20.7. TIME COMPLEXITY CALCULATION

NESTED FOR LOOP (EG-6).

$//outer\ loop\ executed\ \ n\ times$

$for(i = 1; i \leq n; i++)\{$

$\quad //inner\ loop\ executes\ n\ times$

$\quad for(j = 1; j \leq n - 1; j++)\{$

$\qquad k = k + 1\ ;\ //\ constant\ time.$

$\}$

SOLUTION:

$1. Inner\ most\ loop's\ statement \Rightarrow k = k + 1\ which\ runs\ at$

$O(1)\ time.$

$2. No. of\ inputs\ in\ outer\ for\ loop\ takes\ 1\ to\ n\ times.$

$lets\ see\ the\ inner\ loop\ and\ runtime\ of$

$inner\ loop's\ statement.$

$f(1) \leq c \times n \Rightarrow when\ i = 1$

$f(1) \leq c \times n \Rightarrow when\ j = 1$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$f(2) \leq c \times n \Rightarrow when\ j = 2$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$f(3) \leq c \times n \Rightarrow when\ j = 3$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$\ldots \ldots \ldots.$

$f(n-1) \leq c \times n \Rightarrow when\ j = n - 1$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$i.e. when\ i = 1\ , the\ inner\ most\ loop\ statement$

$$run\ (1 + 1 + 1 + 1 + \cdots + n - 1) = n - 1\ times$$

$f(2) \leq c \times n \Rightarrow when\ i = 2$

$f(1) \leq c \times n \Rightarrow when\ j = 1$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$f(2) \leq c \times n \Rightarrow when\ j = 2$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$f(3) \leq c \times n \Rightarrow when\ j = 3$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$\ldots \ldots \ldots.$

$f(n-1) \leq c \times n \Rightarrow when\ j = n - 1$

$c = c + 1\ runs\ 1\ unit\ of\ time.$

$i.e. when\ i = 2\ , the\ inner\ most\ loop\ statement$

$$run\ (1 + 1 + 1 + 1 + \cdots + n - 1) = n - 1\ times$$

.....

$$f(n) \leq c \times n \Longrightarrow when\ i = n$$

$$f(1) \leq c \times n \Longrightarrow when\ j = 1$$

$$c = c + 1\ runs\ 1\ unit\ of\ time.$$

$$f(2) \leq c \times n \Longrightarrow when\ j = 2$$

$$c = c + 1\ runs\ 1\ unit\ of\ time.$$

$$f(3) \leq c \times n \Longrightarrow when\ j = 3$$

$$c = c + 1\ runs\ 1\ unit\ of\ time.$$

… … ….

$$f(n-1) \leq c \times n \Longrightarrow when\ j = n - 1$$

$$c = c + 1\ runs\ 1\ unit\ of\ time.$$

$$i.e. when\ i = n\ , the\ inner\ most\ loop\ statement$$

$$run\ (1 + 1 + 1 + 1 + \cdots + n - 1) = n - 1\ times$$

$$Therefore\ printing\ the\ inner\ most\ statement (k = k + 1)$$

$$n\ times\ n - 1 = (n - 1) + (\ n - 1) + \cdots + (n - 1) = n(n - 1)$$

$$= n^2 - n\ times\ , hence\ O(n^2 - n) = O(n^2)$$

****