2) Dada uma lista L com elementos numéricos, ordenar a lista em ordem crescente. Se um número aparecer mais de uma vez, as repetições devem ser mantidas no resultado.

Para ordenar a lista, usaremos o *selection* sort (Ordenação por seleção). Esse é um clássico e simples algoritmo de ordenação, sua ideia principal é mover o menor valor do vetor para a primeira posição do vetor (nesse caso uma lista). Ademais, o processo se repete para o segundo menor valor, ficando na segunda posição do vetor (lista) e dessa maneira o processo é repetido sucessivamente até o último elemento.

**Código em Lisp:**

**1.** (defun Ordena\_Lista (lista)

**2.** (when lista

**3.** (let ((Menor\_Valor (reduce #'min lista)))

**4.** (cons Menor\_Valor(Ordena\_Lista (remove Menor\_Valor lista))))))

**Explicação do passo a passo:**

**Linha 1:** Com o uso da função pré-definida *defun*, iremos criar a função **Ordena\_Lista**, que recebe como argumento uma lista;

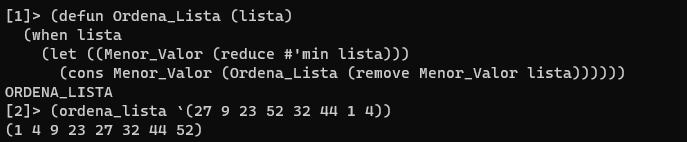
**Linha 2:** Utilizamos a estrutura de controle *when*, que avalia as expressões retorna *True* somente sefor verdadeiro, senão retorna**NIL**. O comportamento do comando *when* é de certa forma semelhante ao comando *while* de outras linguagens de programação

**Linha 3:** Com uso do operador *let* associamos à **Menor\_Valor** a função *reduce* (reduz a sequência da função &chave, chave do fim, do fim do início, do fim do valor inicial => resultado) e *reduce* recebe como argumentos a função pré-definida *min* que verifica o menor valor de uma sequência e uma lista.

**Linha 4:** Uma nova lista é construída utilizando **Menor\_Valor** (que vai possuir o menor valor da sequência) e a lista que foi passada como argumento, porém utilizando a função pré-definida *remove* que retira um valor específico de uma sequência (no caso o menor valor da lista passada como argumento).

**Execução do código – exemplos:**

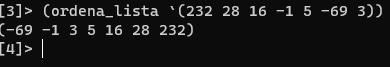
**Exemplo 1:**

****

**Lista desordenada:** (27 9 23 52 32 44 1 4)

**Lista ordenada:** (1 4 9 23 27 32 44 52)

**Exemplo 2:**

****

**Lista desordenada:** (232 28 16 -1 5 -69 3)

**Lista ordenada:** (-69 -1 3 5 16 28 232)