

Name: VEN Dara  
ID : e20200469  
Class: I3-GIC (C)

1. ភាពខុសគ្នារវាង Recursion និង Iteration គឺ៖

ក. Recursion ជាវិធីសាស្ត្រក្នុងការដោះស្រាយបញ្ហា ដែលបញ្ហាត្រូវបានបំបែកជាតូចៗ រួចដោះស្រាយដោយប្រើខ្លួនឯងមកដោះស្រាយម្តងហើយម្តងទៀត រហូតដល់ចំណុចលក្ខណៈណាមួយដែលត្រូវបញ្ឈប់។

ឧទាហរណ៍៖ Recursion Function ត្រូវបានប្រើក្នុងការរកតម្លៃ Factorial ៖

```
public static int calculateFactorial(int n) {  
    if (n == 0) {  
        return 1;  
    } else {  
        return n * calculateFactorial(n - 1);  
    }  
}
```

ដូចតាម Code Java ខាងលើ calculateFactorial បានធ្វើការបោះ តម្លៃ Parameter ទៅខ្លួនឯង បានន័យថាយកខ្លួនឯងមកប្រើ រួចដកមួយរហូតតម្លៃ  $n = 0$ ។

ខ.ចំណែកឯ Iteration ជាវិធីសាស្ត្រមួយ ដែលជំនួសឱ្យការហៅខ្លួនឯងមកប្រើឡើងវិញ Iteration បានប្រើប្រាស់ Loop ជំនួស ដើម្បីដំណើរការក្នុងការដោះស្រាយបញ្ហា តំណើរការត្រូវបានដកដល់លក្ខខណ្ឌមួយដែលត្រូវឈប់។

ឧទាហរណ៍៖ Iteration Function ត្រូវបានប្រើក្នុងការរកតម្លៃ Factorial ៖

```
public static int factorialIterative(int n) {  
    int result = 1;  
    for (int i = 1; i <= n; i++) {  
        result *= i;  
    }  
    return result;  
}
```

ក្នុង Iteration Function loop(while, for... បានដើរតួរសំខាន់ ដើម្បីឱ្យ Code Block ទាំងមូលដើរឡើងវិញ។

2. Logic gate មាន ៣ ប្រភេទ៖

ក. ឈ្មាប់និង (And Gate ៖ ចេញពិតបាន តែករណីធាតុទាំងអស់ពិត

ខ. ឈ្មាប់ឬ (Or Gate ៖ ចេញពិត ឱ្យតែករណីមានធាតុណាមួយពិត

គ. ឈ្មាប់មិន (Not Gate ៖ បំប្លែងធាតុឱ្យទៅបញ្ញាស ពិត -> មិនពិត មិនពិត -> ពិត

**To build a circuit for the function  $xy + (\text{not } x)y + x(\text{not } y)$ :**

- គណនា  $xy$  ដោយប្រើ ឈ្មាប់និង ជាមួយធាតុបញ្ចូល  $x$  និង  $y$  ។
- គណនា  $(\text{Not } x) y$  ដោយប្រើ  $x$  កាត់តាម ឈ្មាប់មិន ហើយបន្ទាប់មកប្រើ ឈ្មាប់និង ដែលមាន  $(\text{Not } x)$  និង  $y$  ជាធាតុបញ្ចូល។
- គណនា  $x (\text{Not } y)$  ដោយប្រើ  $y$  តាមរយៈ NOT gate ហើយបន្ទាប់មកប្រើ ឈ្មាប់និង ដែលមាន  $x$  និង  $(\text{Not } y)$  ជាធាតុបញ្ចូល។
- ជាចុងក្រោយ គណនាមុខងារដោយប្រើ ឈ្មាប់ឬ ជាមួយនឹងលទ្ធផលនៃការគណនាទាំងបីខាងលើជាធាតុបញ្ចូល។