Send the answers as separate files, but in the same email attachment. The files must be clearly marked by the problem number and your last name. For example, if your last name is SMYTHER, the answer to the first problem must in the file smythe1.scm.

Problem 1.
Using the function
\texttt{(define \textit{insert-in-order} \textit{L item})}

\begin{verbatim}
(define (insert-in-order L item)
  (cond
   ((null? L) (cons item L))
   ((< item (car L)) (cons item L))
   (else (cons (car L) (insert-in-order (cdr L) item))))
\end{verbatim}

define a function
\texttt{(define \textit{sort-input})}

\begin{verbatim}
.....
\end{verbatim}
NOTE: the name and structure of this definition may be different!

that will take input form the user until a negative number is input. The output should be all the non-negative input listed in order.

Problem 2.
Write a Scheme program that has a function taking a positive integer \textit{N} as its parameter.
\texttt{(define \textit{triangle N })}

\begin{verbatim}
.....
\end{verbatim}
It must produce a pattern of a triangle that look like the following:

If \textit{N} is 7, the output should be
\begin{verbatim}
$$$$$$$
$$$$$$
$$$$$
$$$$
$$$
$$
$
\end{verbatim}

Show three tests of your program:
Problem 3.

Write a Scheme program to determine if a list is sorted (in non-decreasing order). It should return “SORTED” if the list is sorted and “NOT SORTED” if it isn’t.

```scheme
(define (sorted listing )
    ...
)
```

Show the following tests:

```scheme
(sorted '(1 2 3))
(sorted '(10 1 2))
(sorted '(1 2 3 0))
(sorted '())
(sorted '(3 2 1))
```

Program 5

Write a Scheme program to calculate the mean (average) of a list of numbers. If the list is empty display “Not Possible.”

```scheme
(define (average list)
    ...
)
```

Show the following tests:

```scheme
(average '())
(average '(-9 9))
(average '(8))
(average '(9 8 7 6 5 4 3 2 1 0))
```