**NAMA : ARTHA NUGRAHA FAJAR SIDIQ**

**NIM : 1117000439**

1. **DERET FIBONACCI**

Bilangan Fibonacci adalah suatu deret bilangan bulat (*integer*) yang tak berhingga dimana bilangan tersebut secara berurutan dapat didefinisikan sebagai berikut ini :

**0 1   1   2   3   5   8   13   21   34   55   89..........n**

Deret bilangan Fibonacci di atas, maka dapat dipahami bahwa untuk mencari nilai suku ke-n pada deret bilangan Fibonacci dengan cara menjumlahkan nilai deret yang saling berdekatan.

1. **ALGORITMA ITERATIF**

int min = 0;

int max = 1;

int value;

for (int i = 0; i <= n[j]; i++) {

value = min + max;

min = max;

max = value;

}

Algoritma ini mirip dengan algoritma rekursif namun ada sedikit perbedaan. Pada algoritma ini, fungsi iteratif digantikan oleh sebuah iterasi. Idenya adalah sebagai berikut :

1. Menyimpan nilai f(0) dan f(1) dalam 2 variabel
2. Menghitung nilai f(2) dari nilai f(0) dan f(1) yang sudah disimpan sebelumnya, lalu menyimpan nilai f(2) dan f(1)
3. Menghitung nilai f(3) dari nilai f(1) dan f(2) yang sudah disimpan sebelumnya, lalu menyimpan nilai f(3) dan f(2)
4. Proses tersebut dilakukan berulang-ulang sampai diperoleh nilai f(n)
5. **ANALISIS**

Dari potongan program di atas, langsung terlihat bahwa kompleksitas waktunya adalah :

T(n) = 2 + 3(n-1) = 3n -1

Dengan nilai T(n) tersebut, bisa ditentukan notasi O besarnya :

T(n) = 3n-0

T(n) = 3n

Sehingga diperoleh T(n) = O(n).

Untuk nilai n >1 T(n) pasti bernilai 3n-1. Oleh karena itu, algoritma ini adalah algoritma yang stabil. Jadi kompleksitas untuk kasus terbaik sama dengan kompleksitas untuk kasus terburuknya.