



(修練) :
가
(Dijkstra's "A Discipline of Programming" :
The Eleventh Lecture, The Formal Treatment of
Some Small Examples -)

dkim@sungshin.ac.kr



1) $a = 0$ 가 가
 $n(a = 0)$;
 R : "a := a + 1" 2)
 R : $a^2 \leq n$ and $(a+1)^2 > n$
 가
 (conjunction) P : $(P \text{ and non } BB) = R$
 P : $a^2 \leq n$

1) [1] 2) (massaging operation)
 8

```

if n = 0
  a := 0 {P가};
  do (a+1)^2 ≤ n a := a + 1
    {P가}
  od
  {R}
fi {R}

```

가
 , 가
 :
 t n - a^2
 ;
 : n
 . R
 R
 (b-
)
 P :

P: a^2 ≤ n and b^2 > n and 0 < a < b

P
 가
 (P and (a + 1 = b)) R

```

if n = 0 ...
  :
a, b := 0, n + 1 {P가};
do a + 1 = b
  P b - a
od
{R}

```

가 b - a 가

d .
 b a 가
 가 , a b 가
 : a가 b가
 3) b ,
 a 가 .

```

a, b := 0, n + 1 {P가};
do a + 1 = b
  d := ...
  {d 가 P};
  if ... a := a + d {P가}
  | ... b := b - d {P가}
  fi {P가}
od {R}

```

$$\text{wp}("a := a + d", P) = ((a + d)^2 \leq n \text{ and } b^2 > n)$$

, P가 4)
 5) 가
 ; 가
 :

```

a, b := 0, n + 1;
do a + 1 = b
  d := ...;
  if (a + d)^2 ≤ n a := a + d

```

3) b - a 가
 4) b^2 > n 가
 5) (a + d)^2 ≤ n 가

```

    | (b - d)2 > n    b := b - d
  fi {P가 }
od {R }

d
가 b - a( b - a - 1)
t
d가 d > 0
, (d
)
,6) 가
가
(a + d)2 > n 가 (b - d)2
> n ;

```

$a + d \leq b - d$

$2 * d \leq b - a$

.7) d

(lower bound) (upper bound)

$d := 1$

d가

:

```

a, b := 0, n + 1;
do a + 1 ≤ b
  d := (b - a) div 2;
  if (a + d)2 ≤ n    a := a + d

```

6) 가 가 (n 0

)

7)

```

    | (b - d)2 > n    b := b - d
  fi
od

n div 2 2 | n    n / 2, 2 | (n
- 1) (n - 1) / 2
div , d
b - a가 가
. c := b - a
b
,
P: a2 ≤ n and (a + c)2 > n
and (E i: i 0: c = 2i)

```

(c d) 가 :

a, c := 0, 1;

do c² ≤ n c := 2 * c od;

do c = 1

c := c / 2;

if (a + c)² ≤ n a := a + c

| (a + c)² > n skip

fi

od

[2]

가 3

가

do (a + c)² ≤ n a := a + c od

0

$$r < d \quad r < d \text{ and } (r + d) < d$$

,

$$. (\quad .)$$

(redundancy)

가 ()
가
가
; 가 , 2 ()

가 가
가 ! (, (a, c)-
(p, q, r)-

((semi-)multiplicative operation) 가
((準)-乘算的)

(boolean expression)
(a, c)-

가 , a c
(in abstracto)

가
8) 가 (h
가):10)
: a
c 가

```
p, q, r := 0, 1, n;  
do q = n q := q * 4 od;  
do q = 1  
  q := q / 4; h := p + q; p := p / 2  
  {h = 2 * p + q};  
  if r = h p, r := p + q, r - h  
  | r < h skip  
fi  
od {p} a 가 }
```

(identity)

.9) a c
가 p, q, r
:
p = a * c
q = c^2
r = n - a^2

10) !

, (a, c)-
(p, q, r)- 가

(correctness)

, p, q, r
: p, q, r

8) a c
9)

가
(concrete variable)

가 !
()

가 가

(solution)

(massage)'

h

가

a b



1981 ~ 1985

()

1985 ~ 1987 KAIST

()

1987 ~ 1992 KAIST

()

1992 KAIST

1992 ~

1997 ~ 1998

(representational abstraction)'

[1] Dijkstra, E. W., *A Discipline of Programming*, Prentice Hall, Englewood Cliffs, NJ, 1976.

[2] (修練): 가
- II",
18, 2,
, 2004 .