$n = 8$개인 입력 데이터 $a(1) = 44$, $a(2) = 55$, $a(3) = 22$, $a(4) = 88$, $a(5) = 66$, $a(6) = 11$, $a(7) = 77$, $a(8) = 44$에 대하여 다음 4가지 bubble sort 알고리즘들과 4가지 insertion sort 알고리즘들을 적용했을때 각 알고리즘들의 i^{th} step의 결과를 빈칸에 쓰시오.

1 Bubble Sort Algorithm

1. **bubblesort-1 algorithm**

```
procedure bubblesort_1 (a(1), ..., a(n): real number with n >=2)
for i := 1 to n-1
  for j := 1 to n-i
    if a(j) > a(j+1) then interchange a(j) and a(j+1)
{a(1), ..., a(n) is in increasing order}
```

	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$
44	44	22	22	22	11	11	11
55	22	44	44	11	22	22	22
22	55	55	11	44	33	33	33
88	66	11	55	33	44	44	44
66	11	66	33	55	55	55	55
11	77	33	66	66	66	66	66
77	33	77	77	77	77	77	77
33	88	88	88	88	88	88	88

2. **bubblesort-2 algorithm**

```
procedure bubblesort_2 (a(1), ..., a(n): real number with n >=2)
for i := 1 to n-1
  for j := 1 to n-i
    if a(j) < a(j+1) then interchange a(j) and a(j+1)
{a(1), ..., a(n) is in decreasing order}
```

	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$	$i=7$
44	55	55	88	88	88	88	88
55	44	88	66	66	77	77	77
22	88	66	55	77	66	66	66
88	66	44	77	55	55	55	55
66	22	77	44	44	44	44	44
11	77	33	33	33	33	33	33
77	33	22	22	22	22	22	22
33	11	11	11	11	11	11	11

3. **bubblesort-3 algorithm**

```
procedure bubblesort_3 (a(1), ..., a(n): real number with n >=2)
for i := 1 to n−1
  for j := 8 downto i+1
    if a(j) < a(j−1) then interchange a(j) and a(j−1)
{a(1), ..., a(n) is in increasing order}
```

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$
44	11	11	11	11	11	11	11
55	44	22	22	22	22	22	22
22	55	44	33	33	33	33	33
88	22	55	44	44	44	44	44
66	88	33	55	55	55	55	55
11	66	88	66	66	66	66	66
77	33	66	88	77	77	77	77
33	77	77	77	88	88	88	88

4. **bubblesort-4 algorithm**

```
procedure bubblesort_4 (a(1), ..., a(n): real number with n >=2)
for i := 1 to n−1
  for j := 8 downto i+1
    if a(j) > a(j−1) then interchange a(j) and a(j−1)
{a(1), ..., a(n) is in decreasing order}
```

	$i = 1$	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$
44	88	88	88	88	88	88	88
55	44	77	77	77	77	77	77
22	55	44	66	66	66	66	66
88	22	55	44	55	55	55	55
66	77	22	55	44	44	44	44
11	66	66	22	33	33	33	33
77	11	33	33	22	22	22	22
33	33	11	11	11	11	11	11

2 Insertion Sort Algorithm

1. **insertsort-1 algorithm**

```
procedure insertsort_1 (a(1), ..., a(n): real number with n >=2)
i := 2
while i <= n
    j := i
    while ((j >= 2) and (A(j-1) > A(j)))
        swap A(j) and A(j-1)
        j := j - 1
    end
    i := i + 1
end while
{a(1), ..., a(n) is in increasing order}
```

	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$
44	44	22	22	22	11	11	11
55	55	44	44	44	22	22	22
22	22	55	55	55	44	44	33
88	88	88	88	66	55	55	44
66	66	66	66	88	66	66	55
11	11	11	11	11	88	77	66
77	77	77	77	77	77	88	77
33	33	33	33	33	33	33	88

2. **insertsort-2 algorithm**

```
procedure insertsort_1 (a(1), ..., a(n): real number with n >=2)
i := 2
while i <= n
    j := i
    while ((j >= 2) and (A(j-1) < A(j)))
        swap A(j) and A(j-1)
        j := j - 1
    end
    i := i + 1
end while
{a(1), ..., a(n) is in decreasing order}
```

	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$
44	55	55	88	88	88	88	88
55	44	44	55	66	66	77	77
22	22	22	44	55	55	66	66
88	88	88	22	44	44	55	55
66	66	66	66	22	22	44	44
11	11	11	11	11	11	22	33
77	77	77	77	77	77	11	22
33	33	33	33	33	33	33	11

3. **insertsort-3 algorithm**

```
procedure insertsort_3 (a(1), ..., a(n): real number with n >=2)
i := 2
while i <= n
    j := n - i + 1
    while ((j < n) and (A(j) > A(j+1)))
        swap A(j) and A(j+1)
        j := j + 1
    end
    i := i + 1
end while
{a(1), ..., a(n) is in increasing order}
```

	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$
44	44	44	44	44	44	44	11
55	55	55	55	55	55	11	22
22	22	22	22	22	11	22	33
88	88	88	88	11	22	33	44
66	66	66	11	33	33	55	55
11	11	11	33	66	66	66	66
77	33	33	66	77	77	77	77
33	77	77	77	88	88	88	88

4. **insertsort-4 algorithm**

```
procedure insertsort_4 (a(1), ..., a(n): real number with n >=2)
i := 2
while i <= n
    j := n - i + 1
    while ((j < n) and (A(j) < A(j+1)))
        swap A(j) and A(j+1)
        j := j + 1
    end
    i := i + 1
end while
{a(1), ..., a(n) is in decreasing order}
```

	$i = 2$	$i = 3$	$i = 4$	$i = 5$	$i = 6$	$i = 7$	$i = 8$
44	44	44	44	44	44	44	88
55	55	55	55	55	55	88	77
22	22	22	22	22	88	77	66
88	88	88	88	88	77	66	55
66	66	66	77	77	66	55	44
11	11	77	66	66	33	33	33
77	77	33	33	33	22	22	22
33	33	11	11	11	11	11	11

```
   https://wordpress.com/posts/cprogramex.wordpress.com

::::::::::::::
bub-sort-1.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  for (i=0; i<n-1; i++) {
    for (j=0; j<n-i-1; j++) {
      if (a[j] > a[j+1]) {
        tmp = a[j];
        a[j] = a[j+1];
        a[j+1] = tmp;
      }
      printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");
  }


}
::::::::::::::
bub-sort-2.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  for (i=0; i<n-1; i++) {
    for (j=0; j<n-i-1; j++) {
      if (a[j] < a[j+1]) {
        tmp = a[j];
        a[j] = a[j+1];
        a[j+1] = tmp;
      }
      printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
```

```
      printf(" %3d", a[j]);
    printf("\n");
  }


}
::::::::::::::
bub-sort-3.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  for (i=0; i<n-1; i++) {
    for (j=n-1; j>i; j--) {
      if (a[j] < a[j-1]) {
        tmp = a[j];
        a[j] = a[j-1];
        a[j-1] = tmp;
      }
      printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");
  }


}
::::::::::::::
bub-sort-4.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  for (i=0; i<n-1; i++) {
    for (j=n-1; j>i; j--) {
      if (a[j] > a[j-1]) {
        tmp = a[j];
        a[j] = a[j-1];
        a[j-1] = tmp;
      }
      printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");
  }


}
```

https://wordpress.com/posts/cprogramex.wordpress.com

```
::::::::::::::
bsort-1.out
::::::::::::::
i=0 j=0
i=0 j=1
i=0 j=2
i=0 j=3
i=0 j=4
i=0 j=5
i=0 j=6
  44  22  55  66  11  77  33  88
i=1 j=0
i=1 j=1
i=1 j=2
i=1 j=3
i=1 j=4
i=1 j=5
  22  44  55  11  66  33  77  88
i=2 j=0
i=2 j=1
i=2 j=2
i=2 j=3
i=2 j=4
  22  44  11  55  33  66  77  88
i=3 j=0
i=3 j=1
i=3 j=2
i=3 j=3
  22  11  44  33  55  66  77  88
i=4 j=0
i=4 j=1
i=4 j=2
  11  22  33  44  55  66  77  88
i=5 j=0
i=5 j=1
  11  22  33  44  55  66  77  88
i=6 j=0
  11  22  33  44  55  66  77  88
::::::::::::::
bsort-2.out
::::::::::::::
i=0 j=0
i=0 j=1
i=0 j=2
i=0 j=3
i=0 j=4
i=0 j=5
i=0 j=6
```

```
  55  44  88  66  22  77  33  11
i=1 j=0
i=1 j=1
i=1 j=2
i=1 j=3
i=1 j=4
i=1 j=5
  55  88  66  44  77  33  22  11
i=2 j=0
i=2 j=1
i=2 j=2
i=2 j=3
i=2 j=4
  88  66  55  77  44  33  22  11
i=3 j=0
i=3 j=1
i=3 j=2
i=3 j=3
  88  66  77  55  44  33  22  11
i=4 j=0
i=4 j=1
i=4 j=2
  88  77  66  55  44  33  22  11
i=5 j=0
i=5 j=1
  88  77  66  55  44  33  22  11
i=6 j=0
  88  77  66  55  44  33  22  11
::::::::::::::
bsort-3.out
::::::::::::::
i=0 j=7
i=0 j=6
i=0 j=5
i=0 j=4
i=0 j=3
i=0 j=2
i=0 j=1
  11  44  55  22  88  66  33  77
i=1 j=7
i=1 j=6
i=1 j=5
i=1 j=4
i=1 j=3
i=1 j=2
  11  22  44  55  33  88  66  77
i=2 j=7
i=2 j=6
i=2 j=5
i=2 j=4
```

```
i=2 j=3
  11  22  33  44  55  66  88  77
i=3 j=7
i=3 j=6
i=3 j=5
i=3 j=4
  11  22  33  44  55  66  77  88
i=4 j=7
i=4 j=6
i=4 j=5
  11  22  33  44  55  66  77  88
i=5 j=7
i=5 j=6
  11  22  33  44  55  66  77  88
i=6 j=7
  11  22  33  44  55  66  77  88
::::::::::::::
bsort-4.out
::::::::::::::
i=0 j=7
i=0 j=6
i=0 j=5
i=0 j=4
i=0 j=3
i=0 j=2
i=0 j=1
  88  44  55  22  77  66  11  33
i=1 j=7
i=1 j=6
i=1 j=5
i=1 j=4
i=1 j=3
i=1 j=2
  88  77  44  55  22  66  33  11
i=2 j=7
i=2 j=6
i=2 j=5
i=2 j=4
i=2 j=3
  88  77  66  44  55  22  33  11
i=3 j=7
i=3 j=6
i=3 j=5
i=3 j=4
  88  77  66  55  44  33  22  11
i=4 j=7
i=4 j=6
i=4 j=5
  88  77  66  55  44  33  22  11
i=5 j=7
```

```
i=5 j=6
  88  77  66  55  44  33  22  11
i=6 j=7
  88  77  66  55  44  33  22  11
```

```
   https://wordpress.com/posts/cprogramex.wordpress.com

::::::::::::::
ins-sort-1.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  i = 1;

  while (i < n) {
    j = i;
    printf("i=%d j=%d \n", i, j);

    while ((j>0) && (a[j-1] > a[j])) {
      tmp = a[j];
      a[j] = a[j-1];
      a[j-1] = tmp;

      j--;
      if (j>0) printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");

    i++;
  }
}

::::::::::::::
ins-sort-2.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  i = 1;

  while (i < n) {
    j = i;
    printf("i=%d j=%d \n", i, j);

    while ((j>0) && (a[j-1] < a[j])) {
      tmp = a[j];
      a[j] = a[j-1];
      a[j-1] = tmp;

      j--;
      if (j>0) printf("i=%d j=%d \n", i, j);

    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");

    i++;
  }
}

::::::::::::::
ins-sort-3.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  i = 1;

  while (i < n) {
    j = n-1-i;
    printf("i=%d j=%d \n", i, j);

    while ((j<n-1) && (a[j] > a[j+1])) {
      tmp = a[j];
      a[j] = a[j+1];
      a[j+1] = tmp;


      j++;
      if (j<n-1) printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");

    i++;
  }
}

::::::::::::::
ins-sort-4.c
::::::::::::::
#include <stdio.h>

#define n 8

int main(void) {
  int a[8] = { 44, 55, 22, 88, 66, 11, 77, 33 };
  int i, j, tmp;

  i = 1;
  while (i < n) {
    j = n-1-i;
    printf("i=%d j=%d \n", i, j);

    while ((j<n-1) && (a[j] < a[j+1])) {
      tmp = a[j];
      a[j] = a[j+1];
      a[j+1] = tmp;

      j++;
      if (j<n-1) printf("i=%d j=%d \n", i, j);
    }
    for (j=0; j<n; j++)
      printf(" %3d", a[j]);
    printf("\n");

    i++;
  }
}
```

https://wordpress.com/posts/cprogramex.wordpress.com

```
::::::::::::::
isort-1.out
::::::::::::::
i=1 j=1
  44  55  22  88  66  11  77  33
i=2 j=2
i=2 j=1
  22  44  55  88  66  11  77  33
i=3 j=3
  22  44  55  88  66  11  77  33
i=4 j=4
i=4 j=3
  22  44  55  66  88  11  77  33
i=5 j=5
i=5 j=4
i=5 j=3
i=5 j=2
i=5 j=1
  11  22  44  55  66  88  77  33
i=6 j=6
i=6 j=5
  11  22  44  55  66  77  88  33
i=7 j=7
i=7 j=6
i=7 j=5
i=7 j=4
i=7 j=3
i=7 j=2
  11  22  33  44  55  66  77  88
::::::::::::::
isort-2.out
::::::::::::::
i=1 j=1
  55  44  22  88  66  11  77  33
i=2 j=2
  55  44  22  88  66  11  77  33
i=3 j=3
i=3 j=2
i=3 j=1
  88  55  44  22  66  11  77  33
i=4 j=4
i=4 j=3
i=4 j=2
i=4 j=1
  88  66  55  44  22  11  77  33
i=5 j=5
  88  66  55  44  22  11  77  33
i=6 j=6
```

```
i=6 j=5
i=6 j=4
i=6 j=3
i=6 j=2
i=6 j=1
  88  77  66  55  44  22  11  33
i=7 j=7
i=7 j=6
i=7 j=5
  88  77  66  55  44  33  22  11
::::::::::::::
isort-3.out
::::::::::::::
i=1 j=6
  44  55  22  88  66  11  33  77
i=2 j=5
  44  55  22  88  66  11  33  77
i=3 j=4
i=3 j=5
i=3 j=6
  44  55  22  88  11  33  66  77
i=4 j=3
i=4 j=4
i=4 j=5
i=4 j=6
  44  55  22  11  33  66  77  88
i=5 j=2
i=5 j=3
  44  55  11  22  33  66  77  88
i=6 j=1
i=6 j=2
i=6 j=3
i=6 j=4
  44  11  22  33  55  66  77  88
i=7 j=0
i=7 j=1
i=7 j=2
i=7 j=3
  11  22  33  44  55  66  77  88
::::::::::::::
isort-4.out
::::::::::::::
i=1 j=6
  44  55  22  88  66  11  77  33
i=2 j=5
i=2 j=6
  44  55  22  88  66  77  33  11
i=3 j=4
i=3 j=5
  44  55  22  88  77  66  33  11
```

```
i=4 j=3
  44  55  22  88  77  66  33  11
i=5 j=2
i=5 j=3
i=5 j=4
i=5 j=5
i=5 j=6
  44  55  88  77  66  33  22  11
i=6 j=1
i=6 j=2
i=6 j=3
i=6 j=4
  44  88  77  66  55  33  22  11
i=7 j=0
i=7 j=1
i=7 j=2
i=7 j=3
i=7 j=4
  88  77  66  55  44  33  22  11
```