



SAMPLE TEST PDF

THE STARTING BLOCK OF CODING

---

# C Programming Language

---

*Purpose :*  
Learning

*Author :*  
Louveau Simon

24 mars 2021

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>4</b>
<b>2</b>	<b>Compilation</b>	<b>5</b>
<b>3</b>	<b>The basic components of the C</b>	<b>6</b>
3.1	Identifiers . . . . .	6
3.2	Keywords . . . . .	6
3.3	Comments . . . . .	6
<b>4</b>	<b>Structure of a C program</b>	<b>7</b>
<b>5</b>	<b>The predefined types</b>	<b>8</b>
5.1	The character type . . . . .	8
5.2	Integer types . . . . .	8
5.3	Floating types . . . . .	8
<b>6</b>	<b>Constants</b>	<b>9</b>
6.1	Integer constants . . . . .	9
6.2	The real constants . . . . .	9
6.3	Character constants . . . . .	9
6.4	String constants . . . . .	9
<b>7</b>	<b>The operators</b>	<b>10</b>
7.1	Assignment . . . . .	10
7.2	Arithmetic operators . . . . .	10
7.3	Relational operators . . . . .	10
7.4	Boolean logical operators . . . . .	10
7.5	Bitwise logical operators . . . . .	10
7.6	Compound assignment operators . . . . .	10
7.7	The increment and decrement operators . . . . .	10
7.8	The comma operator . . . . .	10
7.9	The ternary conditional operator . . . . .	10
7.10	The type conversion operator . . . . .	10
7.11	The operator addresses . . . . .	10
<b>8</b>	<b>Conditional branching instructions</b>	<b>11</b>
8.1	"if — else" conditional branching . . . . .	11
8.2	Multiple "switch" . . . . .	11
<b>9</b>	<b>The loop</b>	<b>12</b>

9.1	"while" loop . . . . .	12
9.2	"do — while" loop . . . . .	12
9.3	"for" loop . . . . .	12
<b>10</b>	<b>Unconditional branching instructions</b>	<b>13</b>
10.1	Unconditional "break" branch . . . . .	13
10.2	"Continuous" unconditional branching . . . . .	13
<b>11</b>	<b>Classic input-output functions</b>	<b>14</b>
11.1	The "printf" write function . . . . .	14
11.2	The "scanf" input function . . . . .	14
11.3	Printing and reading characters . . . . .	14
<b>12</b>	<b>Compound types</b>	<b>15</b>
12.1	Tables . . . . .	15
12.2	Structures . . . . .	15
12.3	The bit fields . . . . .	15
12.4	Unions . . . . .	15
12.5	Enumerations . . . . .	15
12.6	Defining compound types with typedef . . . . .	15
<b>13</b>	<b>Pointers</b>	<b>16</b>
13.1	Introduction . . . . .	16
13.2	Basic operators . . . . .	16
13.2.1	The 'address of' operator : & . . . . .	16
13.2.2	The 'content of' operator : * . . . . .	16
13.3	Elementary operations on pointers . . . . .	16
13.4	Addressing the components of an array . . . . .	16
13.5	Pointers and strings . . . . .	16
<b>14</b>	<b>Dynamic allocation of memory</b>	<b>17</b>
<b>15</b>	<b>Linked lists and stack LIFO implementations</b>	<b>18</b>
<b>16</b>	<b>Matrices implementations</b>	<b>19</b>
<b>17</b>	<b>Queue, FIFO buffer</b>	<b>20</b>
<b>18</b>	<b>The main function</b>	<b>21</b>
<b>19</b>	<b>Exercice Seam Carving</b>	<b>22</b>

<b>20</b>	<b>Some tips for writing a C program</b>	<b>23</b>
<b>21</b>	<b>References</b>	<b>24</b>

# **1 Introduction**

## 2 Compilation

## **3 The basic components of the C**

### **3.1 Identifiers**

### **3.2 Keywords**

### **3.3 Comments**

## 4 Structure of a C program



## 5 The predefined types

### 5.1 The character type

### 5.2 Integer types

### 5.3 Floating types

## 6 Constants

### 6.1 Integer constants

### 6.2 The real constants

### 6.3 Character constants

### 6.4 String constants

## 7 The operators

### 7.1 Assignment

### 7.2 Arithmetic operators

### 7.3 Relational operators

### 7.4 Boolean logical operators

### 7.5 Bitwise logical operators

### 7.6 Compound assignment operators

### 7.7 The increment and decrement operators

### 7.8 The comma operator

### 7.9 The ternary conditional operator

### 7.10 The type conversion operator

### 7.11 The operator addresses

## 8 Conditional branching instructions

### 8.1 "if — else" conditional branching

### 8.2 Multiple "switch"

## 9 The loop

### 9.1 "while" loop

### 9.2 “do — while” loop

### 9.3 “for” loop

## 10 Unconditional branching instructions

### 10.1 Unconditional "break" branch

### 10.2 "Continuous" unconditional branching

## 11 Classic input-output functions

### 11.1 The "printf" write function

### 11.2 The "scanf" input function

### 11.3 Printing and reading characters

## 12 Compound types

### 12.1 Tables

### 12.2 Structures

### 12.3 The bit fields

### 12.4 Unions

### 12.5 Enumerations

### 12.6 Defining compound types with typedef



## 13 Pointers

### 13.1 Introduction

### 13.2 Basic operators

#### 13.2.1 The 'address of' operator : &

#### 13.2.2 The 'content of' operator : \*

### 13.3 Elementary operations on pointers

### 13.4 Addressing the components of an array

### 13.5 Pointers and strings

## 14 Dynamic allocation of memory

## 15 Linked lists and stack LIFO implementations

## 16 Matrices implementations

## 17 Queue, FIFO buffer

## 18 The main function

## 19 Exercice Seam Carving

## 20 Some tips for writing a C program



## 21 References