Nama : Fajar Satria

NRP : 05111940000083

Resume Kuliah PAA – 29 Maret 2021

Teknik yang digunakan untuk membuktikan apakah sebuah algoritma berlaku benar untuk segala kejadian adalah teknik *loop invariant*. Loop invariant mirip dengan induksi matematika, tapi bukan induksi matematika sepenuhnya karena induksi matematika tidak cukup diterapkan pada struktur algoritma loop invariant. Step pada loop invariant adalah inisiasi, maintenance, dan termination.

Kebenaran suatu algoritma bisa dihadirkan dengan mencari relevansi atau hubungan antara sebuah persoalan dengan kajian teori yang benar. Meskipun demikian, kajian teori tidak bisa semuanya dibahas dalam satu mata kuliah.

Pendekatan konseptual teoritis terhadap persoalan perancangan algoritma dengan contoh kasus CUTSTICK dan MONOTONOUS NUMBER.

a. CUTSTICK

- Untuk menyelesaikan problem ini, perlu dilakukan observasi terlebih dahulu. Observasi bertujuan mendapatkan keteraturan yang merelasikan panjang tongkat dengan jumlah potongan optimal.
- Prinsip yang digunakan adalah ketidaksamaan segitiga *(Triangle Inequality)*
- Dalam matematika, ketidaksamaan segitiga menyatakan bahwa untuk sembarang segitiga, jumlah panjang sembarang dua sisinya haruslah lebih besar daripada panjang sisi ketiganya.
- Untuk dapat memahami lebih mendalam soal ini, maka harus dibuat terlebih dahulu table case nya.

Panjang Tongkat	Potongan Optimal Ke-				
	1	2	3	4	5
1	1				
2	1	1			
3	1	2			
4	1	1	2		
5	1	1	3		
6	1	1	4		
7	1	1	2	3	
8	1	1	2	4	
9	1	1	2	5	
10	1	1	2	6	
11	1	1	2	7	
12	1	1	2	3	5
13	1	1	2	3	6
14	1	1	2	3	7
15	1	1	2	3	8

- Dengan melihat table diatas, akan terlihat pola bilangan Fibonacci dengan urutan angka 1, 1, 2, 3, 8, dst. Dengan demikian penyelesaian soal ini akan lebih efisien jika menggunakan deret Fibonacci.
- Dengan melihat batasan pada soal hanya mencapai 10^5, jadi deret Fibonacci yang dibutuhkan hanya sampai deret ke 27.

- PSEUDOCODE

```
SOLVE(n)
let A[0..27] be a new array
A[1] = A[2] = 1;
for i=3 to 26
        A[i] = A[i-1] + A[i-2]
if n==1
        return 1
else if n==2
        return 2
else    ans = 0
        for i=1 to 26
                if n >= A[i]
                        ans = ans+1
                        n = n-A[i]
                else i = 27
        return ans
```

- ACCEPTED SOURCECODE

```
#include <cstdio>
using namespace std;
int a[27];
int main(){
    a[1]=a[2]=1;
    for(int i=3;i<27;++i)
        a[i] = a[i-1] + a[i-2];
    int n;
    while(scanf("%d",&n)!=EOF){
        if(n==1) puts("1");
        else if(n==2) puts("2");
        else{
            int ans=0;
            for(int i=1;i<27;++i){
                if(n>=a[i]){
                    ans++;
                    n -= a[i];
                }else i=27;
            }
            printf("%d\n",ans);
        }
    }
    return 0;
}
```

- SCREENSHOT SUBMISSION

Fajar: submissions
Cut Stick

ID		DATE	PROBLEM	RESULT	TIME	MEM	LANG
27619309	☐	2021-03-29 09:41:30	Cut Stick	**accepted** edit ideone it	0.00	4.7M	CPP

b. MONOTONOUS NUMBER

- Untuk menyelesaikan problem ini, perlu dilakukan observasi terlebih dahulu. Dari observasi didapatkan bahwa

  * x*i* = banyaknya digit *i* pada bilangan

  * Increasing n-digit order :

  x1 + x2 + x3 + x4 + x5 + x6 + x7 + x8 + x9 = n

  x1 .. x9 >= 0

  * Decreasing n-digit order :

  x9 + x8 + x7 + x6 + x5 + x4 + x3 + x2 + x1 + x0 = n

  x0 .. x9 >= 0

- Dengan prinsip increasing dan decreasing tersebut maka diperlukan teorema kombinasi dengan repetisi.

  TEOREMA 2 : There are C(n+r−1,r) = C(n+r−1,n−1) r-combinations from a set with n elements when repetition of elements is allowed.

  Contoh :

  How many solutions does the equation

  $$x_1 + x_2 + x_3 = 11$$

  have, where x_1, x_2, and x_3 are nonnegative integers?

  Solution: To count the number of solutions, we note that a solution corresponds to a way of selecting 11 items from a set with three elements so that x_1 items of type one, x_2 items of type two, and x_3 items of type three are chosen. Hence, the number of solutions is equal to the number of 11-combinations with repetition allowed from a set with three elements. From Theorem 2 it follows that there are

  $$C(3 + 11 - 1, 11) = C(13, 11) = C(13, 2) = \frac{13 \cdot 12}{1 \cdot 2} = 78$$

  solutions.

- Solusi dengan kombinatorik

$$\text{Rasio} : \frac{\binom{N+10-1}{N} - 1}{\binom{N+9-1}{N}} = \frac{\frac{(N+9)!}{N!\,9!} - 1}{\frac{(N+8)!}{N!\,8!}}$$

$$= \frac{\frac{(N+9)(N+8)\dots(N+1) - 9!}{\cancel{9!}\;9}}{\frac{(N+8)(N+7)\dots(N+1)}{\cancel{8!}}}$$

$$= \frac{N+9}{9} - \frac{8!}{(N+8)(N+7)\dots(N+1)}$$

$$= \frac{N+9}{9} - \frac{8!}{\prod_{i=N+1}^{N+8} i}$$

- PSEUDOCODE

  SOLVE(n)

  1. val = 1.0
  2. for i=1 to 8
  3. val = val * (n + i)
  4. ans = ((n + 9.0) / 9.0 – (40320.0 / val))
  5. return ans

- ACCEPTED SOURCECODE

```
#include <cstdio>
using namespace std;
int main(){
    int t,n;
    scanf("%d",&t);
    while(t--){
        scanf("%d",&n);
        double val=1.0;
        for(int i=1;i<9;i++)
            val=val*(double)(n+i);
        double result=(((double)n+9.0)/9.0-(40320.0/val));
        printf("%.6lf\n",result);
    }
    return 0;
}
```

- SCREENSHOT SUBMISSION

Fajar: submissions
Monotonous numbers

ID		DATE	PROBLEM	RESULT	TIME	MEM	LANG
27619463	☐	2021-03-29 10:21:18	Monotonous numbers	accepted edit ideone it	0.01	4.7M	CPP