LKP 4 Analgor

Halida Fiadnin - G6401211142

No
Date

A. Iteratif Sederhana

1)
```
int n;
scanf ("%d", &n);
for (int i=n; i>0; i=i/2) {
    //nilai konstan
    break;
}
```

operasi	times
int n	1
scanf	1
i=n	1
i>0	$\log_2(n)+2$
i=i/2	$\log_2(n)+1$
konstan	$(\log_2(n)+1)\times 1$
break	$(\log_2(n)+1)\times 1$

$T(n) = 1+1+1+\log_2 n+2+\log_2 n+1$
$+\log_2 n+1+\log_2 n+1$

$Tn = 4\log_2 n + 7$

Kompleksitas $= O(\log_2 n)$ //

2)
```
for(int i=1; i<n; i=i*2) {
    // suatu pernyataan konstan O(1)
}
```

operasi	times
i=1	1
i<n	$\log_2(n)+1$
i=i*2	$\log_2(n)$
konstan	$(\log_2(n))\times 1$

$T(n) = 1+\log_2 n+1+\log_2 n+\log_2 n$

$T(n) = 3\log_2(n)+2$

Kompleksitas $= O(\log_2 n)$ //

3)
```
int n;
scanf ("%d", &n);
while (n>0) {
    n = n/2;
}
```

operasi	times
int n	1
scanf	1
n>0	$\log_2(n)+2$
n=n/2	$\log_2(n)+1$

$T(n) = 1+1+\log_2 n+2+\log_2 n+1$

$T(n) = 2\log_2 n+5$

Kompleksitas $= O(\log_2 n)$ //

4)
```
int n;
scanf ("%d", &n);
for (int i=0; i<sqrt(n); i++){
    // nilai konstan O(1)
}
```

operasi	times
int n	1
scanf	1
i=0	1
i<sqrt(n)	$\sqrt{n}+1$
i++	$\sqrt{n}$
konstan	$(\sqrt{n})\times 1$

$T(n) = 1+1+1+\sqrt{n}+1+\sqrt{n}+\sqrt{n}$

$T(n) = 3\sqrt{n}+4$

Kompleksitas $= O(\sqrt{n})$ //

5)
```
for (int i=n; i>1; i=i--) {
    printf ("%d", &i);
}
```

operasi	times
int i=n	1
i>1	n
i=i--	n-1
printf	n-1

$T(n) = 1+n+n-1+n-1$

$T(n) = 3n-1$

Kompleksitas $= O(n)$ //

6)
```
for (int i=2; i<n; i=pow(i,2)){
    //suatu pernyataan konstan O(1)
}
```

operasi	times
i=2	1
i<n	$\log_2(n)$
i=pow(i,2)	$\log_2(\log_2(n))$
konstan	$\log_2(\log_2(n))$

$T(n) = 1 + \log_2(n) + \log_2(\log_2 n)$
$\log_2(\log_2 n)$

$T(n) = 2\log_2(\log_2(n)) + \log_2 n + 1$
Kompleksitas $= O(\log_2(\log_2(n)))$

7)
```
int n;
scanf ("%d", &n)
for(i=1; i<sqrt(n); i=i*2){
    // nilai konstan O(1)
}
```

operasi	times
int n	1
scanf	1
i=1	1
i<sqrt(n)	$\log_2(\sqrt{n})+1$
i=i*2	$\log_2(\sqrt{n})$
konstan	$(\log_2(\sqrt{n}))\times 1$

$T(n) = 1+1+1+\log_2\sqrt{n}+1$
$+\log_2\sqrt{n} + \log_2\sqrt{n}$

$T(n) = 3\log_2\sqrt{n} + 4$
Kompleksitas: $O(\log_2\sqrt{n})$

8)
```
int n;
scanf ("%d", &n);
for (i=2; i<n ; i=i*2){
    // nilai konstan O(n)
}
```

operasi	times
int n	1
scanf	1
i=2	1
i<n	$\log_2(n)$
i=i*2	$\log_2(n)-1$
konstan	$(\log_2(n)-1)\times n$

$T(n) = 1+1+1+\log_2(n)+\log_2(n)-1$
$+ n\cdot\log_2(n) - n$

$T(n) = (n+2)\log_2(n) - n + 2$
Kompleksitas $= O(n\cdot\log_2 n)$

9)
```
int n;
scanf ("%d", &n);
for (int i=0 ; i< n+n+n; i=i++){
    // nilai konstan O(n²)
}
```

operasi	times
int n	1
scanf	1
i=0	1
i<3n	3n+1
i++	3n
konstan	$(3n)\times n^2$

$T(n) = 1+1+1+3n+1+3n+3n^3$

$T(n) = 3n^3+6n+4$
Kompleksitas $= O(n^3)$

10)
```
int n;
scanf("%d", &n);
for (int i=0; i<n*n; i=i+1){
    //nilai konstan O(1)
}
```

operasi	times
int n	1
scanf	1
i=0	1
i<n²	n^2+1
i++	n^2
konstan	$(n^2)\times 1$

$T(n) = 1+1+1+n^2+1+n^2+n^2$

$T(n) = 3n^2+4$
Kompleksitas $= O(n^2)$

B. Iteratif Majemuk

1)
```
for(int i=0; i<n; i++){
    for(int j=0; j<n; j++){
        printf("*");
    }
}
```

loop	operasi	times
	i=0	1
1	i<n	n+1
	i++	n
	j=0	n×1
2	j<n	n×(n+1)
	j++	n×n
	printf	n×n

$\Sigma loop 1 = 1+n+1+n$
$= 2n+2$
$\Sigma loop 2 = n+n^2+n+n^2+n^2$
$= 3n^2+2n$

$T(n) = 3n^2+2n+2n+2$
$= 3n^2+4n+2$
Kompleksitas $= O(n^2)$

2)
```
for(int i=0; i<n; i++){
    for(int j=0; j<sqrt(n); j++){
        // nilai konstan O(1)
    }
}
```

loop	operasi	times
	i=0	1
1	i<n	n+1
	i++	n
	j=0	n×1
2	j<sqrt(n)	n×(√n+1)
	j++	n×√n
	konstan	n×√n×1

$\Sigma loop 1 = 1+n+1+n$
$= 2n+2$
$\Sigma loop 2 = n+\sqrt{n}\cdot n+n+\sqrt{n}\cdot n+\sqrt{n}\cdot n$
$= 3\sqrt{n}\cdot n+2n$
$T(n) = 2n+2+3n^{3/2}+2n$

$T(n) = 3n^{3/2}+4n+2$
Kompleksitas $= O(n^{3/2})$

3)
```
int n;
int count = 0;
scanf("%d", &n);
for(int i=0; i<n; i++){
    while (c<i) {
        sum = sum+count;
        count ++;
    }
}
```

loop	operasi	times
	int n	1
	count=0	1
	scanf	1
1	i=0	1
	i<n	n+1
	i++	n
	c<i	$\frac{(n+1)(n+2)}{2}$
2	sum+=count	$\frac{n(n+1)}{2}$
	count++	$\frac{n(n+1)}{2}$

$\Sigma loop 1 = 1+1+1+1+n+1+n$
$= 2n+5$
$\Sigma loop 2 = \frac{n^2+3n+2}{2}+\frac{n^2+n}{2}+\frac{n^2+n}{2}$
$= \frac{3n^2+5n+2}{2}$
$T(n) = \frac{4n+10}{2}+\frac{3n^2+5n+2}{2}$

$T(n) = \frac{3n^2+9n+12}{2}$
Kompleksitas $= O(n^2)$

4)
```
int n;
int c=0;
scanf("%d",&n);
for (int i=0; i<n; i++){
    c=0
    while (c<i){
        sum=sum+c;
        c++;
    }
}
```

loop	operasi	times
	int n	1
	c=0	1
1	scanf	1
	i=0	1
	i<n	n+1
	i++	n

loop	operasi	times
1	c=0	n
	c<i	$\frac{(n+1)(n+2)}{2}$
2	sum+=c	$\frac{n(n+1)}{2}$
	c++	$\frac{n(n+1)}{2}$

No
Date

$\Sigma loop 1 = 1+1+1+1+n+1+n+n$
$= 3n+5$

$\Sigma loop 2 = \frac{n^2+3n+2}{2} + \frac{n^2+n}{2} + \frac{n^2+n}{2}$
$= \frac{3n^2+5n+2}{2}$

$T(n) = \frac{6n+10}{2} + \frac{3n^2+5n+2}{2}$

$T(n) = \frac{3n^2+11n+12}{2}$

Kompleksitas $= O(n^2)$

5)
```
scanf("%d", &n);
For(int i=0; i<n=pow(n,3); i++){
    if(i<n = pow(n,3)){
        printf("%d", i);
    }
}
```

loop	operasi	times
	scanf	1
	i=0	1
1	$i<n^3$	n^3+1
	i++	n^3
	$i<n^3$	n^3
	printf	n^3

$T(n) = 1+1+n^3+1+n^3+n^3+n^3$

$T(n) = 4n^3+3$

Kompleksitas $= O(n^3)$

6)
```
int n;
int count =0;
scanf("%d", &n);
For(int i=0; i<n; i++){
    For(int j=0; j<n*n; j++){
        For(int k=n; k>0; k--){
            count = count+k;
        }
}}
```

loop	operasi	times
	int n	1
	count=0	1
1	scanf	1
	i=0	1
	i<n	n+1
	i++	n
2	j=0	n×1

loop	operasi	times
2	$j<n^2$	$n\times(n^2+1)$
	j++	$n\times n^2$
3	k=n	$n\cdot n^2\cdot 1$
	k>0	$n\cdot n^2\cdot(n+1)$
	k--	$n\cdot n^2\cdot n$
	count+=k	$n\cdot n^2\cdot n$

$\Sigma loop 1 = 1+1+1+1+n+1+n$
$= 2n+5$

$\Sigma loop 2 = n+n^3+n+n^3$
$= 2n^3+2n$

$\Sigma loop 3 = n^3+n^4+n^3+n^4+n^4$
$= 3n^4+2n^3$

$T(n) = 2n+5+2n^3+2n+3n^4+2n^3$

$T(n) = 3n^4+4n^3+4n+5$

Kompleksitas $= O(n^4)$

7)
```
int count=0;
For(int i=n/2; i<=n; i++){
    For(int j=1; j+n/2<=n; j++){
        For(int k=1; k<=n; k=k*2){
            count ++;
        }
    }
}
```

loop	operasi	times
	count=0	1
1	i=n/2	1
	i<=n	$\frac{n}{2}+2$
	i++	$\frac{n}{2}+1$
2	j=1	$(\frac{n}{2}+1)\times 1$
	j+n/2<=n	$(\frac{n}{2}+1)(\frac{n}{2}+1)$

loop	operasi	times
2	j++	$(\frac{n}{2}+1)(\frac{n}{2})$
3	k=1	$(\frac{n}{2}+1)(\frac{n}{2})\times 1$
	k<=n	$(\frac{n}{2}+1)(\frac{n}{2})(\log_2 n+2)$
	k=k*2	$(\frac{n}{2}+1)(\frac{n}{2})(\log_2 n+1)$
	count++	$(\frac{n}{2}+1)(\frac{n}{2})(\log_2 n+1)$

No ....
Date

Σ loop 1 $= 1+1+\frac{n}{2}+2+\frac{n}{2}+1$
$= n+5$

Σ loop 2 $= \frac{n}{2}+1+\frac{n^2}{4}+\frac{n}{2}+\frac{n}{2}+1+\frac{n^2}{4}+\frac{n}{2}$
$= n^2/2+2n+1$

Σ loop 3 $= \frac{n^2}{4}+\frac{n}{2}+\left(\frac{n^2}{4}+\frac{n}{2}\right)\times(3\log_2 n+4)$
$= \frac{n^2}{4}+\frac{n}{2}+\frac{3}{2}n^2\log_2 n+n^2+\frac{3}{2}n\log_2 n+2n$
$= \frac{5n^2}{4}+\frac{5n}{2}+\frac{3}{2}n^2\log_2 n+\frac{3}{2}n\log_2 n$

$T(n) = n+5+\frac{n^2}{2}+2n+1+\frac{5n^2}{4}+\frac{5n}{4}+\frac{3}{2}n^2\log_2 n+\frac{3}{2}n\log_2 n$

$T(n) = \frac{3}{2}n^2\log_2 n+\frac{3}{2}n\log_2 n+\frac{7n^2}{4}+\frac{11n}{2}+6$

Komplerksitas $= O(n^2\log_2 n)$

8)
```
for(int i=0; i<=sqrt(n); i++){
    for(int j=n; j>=1; j=j--){
        for(int k=1; k<=n; k=k*2){
            printf("*");
        }
    }
}
```

loop	operasi	times
	i=0	1
1	i<=sqrt(n)	$\sqrt{n}+2$
	i++	$\sqrt{n}+1$
	j=n	$(\sqrt{n}+1)\times 1$
2	j>=1	$(\sqrt{n}+1)\times(n+1)$
	j--	$(\sqrt{n}+1)\cdot n$
	k=1	$(\sqrt{n}+1)\cdot n\cdot 1$
3	k<=n	$(\sqrt{n}+1)\cdot n\cdot(\log_2 n+2)$
	k=k*2	$(\sqrt{n}+1)\cdot n\,(\log_2 n+1)$
	printf	$(\sqrt{n}+1)\cdot n\,(\log_2 n+1)$

Σ loop 1 $= 1+\sqrt{n}+2+\sqrt{n}+1$
$= 2\sqrt{n}+4$

Σ loop 2 $= \sqrt{n}+1+\sqrt{n}\cdot n+\sqrt{n}+n+1+\sqrt{n}\,n+n$
$= 2n^{3/2}+2n+2\sqrt{n}+2$

Σ loop 3 $= (n^{3/2}+n)(5+3\log_2 n)$
$= 3n^{3/2}\log_2 n+5n^{3/2}+3n\log_2 n+5n$

$T(n) = 2\sqrt{n}+4+2^{3/2}+2n+2\sqrt{n}+2+3n^{3/2}\log_2 n+5n^{3/2}+3n\log_2 n+5n$

$T(n) = 3n^{3/2}\log_2 n+3n\log_2 n+7n^{3/2}+7n+4\sqrt{n}+6$

Komplerksitas $= O(n^{3/2}\cdot\log_2 n)$

9)
```
int arr[M] = {//array M element}
int top = arr[0]
int best = arr[0]
for(int i=0; i<M; i++){
    top=max(arr[i], top+arr[i]);
    best = max(top, best);
}
printf("%d", best);
```

operasi	times
int arr[M]	m
top=arr[0]	1
best=arr[0]	1
i=0	1
i<M	m+1
i++	m
top = max(arr[i], top+arr[i])	m
best=max(top, best)	m
printf	1

$T(n) = m+1+1+1+m+1+m+m+m+1$

$T(n) = 5m+5$

Komplerksitas $= O(m)$

No

Date

10)
```
scanf ("%d",&n);
  For (int i=0; i<n*n; i++){
      For (int j=m/4; j<m/2; j*=2)
          printf("I");
      For (int j=0; j<2*m; j+=2){
          printf("B");
            For (int k=0; j<2*m; k+=2)
                printf("p");
      }
  }

For (int i=1; i/2<n*m; i*=2)
    printf("B");
```

loop	operasi	times	
	scanf	1	$\Sigma loop 1a = 1+1+n^2+1+n^2$
1a	$i=0$	1	$= 2n^2+3$
	$i<n^2$	n^2+1	
	$i++$	n^2	
	$j=m/4$	$n^2 \cdot 1$	$\Sigma loop 2a: n^2+2n^2+n^2+n^2$
2a	$j<m/2$	$n^2 \cdot 2$	$= 5n^2$
	$j*=2$	$n^2 \cdot 1$	
	printf	$n^2 \cdot 1$	
	$j=0$	$n^2 \cdot 1$	$\Sigma loop 2b$
2b	$j<2*m$	$n^2(m+1)$	$= n^2+n^2m+n^2+n^2m+n^2m$
	$j+=2$	$n^2 \cdot m$	$= 3n^2m+2n^2$
	printf	$n^2 \cdot m$	
	$k=0$	$n^2m \cdot 1$	$\Sigma loop 3$
3	$j<2*m$	$n^2m \cdot (m+1)$	$= n^2m+n^2m^2+n^2m+n^2m^2$
	$k+=2$	$n^2m \cdot m$	$+n^2m^2$
	printf	$n^2m \cdot m$	$= 3n^2m^2+2n^2m$
	$i=1$	1	$\Sigma loop 1b$
1b	$i/2<n*m$	$\log_2 nm+2$	$= 1+3\log_2 nm+4$
	$i*=2$	$\log_2 nm+1$	$= 3\log_2 nm+5$
	printf	$\log_2 nm+1$	

$T(n) = 2n^2+3+5n^2+3n^2m+2n^2+3n^2m^2$
$+2n^2m+3\log_2 nm+5$

$$\boxed{\begin{array}{l} T(n) = 3n^2m^2+5n^2m+9n^2+3\log_2 nm+8 \\ \text{kompleksitas} = O(n^2m^2) \end{array}}$$