**EXPERIMENT-17**

**Aim** - Write a prolog program to find the last element of a given list.

**Theory -**

Prolog is a [logic programming](https://en.wikipedia.org/wiki/Logic_programming) language. It has important role in artificial intelligence. Unlike many other programming languages, Prolog is intended primarily as a declarative programming language. In prolog, logic is expressed as relations (called as Facts and Rules). Core heart of prolog lies at the logic being applied. Formulation or Computation is carried out by running a query over these relations.

X is a member of list X.

Removes the first element of the list recursively.

Returns the last element of the list.

**Program -**

lastElement(X,[X]).

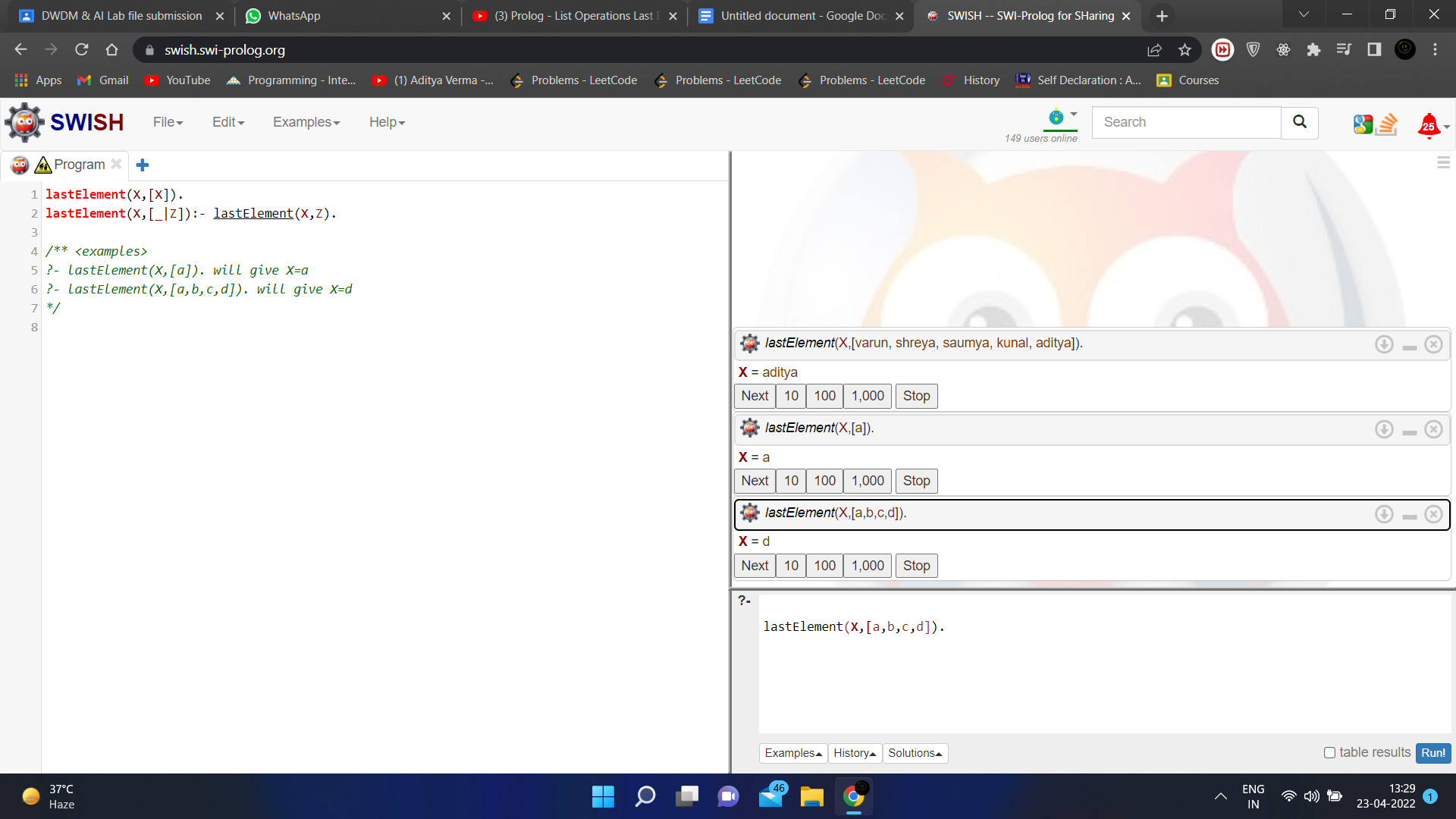
lastElement(X,[Y|Z]):- lastElement(X,Z).

**Query -**

lastElement(X,[a,b,c,d]).

lastElement(X,[a]).

lastElement(X,[varun, shreya, saumya, kunal, aditya]).

**Output -** 

**EXPERIMENT-27**

**Aim** - Define a LISP function to reverse the number entered as parameter in function call.

**Theory -**

**Program -**

(defun read-number () (format t "Enter a number: ~%") (read))

(defun reverse-string (the-string)

(if (eq (length the-string) 0)

""

(concatenate 'string (reverse-string (subseq the-string 1)) (subseq the-string 0 1))))

(defun reverse-digits (the-number) (reverse-string (write-to-string the-number)))

(let ((the-number (read-number)))

(format t "N->: ~a~%<-N: ~a~%" the-number (reverse-digits the-number)))

**Output -**

