# Introduction to C++ Assignment\_ FARHAT Nour

## Versions of C++ and Major Changes

C++ has evolved over the years, introducing several improvements and new features in each version.

C++98 (1998) – This was the first official C++ standard, introducing the Standard Template Library (STL), which included containers, algorithms, and iterators.

C++03 (2003) – A minor update that improved the compatibility of C++98 with different compilers and fixed some technical inconsistencies.

C++11 (2011) – A major update that modernized the language with new features like the auto keyword, smart pointers, lambda expressions and multi-threading support.

C++14 (2014) – A small update that enhanced C++11 by introducing `std::make\_unique`, relaxed `constexpr` functions, and generic lambda expressions.

C++17 (2017) – Added new features such as structured bindings, parallel algorithms, and constexpr if

C++20 (2020) – Introduced modules, coroutines, concepts, ranges, `std::span`, and improved concurrency features.

C++23 (2023) – Focused on enhancing safety, improving algorithms, and and expressiveness.

## Reserved Keywords in C++ and Their Usage

C++ has several reserved keywords that have specific meanings and cannot be used as variable names or identifiers. We have, for example:

### 1. int

It’s used to declare integer variables.

Example: int age = 25;

### 2. float

It is used to declare floating-point numbers with single precision.

Example: float pi = 3.14;

### 3. if

It is used to create conditional logic.

Example:   
if (age >= 18) {  
 std::cout << "Adult";  
}

### 4. while

The `while` loop is used to execute a block of code as long as the condition is true.

Example:  
int i = 0;  
while (i < 5) {  
 std::cout << i << " ";  
 i++;  
}

### 5. for

The `for` loop is used for iterating over a sequence.

Example:  
for (int i = 0; i < 5; i++) {  
 std::cout << i << " ";  
}

### 6. return

It is used to return a value from a function.

Example:  
int sum(int a, int b) {  
 return a + b;  
}

### 7. class

It is used to define a class in object-oriented programming.

Example:

class Car {  
public:  
 std::string brand;  
};

### 8. public

The `public` access specifier allows class members to be accessed from outside the class.

Example:

class Person {  
public:  
 std::string name;  
};

### 9. new

It is used for dynamic memory allocation.

Example:  
int\* ptr = new int(10);

### 10. delete

It is used to free dynamically allocated memory.

Example:  
delete ptr;

## How are bits allocated for each data type in 32-bit and 64-bit systems?

Different data types in C++ have varying memory allocations depending on whether the system is a 32-bit or 64-bit architecture.

• char: 1 byte (8 bits) on both 32-bit and 64-bit systems.

• bool: 1 byte (8 bits) on both 32-bit and 64-bit systems.

• short: 2 bytes (16 bits) on both 32-bit and 64-bit systems.

• int: 4 bytes (32 bits) on both 32-bit and 64-bit systems.

• long: 4 bytes (32 bits) on a 32-bit system, 8 bytes (64 bits) on a 64-bit system.

• long long: 8 bytes (64 bits) on both 32-bit and 64-bit systems.

• float: 4 bytes (32 bits) on both 32-bit and 64-bit systems.

• double: 8 bytes (64 bits) on both 32-bit and 64-bit systems.

• void: 0 bytes, as it represents the absence of a value.