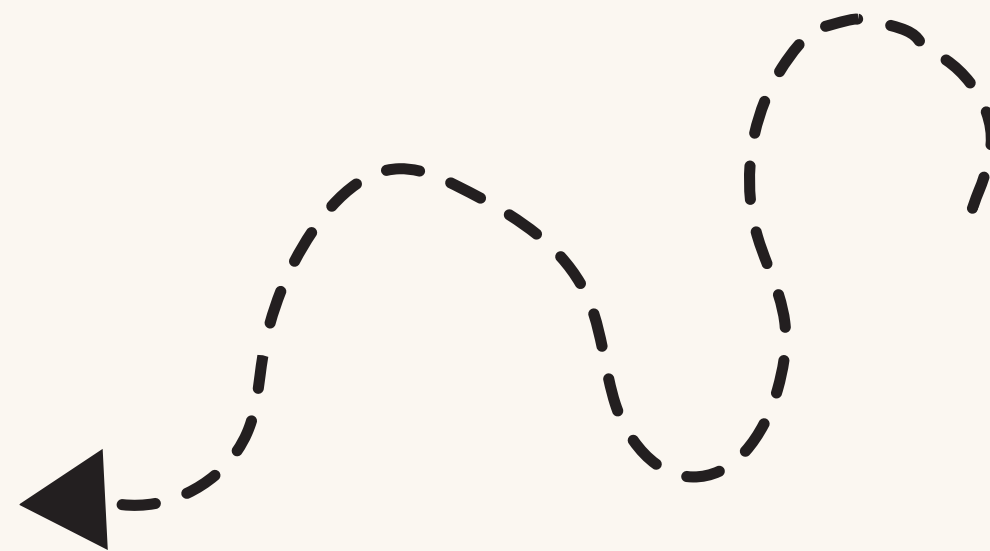




C++



RECURSIVE



Agenda

1. Apa itu Rekursif ?
 2. Apa fungsinya?
 3. Bagaimana Ide Dasar?
 4. Penerapan Bahasa C++?
- 
- 
- 

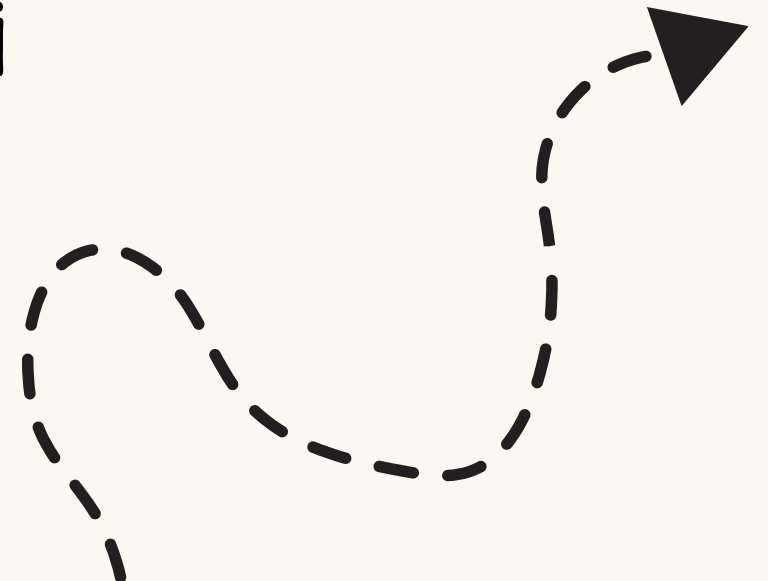


Hi there! I'm Recursive!!



Introduction

- Rekursi adalah proses pengulangan sesuatu dengan cara kesamaan-diri
- Rekursif adalah salah satu metode dalam dunia matematika dimana definisi sebuah fungsi mengandung fungsi itu sendiri

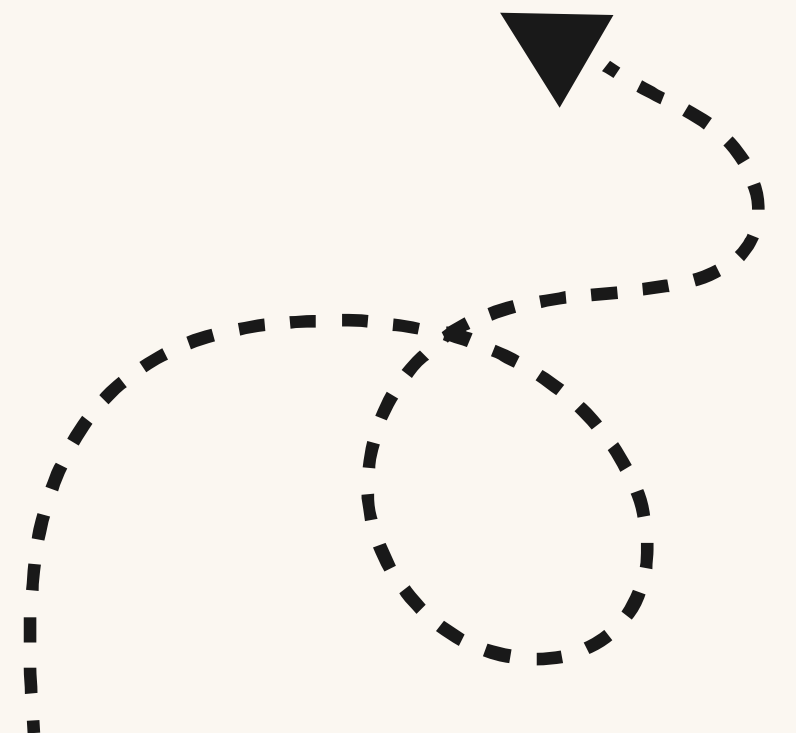




* * Apa sih sebenarnya fungsi Rekursif

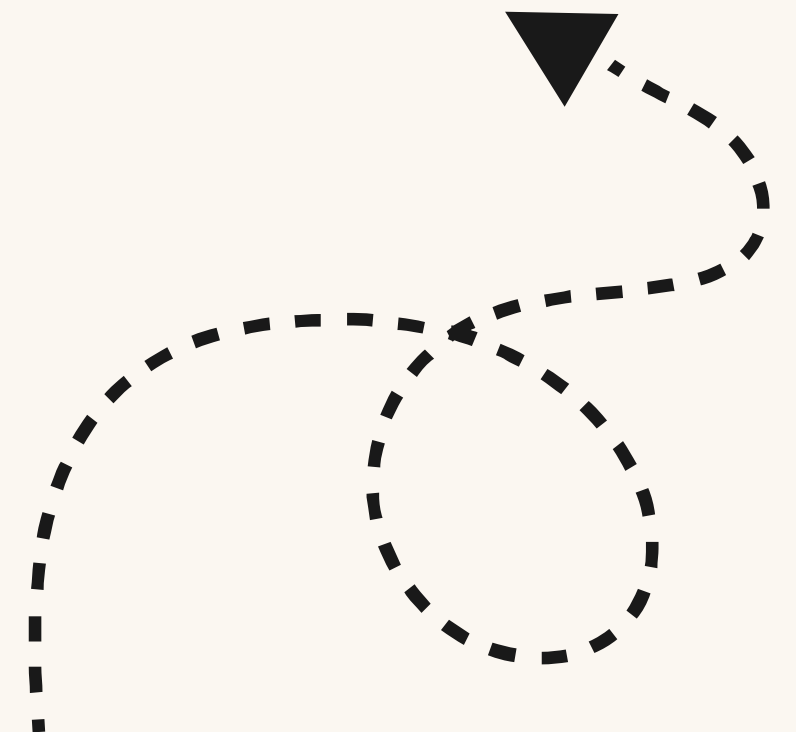
Fungsi rekursif merupakan fungsi yang melakukan perulangan dengan mengacu pada dirinya sendiri, sedangkan fungsi iteratif merupakan fungsi yang melakukan perulangan terutama dengan menggunakan loop

Finite Recursion => Apa maksudnya ??



~~Kalu~~ apa Ide Dasar Dari Rekursif ?

- Ketika dilakukan pemanggilan fungsi maka komputer akan menyimpan
 1. Semua argumen dan variabel lokal
 2. Fungsi yang aktif (current function)
- Penyimpanan dilakukan dengan memasukkan fungsi dan variabel ke dalam stack



Penerapan Pangkat pada rekursif

```
13 int recursive(int a, int b)
14 {
15     if(b == 0)
16     {
17         return 1;
18     }
19     else
20     {
21         return a * recursive(a, b - 1);
22     }
23 }
```

Penerapan Pangkat pada iterasi

```
5 int pangkat(int a, int b) {
6     int hasil = a;
7     for (int i = 1; i < b; i++) {
8         hasil = hasil * a;
9     }
10    return hasil;
11 }
```

Penerapan factorial pada rekursif

```
15 int recursive(int n)
16 {
17     if (n == 0)
18     {
19         return 1;
20     }
21     else
22     {
23         return n * recursive(n-1);
24     }
25 }
```

Penerapan factorial pada iterasi

```
5 int factorial(int n)
6 {
7     int hasil = n;
8     for (int i = n-1; i > 0; i--)
9     {
10         hasil = hasil*i;
11     }
12     return hasil;
13 }
```


Penerapan fibonacci pada rekursif

```
21 ✓ int recursive(int n) {  
22     int s1, s2;  
23 ✓   if (n == 0) {  
24       return 0;  
25 ✓   } else if (n == 1) {  
26       return 1;  
27 ✓   } else {  
28       s1 = recursive(n - 1);  
29       s2 = recursive(n - 2);  
30       return s1 + s2;  
31   }  
32 }
```

Penerapan fibonacci pada iterasi

```
5  int fibonacci(int n)  
6  {  
7      int f1 = 0, f2 = 1, fibo;  
8  ✓   if(n == 0){  
9       return 0;  
10     }  
11 ✓   else if(n == 1){  
12       return 1;  
13     }  
14     else  
15 ✓   {  
16 ✓       for(int i = 0; i < n; i++){  
17           fibo = f1 + f2;  
18           f2 = f1;  
19           f1 = fibo;  
20       }  
21       return fibo;  
22   }  
23 }
```


IDEAS BUBBLE SORT

```
main.cpp x +
main.cpp > f swap Format
1  #include <iostream>
2
3  using namespace std;
4
5  void swap(int &a, int &b) {
6      int temp = a;
7      a = b;
8      b = temp;
9  }
10
11 void bubbleSortRecursive(int arr[], int n) {
12     if (n == 1) {
13         return;
14     }
15     for (int i = 0; i < n - 1; ++i) {
16         if (arr[i] > arr[i + 1]) {
17             swap(arr[i], arr[i + 1]);
18         }
19     }
20     bubbleSortRecursive(arr, n - 1);
21 }
22
23 int main() {
24     int arr[] = {64, 25, 12, 22, 11};
25     int n = sizeof(arr) / sizeof(int);
26     cout << "Array sebelum diurutkan: ";
27     for (int i = 0; i < n; ++i) {
28         cout << arr[i] << " ";
29     }
30     cout << endl;
31     bubbleSortRecursive(arr, n);
32     cout << "Array setelah diurutkan: ";
33     for (int i = 0; i < n; ++i) {
34         cout << arr[i] << " ";
35     }
36     cout << endl;
37     return 0;
38 }
```

IMPLEMENTASI RECURSIVE

Bubble Sort, mengurutkan bilangan terkecil hingga terbesar. Implementasi kehidupan sehari-hari : Mengurutkan peringkat berdasarkan absensi atau NRP

THANK
YOU!

