pointer?
 A pointer is a variable that stores the hexadecimal address of the variable it is pointing to within memory

Declaring pointers
`type* ptr;`
`type *ptr;`

Accessing Memory Address
 A Memory Address of a variable is obtained using the reference operator (&).
 *example &var

Dereferencing Pointers
 A Pointer is dereference using the dereference operator (*).
 *Example *ptr

Incrementing and Decrementing Pointers
 Pointers can be incremented and decremented using the + and - arithmetic operators

Accessing Arrays
 Arrays can be accessed by using a pointer to the first element and incrementing and decrementing as necessary



By [Randomgirll13](https://randomgirll13)

cheatography.com/randomgirll13/

Not published yet.

Last updated 17th August, 2023.

Page 2 of 3.

Sponsored by [CrosswordCheats.com](https://crosswordcheats.com)

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>

Operators

Mathematical Symbols in C	Addition: + Subtraction: - Division: / Multiplication: * Incrementing: ++ Decrementing: -- Modulo: %
Assignment Operations in C	Assignment: = Addition then Assignment: += Subtraction then Assignment: -= Multiplication then Assignment: *= Division then Assignment: /= Modulo then Assignment: %=
Comparing Values in C	Both sides Equal?: == Two sides not equal?: != left lower then right?: < left lower or equal to right?: <= left greater than right?: > left greater than or equal to right?: >=

Operators (cont)

Logical Operators in C	and: && (Both sides true?) or: (At least one side true?) not: ! (true = false, false = true)
------------------------	---

Arrays and Strings

Creating Uninitialized Arrays	type arr[array_size]; char word[15];
Creating initialized Arrays	type arr[] = {el1, el2}; char word[] = {'H', 'e', 'l', 'l', 'o', '\0'}
Accessing Array Elements	arr[idx]; word[0]; ('H')
First and Last Array Elements	firstElement = arr[0]; lastElement = arr[arraySize - 1];
sizeof()	int arr[17]; size_t sizeArr = sizeof(arr) / sizeof(arr[0]); Note C does not have a built in way to actually find the size of an array, this is simply a handy trick to find the amount of elements an array should be able to hold. **Note size_t is an unsigned integer type used to represent the size of objects in bytes, its the return type of sizeof()

Arrays and Strings (cont)

Invalid Array Access	While it is possible to access beyond an array, it will cause the program to behave unpredictably
Creating Multidimensional Arrays	initializedMultArr = type arr[12][15]...[13]; initializedMultArr = type arr[15][20]...[n] = {{elm1, elm2}, {elm3}}
Arrays are static	The length of a string cannot be modified as a string is a char array in C
Null Character	All Strings terminate with a null character ('\0')
The length of a string	find the length using the strlen() function
string concatenation	strcat() function can be used to concate two strings
strcpy()	a string can be copied into an empty char array using this function



By Randomgirll13

cheatography.com/randomgirll13/

Not published yet.

Last updated 17th August, 2023.

Page 3 of 3.

Sponsored by [CrosswordCheats.com](https://crosswordcheats.com)

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>