

```
#include <stdio.h>

/* -----
   return the index of the correct position of element in the sorted
   sublist [0 .. stop]
*/
int sort_pos(int list[], int element, int stop)
{
    int i = 0;

    while ( element > list[i] && i < stop )
        i++;
    return i;
}

/* -----
   insert element in the sorted sublist [start .. stop]
   by shifting the following elements one index up
*/
void insert(int list[], int element, int start, int stop)
{
    int i;

    for (i=stop; i>start; i--)
        list[i] = list[i-1];
    list[start] = element;
}

/* -----
   insertion sort routine

   sort a list of processor word-sized elements (int) of size 'size'
*/
void ins_sort(int list[], int size)
{
    int i;          /* list index: partitions list into sorted [0 .. i-1] and
                    /* unsorted [i .. size-1] sublists
                    */
    int v;          /* next element to be inserted
    int ins;        /* insertion point index in sorted sublist

    for (i=1; i<size; i++)
    {
        // pick an element from the unsorted (sub)list
```

```
    v = list[i];

    // determine the correct place for the element in the sorted sublist
    ins = sort_pos(list, v, i);

    // insert the element in its sorted place
    insert(list, v, ins, i);
}

/* -----
   test function (do not implmt in assembly)
*/
int main (void)
{
    int x[] = {44,32,23,8}; // compile-time array
    int size = sizeof(x)/sizeof(int); // compile-time initializer

    ins_sort ( x, size );

    return 0;
}
```