



# Data structure & Algorithms

with C

Kiarash Rahmani

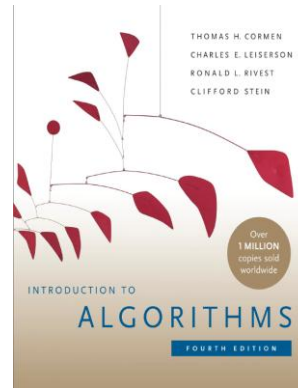
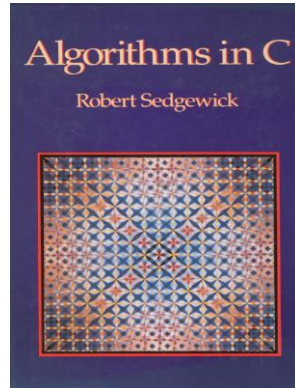
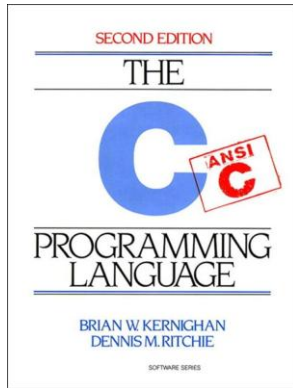


# Introduction

Goals of this course:

- ❖ Learning basics of C programming language
- ❖ Learning basics of Data Structures and Algorithms

Resources :



HomeWorks and Slides:

- ❖ Weekly homework assignments will be available at: <https://kiarashrahmani.ir/DA.html>
- ❖ Slides will be uploaded on the same page every week.

# C Programming Language: What is C?

-C is a general-purpose programming language created by Dennis Ritchie at the Bell Laboratories in 1972.

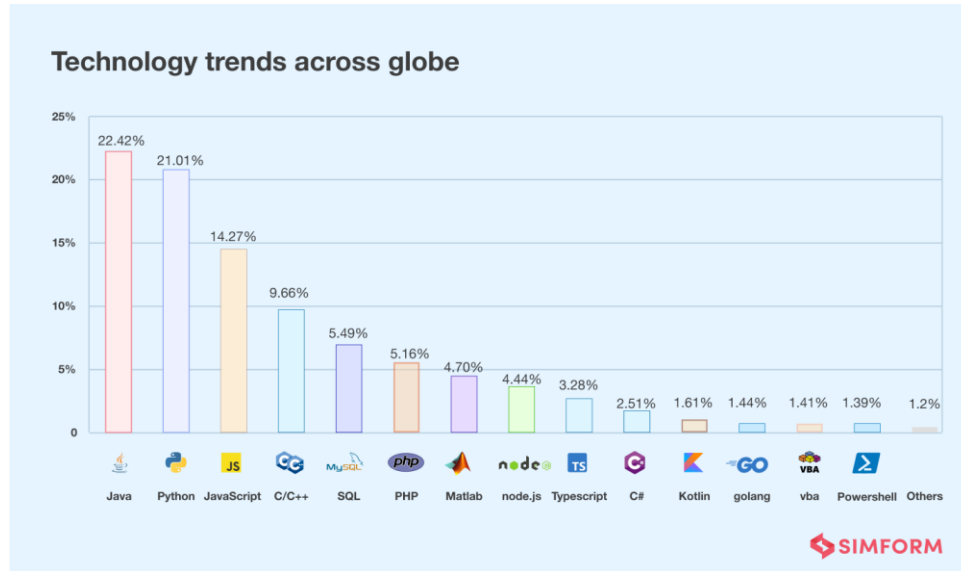


-It is a very popular language, despite being old. Its popularity is mainly because it is a fundamental language in computer science. (which is the main reason I chose C for this course.)

-C is strongly associated with UNIX, as it was developed to write the UNIX operating system.

# C Programming Language: Why Learn C?

- ❖ It is one of the most popular programming languages in the world
- ❖ If you know C, you will have no problem learning other popular programming languages such as Java, Python, C++, C#, etc, as the syntax is similar
- ❖ C is very fast, compared to other programming languages, like Java and Python
- ❖ C is very versatile; it can be used in both applications and technologies



# C Programming Language: C Quickstart

Let's create our first C file.

Open Codeblocks and go to File > New > Empty File.

Write the following C code and save the file as myfirstprogram.c (File > Save File as):

myfirstprogram.c

Code:

```
#include <stdio.h>

int main() {
    printf("Hello World!");
    return 0;
}
```

Result:

```
Hello World!
```

# C Programming Language: C Syntax

main function  
ends the function

```
1 #include <stdio.h>
2
3 int main() {
4     printf("Hello World!");
5     return 0;
6 }
```

header file lib

Blank line

Print function

don't forget it

**Remember:** The compiler ignores white spaces. However, multiple lines make the code more readable.



# C Programming Language: C Statements

A computer program is a list of "instructions" to be "executed" by a computer. In a programming language, these programming instructions are called statements. The following statement "instructs" the compiler to print the text "Hello World" to the screen:

```
printf("Hello World!");
```

**Remember:** If you forget the semicolon (;), an error will occur and the program will not run.

```
printf("Hello World!")
```

```
error: expected ';' before 'return'
```



# C Programming Language: C Statements

Most C programs contain many statements.

The statements are executed, one by one, in the same order as they are written:

```
printf("Hello World!");  
printf("Have a good day!");  
return 0;
```

```
Hello World!Have a good day!
```





# C Programming Language: C Output

To output values or print text in C, you can use the `printf()` function.

## Double Quotes :

When you are working with text, it must be wrapped inside double quotation marks `""`.

If you forget the double quotes, an error occurs:

```
printf("This sentence will work!");
```

```
printf(This sentence will produce an  
error.);
```

```
prog.c: In function 'main':  
prog.c:5:10: error: 'This' undeclared (first use in this function)  
  5 |   printf(This sentence will produce an error.);  
    |           ^~~~~  
prog.c:5:10: note: each undeclared identifier is reported only once for each function  
prog.c:5:14: error: expected ')' before 'sentence'  
  5 |   printf(This sentence will produce an error.);  
    |           ^~~~~~  
    |           )
```

# C Programming Language: C New Lines

The newline character (`\n`) is called an escape sequence, and it forces the cursor to change its position to the beginning of the next line on the screen. This results in a new line.

```
#include <stdio.h>
```

```
int main() {  
    printf("Hello World!\nI am learning C.\nAnd it is awesome!");  
    return 0;  
}
```

```
Hello World!  
I am learning C.  
And it is awesome!
```

# C Programming Language: C Statements

Examples of other valid escape sequences are:

Escape Sequence	Description
<code>\t</code>	Creates a horizontal tab
<code>\\</code>	Inserts a backslash character ( <code>\</code> )
<code>\"</code>	Inserts a double quote character