Insertion sort

- Pseudo code:

```
for i = 2 to n
    key = A[i]
    j = i-1
    while j > 0 and A[j] > key
        A[j+1] = A[j]
        j -= 1
    A[j+1] = key
```

Dry run:

0	1	2	3	4
3	4	1	2	0

$A = [3, 4, 1, 2, 0]$

- for $i = 2$ to 5

$i = 2$

$key = A[2] = 1$

$j = i-1 = 2-1 = 1$

while $1 > 0$ and $A[1] > 1$ $\{4 > 1\}$ (True)

$A[1+1] = A[1]$

$\Rightarrow 1 = 4$ ∴ $A = [3, 4, 4, 2, 0]$

$j -= 1 \Rightarrow j = 0$

while $0 > 0$ and $A[0] > 1$ $\{3 > 1\}$ (False)

$A[0+1] = key \Rightarrow A[1] = 1 \Rightarrow 4 = 1$

0	1	2	3	4
3	1	4	2	0

∴ $A = [3, 1, 4, 2, 0]$

- for $i = 3$ to 5

$i = 3$

$Key = A[3] = 2$

$j = i-1 = 3-1 = 2$

while $2 > 0$ and $A[2] > 2$ $\{4 > 2\}$ (True)

$A[2+1] = A[2] \Rightarrow A[3] = A[2] \Rightarrow 2 = 4$

$j -= 1 \Rightarrow j = 1$

0	1	2	3	4
3	1	4	4	0

∴ $A = [3, 1, 4, 4, 0]$

while $1 > 0$ and $A[1] > 2$ $\{1 > 2\}$ (False)

$A[j+1] = key \Rightarrow A[1+1] = 2 \Rightarrow 4 = 2$

0	1	2	3	4
3	1	2	4	0

∴ $A = [3, 1, 2, 4, 0]$

- for $i = 4$ to 5

  $i = 4$
  $key = A[4] = 0$
  $j = i - 1 = 4 - 1 = 3$

  while $3 > 0$ and $A[3] > 0$ { $4 > 0$ } True
  $A[3+1] = A[3] \Rightarrow 0 = 4$ ∴ $A = [3, 1, 2, 4, 4]$ (0 1 2 3 4)
  $j = 1 \Rightarrow j = 2$

  while $2 > 0$ and $A[2] > 0$ { $2 > 0$ } (True)
  $A[2+1] = A[2] \Rightarrow 4 = 2$
  $j = 1 \Rightarrow j = 1$ ∴ $A = [3, 1, 2, 2, 4]$ (0 1 2 3 4)

  while $1 > 0$ and $A[1] > 0$ { $1 > 0$ } (True)
  $A[1+1] = A[1] \Rightarrow 2 = 1$
  $j = 1 \Rightarrow j = 0$ ∴ $A = [3, 1, 1, 2, 4]$ (0 1 2 3 4)

  $A[1] = key$
  $\Rightarrow A[1] = 0 \Rightarrow$ ∴ $A = [3, 0, 1, 2, 4]$ (0 1 2 3 4)

* Making $j >= 0$ would successfully sort the whole array

Exercises

2.1-1 $A = [31, 41, 59, 26, 41, 58]$

(a) 0 1 2 3 4 5
31 41 59 26 41 58

(b) 0 1 2 3 4 5
26 31 41 59 41 58

(c) 0 1 2 3 4 5
26 31 41 41 59 58

(d) 0 1 2 3 4 5
26 31 41 41 58 59

2.1-3 Pseudocode:

$A = [3, 4, 1, 2, 0]$, $n = 5$

```
for i = 1 to n
    key = A[i]
    j = i - 1

    while j >= 0 and A[j] < key:
        A[j+1] = A[j]
        j = j - 1
```

A[j+1] = key

2.1-4 Pseudocode:

```
Linear search:
x = an element entered by the user
flag = false
for i = 0 to n-1
    if A[i] == x
        flag = true

if flag:
    print ("Element exists")
else:
    print ("Element doesn't exist")
```