

## QCM - Introduction au C++

**Question 1.** Quel est l'affichage du programme suivant ?

```
#include <iostream>

int main() {
    int a = 10;
    int b = 3;
    std::cout << a % b << std::endl;
}
```

- A) 3
- B) 1
- C) 0
- D) 3.33

**Question 2.** Quelle déclaration initialise correctement une variable de type `std::string` ?

- A) `std::string s = "hello";`
- B) `string s = 'hello';`
- C) `char[] s = "hello";`
- D) `std::string s = 'hello';`

**Question 3.** Le C++ est un langage :

- A) Interprété
- B) Exécuté sur une machine virtuelle
- C) Compilé en code machine natif
- D) Uniquement script

**Question 4.** Quel est l'affichage du programme suivant ?

```
#include <iostream>

int main() {
    bool a = true;
    bool b = false;
    std::cout << (a && b) << " " << (a || b) << std::endl;
}
```

- A) true false
- B) 1 0
- C) 0 1
- D) Erreur de compilation

**Question 5.** Quelle directive permet d'inclure la bibliothèque d'entrées/sorties standard en C++ ?

- A) `#import <iostream>`

- B) `#include <stdio>`
- C) `#include <iostream>`
- D) `using iostream;`

**Question 6.** Quelle est la fonction obligatoire dans tout programme C++ ?

- A) `void start()`
- B) `int main()`
- C) `void main()`
- D) `int run()`

**Question 7.** Quel opérateur est utilisé pour envoyer des données vers `std::cout` ?

- A) `>>`
- B) `<<`
- C) `->`
- D) `::`

**Question 8.** Quelle est la taille en octets d'un `int` sur une machine standard ?

- A) 1 octet
- B) 2 octets
- C) 4 octets
- D) 8 octets

**Question 9.** Quel mot-clé permet au compilateur de déduire automatiquement le type d'une variable ?

- A) `var`
- B) `let`
- C) `auto`
- D) `typeof`

**Question 10.** Quel est l'affichage du programme suivant ?

```
#include <iostream>

int main() {
    int x = 5;
    int y = x++;
    std::cout << x << " " << y << std::endl;
}
```

- A) 5 5
- B) 6 5
- C) 6 6
- D) 5 6

**Question 11.** Quel est le résultat de l'expression suivante ?

```
int a = 5 / 2;
std::cout << a;
```

- A) 2.5
- B) 2
- C) 3

D) Erreur de compilation

**Question 12.** Que fait le mot-clé `const` dans la déclaration suivante ?

```
const float pi = 3.14159f;
```

- A) Il déclare une variable qui peut être modifiée une seule fois
- B) Il déclare une variable dont la valeur ne peut plus être modifiée après initialisation
- C) Il déclare une variable globale
- D) Il déclare une variable statique

**Question 13.** Quel est l'affichage du programme suivant ?

```
#include <iostream>

void increment(int a) {
    a = a + 1;
}

int main() {
    int x = 3;
    increment(x);
    std::cout << x << std::endl;
}
```

- A) 4
- B) 3
- C) 0
- D) Erreur de compilation

**Question 14.** On souhaite modifier la variable originale dans une fonction. Quelle signature est correcte ?

- A) `void modifier(int a)`
- B) `void modifier(int* a)`
- C) `void modifier(int& a)`
- D) `void modifier(const int& a)`

**Question 15.** Quel est le résultat du code suivant ?

```
#include <iostream>
#include <vector>

int main() {
    std::vector<int> v = {10, 20, 30};
    v.push_back(40);
    std::cout << v.size() << std::endl;
}
```

- A) 3
- B) 4
- C) 5
- D) Erreur de compilation

**Question 16.** Quelle est la différence principale entre `struct` et `class` en C++ ?

- A) `struct` ne peut pas contenir de méthodes
- B) `class` ne peut pas contenir d'attributs publics

- C) Les membres d'une `struct` sont publics par défaut, ceux d'une `class` sont privés par défaut  
D) `struct` est un type C incompatible avec le C++

**Question 17.** Quel est l'affichage du code suivant ?

```
#include <iostream>

int main() {
    for (int i = 0; i < 3; i++) {
        // boucle
    }
    std::cout << i << std::endl;
}
```

- A) 3  
B) 2  
C) 0  
D) Erreur de compilation (i n'est plus accessible)

**Question 18.** Quelle est la syntaxe correcte pour une boucle `for` étendue (range-based `for`) sur un vecteur ?

```
std::vector<int> valeurs = {1, 2, 3, 4, 5};
```

- A) `for (int v in valeurs)`  
B) `for (int v : valeurs)`  
C) `for (int v of valeurs)`  
D) `for each (int v, valeurs)`

**Question 19.** Que produit le code suivant ?

```
#include <iostream>

auto a = 5;
auto b = 8.4f;
auto c = 4.2;
```

Quels sont les types déduits pour `a`, `b` et `c` ?

- A) `int`, `float`, `float`  
B) `int`, `double`, `double`  
C) `int`, `float`, `double`  
D) `auto`, `auto`, `auto`

**Question 20.** Quelle est la bonne commande pour compiler un fichier `main.cpp` en un exécutable `prog` avec `g++` ?

- A) `g++ -o prog main.cpp`  
B) `gcc -o prog main.cpp`  
C) `g++ main.cpp --output prog`  
D) `compile main.cpp prog`

**Question 21.** Quel est l'affichage du programme suivant ?

```
#include <iostream>
#include <vector>

void multiply(std::vector<float> vec, float s) {
    for (int k = 0; k < vec.size(); ++k)
        vec[k] = s * vec[k];
}

int main() {
    std::vector<float> v = {1.0f, 2.0f, 3.0f};
    multiply(v, 10.0f);
    std::cout << v[0] << " " << v[1] << " " << v[2] << std::endl;
}
```

- A) 10 20 30
- B) 1 2 3
- C) 0 0 0
- D) Erreur de compilation

**Question 22.** Considérez le code suivant. Quel est l'affichage ?

```
#include <iostream>

int x = 5;

int main() {
    int x = 10;
    {
        int x = 20;
        std::cout << x << " ";
    }
    std::cout << x << std::endl;
}
```

- A) 5 10
- B) 20 10
- C) 20 5
- D) Erreur de compilation (redéfinition de x)

**Question 23.** Quel est le problème dans le code suivant ?

```
#include <iostream>

int main() {
    int c = addition(5, 3);
    std::cout << c << std::endl;
    return 0;
}

int addition(int a, int b) {
    return a + b;
}
```

- A) La fonction addition a un type de retour incorrect
- B) La fonction addition est utilisée avant d'être déclarée
- C) Il manque #include <cmath>
- D) Le programme compile correctement

**Question 24.** Considérez la structure et la fonction suivantes :

```

#include <iostream>
#include <cmath>

struct vec3 {
    float x, y, z;

    float norm() const {
        return std::sqrt(x*x + y*y + z*z);
    }
};

float norm(const vec3& p) {
    return std::sqrt(p.x*p.x + p.y*p.y + p.z*p.z);
}

int main() {
    vec3 v = {3.0f, 4.0f, 0.0f};
    std::cout << v.norm() << " " << norm(v) << std::endl;
}

```

Quel est l'affichage ?

- A) 5 5
- B) 7 7
- C) 25 25
- D) Erreur de compilation (conflit de noms)

**Question 25.** Quel est le résultat du code suivant ?

```

#include <iostream>

int solve(int a, int b) {
    return -b / a;
}

float solve(float a, float b, float c) {
    float delta = b*b - 4*a*c;
    return (-b + std::sqrt(delta)) / (2*a);
}

int main() {
    auto x = solve(2, 6);
    std::cout << x << std::endl;
}

```

- A) -3.0
- B) -3
- C) 3
- D) Erreur de compilation (ambiguïté de surcharge)