

TOÁN TỬ VÀ PHẠM VI TRUY XUẤT

Biên soạn: Nhóm Nghiên Cứu Blockchain Khoa HTTT

1. Toán tử

- **Toán tử số học:** Trong ngôn ngữ lập trình Solidity có các toán tử: +, -, *, /, %, ++, -- nhưng ngôn ngữ lập trình C++
- **Toán tử so sánh:** Vẫn có toán tử so sánh như c++: >, <, =, !=, ==, >=, <=
- **Toán tử Logic:** Giống như C++ những toán tử logic vẫn có &&, ||, !
- **Toán tử ba ngôi:** Vẫn thực hiện trong ngôn ngữ C++ <Điều kiện> ? A:B

2. Biến và phạm vi truy xuất biến

- Có 3 loại phạm vi biến cơ bản: **State Variables, Local Variables, Global Variables.**
 - **State Variables:** Biến truy xuất trong phạm vi Contract.
 - **Local Variables:** Biến truy xuất **bên trong thân hàm** của Contract.
 - **Global Variables:** Biến đặc biệt được **tạo ra bởi Blockchain** cho biết thông tin về **một block** và **thành phần của transactions**
 - Ví dụ 1 về phạm vi biến: **States Variables, Local Variables**
 - ⇒ Xem ví dụ: [03_Variables_Scopes.sol](#)

```
1  pragma solidity ^0.8.6;
2
3  /**
4   * Smart Contract About Variables Scope Test
5   */
6  contract VariablesScopeTest {
7      uint soNguyenContract = 1; //States Variables
8
9      /**
10     * Function Gõ So Nguyen
11     */
12     function goiSoNguyen() public view returns(uint) {
13         uint soNguyenFunction = 2; //Local Variables
14         return soNguyenContract + soNguyenFunction;
15     }
16 }
```

- Ví dụ 2 về phạm vi biến: **Global Variables**

⇒ Xem ví dụ: **03_GlobalVariables.sol**

```
pragma solidity ^0.8.6;

/**
 * Smart Contract use Global Variables
 */
contract GlobalVariablesTest {

    /**
     * Function Get Time Stamp Block in Solidity
     */
    function getTimeStampBlock() public view returns(uint) {
        return block.timestamp;
    }
}
```

- Phạm vi truy xuất biến: **Sử dụng cho hàm và biến**
 - **public**: trong phạm vi contract, ngoài phạm vi contract
 - ⇒ Xem ví dụ **03_PublicContract.sol**
 - ⇒ Xem ví dụ **03_GetPublicContractOut.sol**
 - **private**: trong phạm vi contract chứa
 - ⇒ Xem ví dụ **03_PrivateContract.sol**
 - **Internal**: trong phạm vi contract chứa và những contract con.

3. Tài liệu tham khảo

- [1] https://www.tutorialspoint.com/solidity/solidity_variables.htm, [Online] [Thời gian truy cập: 25/05/2022].
- [2] https://www.tutorialspoint.com/solidity/solidity_variable_scope.htm, [Online] [Thời gian truy cập: 17/06/2022]
- [3] https://www.tutorialspoint.com/solidity/solidity_operators.htm, [Online] [Thời gian truy cập: 17/06/2022]