Jordan Taranto
Code walkthrough

Error 1

```
generate-random-integers: undefined;
 cannot reference an identifier before its definition
```

Change 1

This is defined at the end of the program so cannot be called by a function above it

```
;; Function to generate random integers (provided in the problem statement)
(define (generate-random-integers count min-value max-value)
  (define (generate n)
    (if (zero? n)
        '()
        (cons (+ min-value (random (- max-value min-value)))
              (generate (- n 1)))))
  (generate count))
```

To fix this Is imply copy and pasted it above the functions calling it

---

Error 2

```
take: contract violation
 expected: a list with at least 5 elements
 given: '(78 79 35)
```

Change 2

in the function `partition-into-scublists`
Changed this: `(cons (take lst n) (partition-into-sublists (drop lst n) n))))`
To this `(let ([sublist (if (< (length lst) n) lst (take lst n))]) ;; Handle cases where lst has fewer than n elements (cons sublist (partition-into-sublists (drop lst (length sublist)) n))))) ;; Drop only the actual size of sublist`

Test Cases

The three test cases I wrote. The original one I got from the LLM

Original

```
;; Example usage: (define test-list (generate-random-integers 43 1 100))
(displayln "Original list: ") (displayln test-list) (displayln "Sorted list
using quicksort with median-of-medians: ") (displayln (quicksort-mom test-
list))
```

Added test cases

```
;; 4
(define 4-test (generate-random-integers 4 1 100))
(displayln "Original list: ")
(displayln 4-test)
(displayln "Sorted list using quicksort with median-of-medians: ")
(displayln (quicksort-mom 4-test))

;; 43
(define test-43 (generate-random-integers 43 1 100))
(displayln "Original list: ")
(displayln test-43)
(displayln "Sorted list using quicksort with median-of-medians: ")
(displayln (quicksort-mom test-43))

;; 403
(define 403-test (generate-random-integers 403 1 100))
(displayln "Original list: ")
```

```
(displayln 403-test)
(displayln "Sorted list using quicksort with median-of-medians: ")
(displayln (quicksort-mom 403-test))



;; 400,003
(define 400003-test (generate-random-integers 400003 1 100))
(displayln "Original list: ")
(displayln 400003-test)
(displayln "Sorted list using quicksort with median-of-medians: ")
(displayln (quicksort-mom 400003-test))
```