```c
#include <stdio.h>
#define MAX 10

int main( void )
{

/* initialize array a with initializer list */
int a[ MAX ] = { 10, 9, 8, 7, 6, 5, 4, 3, 2, 1};
int i; /* loop counter */
int pass; /* loop counter */
int hold; /* temporary variable for swapping */
int swap; /* flag to break loop if elements are sorted */
printf( "Data items in original order\n" );

/* display original, unsorted array */
for ( i = 0; i < MAX; i++ ) {
printf( "%4d", a[ i ] );
} /* end for */

printf( "\n\n" );

/* begin sorting the array */
for ( pass = 1; pass < MAX; pass++ ) {
swap = 0;

/* traverse and compare unsorted part of array */
for ( i = 0; i < MAX - pass; i++ ) {

/* compare adjacent array elements */
if ( a[ i ] > a[ i + 1 ] ) {
swap = 1; /* raise flag if any elements are swapped */
hold = a[ i ];
a[ i ] = a[ i + 1 ];
a[ i + 1 ] = hold;
} /* end if */

} /* end for */
```

```c
printf( "After Pass %d: ", pass );

/* display array after each pass */
for ( i = 0; i <= MAX-pass; i++ ) {
printf( " %d", a[ i ] );
} /* end for */

printf( "\n" );

/* break loop if array is sorted */
if ( !swap ) {
break;
} /* end if */

} /* end for */

printf( "\nData items in ascending order\n" );

/* display array in sorted order */
for ( i = 0; i < 10; i++ ) {
printf( "%4d", a[ i ] );
} /* end for */

printf( "\n" );
return 0; /* indicate successful termination */
} /* end main */
```

```
C:\Windows\system32\cmd.exe

Data items in original order
  10   9   8   7   6   5   4   3   2   1

After Pass 1:  9 8 7 6 5 4 3 2 1 10
After Pass 2:  8 7 6 5 4 3 2 1 9
After Pass 3:  7 6 5 4 3 2 1 8
After Pass 4:  6 5 4 3 2 1 7
After Pass 5:  5 4 3 2 1 6
After Pass 6:  4 3 2 1 5
After Pass 7:  3 2 1 4
After Pass 8:  2 1 3
After Pass 9:  1 2

Data items in ascending order
   1   2   3   4   5   6   7   8   9  10
請按任意鍵繼續 . . .
```