|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Program or module:** | *Tokenizing()* | **Save As tokenizing\_test\_cases** | | | | |
| **Version 1** | | | | | **Run by: Date:** | ***Manav Zadafiya 3rd August 2023*** |
| **Description** | **+ / − Purpose** | **Data Input** | **Expected result** | **Actual result if unexpected** | **Result** | **Comments** |
| Test 1 | "+"checking minimum edge case | "hello this is tokenizing test" | Same expected output: word #1 is 'hello' word #2 is 'this' word #3 is 'is' word #4 is 'tokenizing' word #5 is 'test' |  | PASS | program works properly under normal conditions as expected. |
| Test 2 | "+"checking middle edge case | long string of 30 words(153 characters) to check if expected output is produced | all 30 words are seperated prefectly |  | PASS | program works properly under normal conditions as expected. |
| Test 3 | "+"giving unepected double spaced input | providing extra spaces between every word of string: "hello this is tokenizing test " | expected output: word #1 is 'hello' word #2 is 'this' word #3 is 'is' word #4 is 'tokenizing' word #5 is 'test' | Extra space at end: word #1 is 'hello ' word #2 is 'this ' word #3 is 'is ' word #4 is 'tokenizing ' word #5 is 'test ' | FAIL | Use strchr function to find the delimiter in the string. -Store the index of the delimiter and the length of the token. Copy the token to a new string with additional whitespaces. |
| Test 4 | "+" checking maximum edge case | input 299 total characters to check for expected output. | all words are seperated prefectly at maximum edge as expected |  | PASS | loops function properly under valid condition |
| Test 5 | "-"checking minimum over the edge case | direct enter "q" to exit without entering word | successfully exit the loop |  | PASS | everytime exit the loop when q is entered |
| Test 6 | "-"checking maximum over the edge case | entering characters out of range | loop should stop after 300 characters | Automatically starts another loop for extra characters above 300 | FAIL | validate conditon to stop loop to avoid unexpected output and prompt user to input data within range. |
| Test 7 | +"q" and enter to quit or exiting loop | small strings and then "q + enter" | successfully exit the loop |  | PASS | work as expected |
| Test 8 | "-"using strcmp to give input oustide the conditon | pressing only enter key | Expected to Return False (non-zero value) to represent the comparison failed to match with the given string. |  | PASS | The Program goes into condition and tries to perform the code block operations and returns to the user Input loop |

**INPUT/OUTPUT:**

OUTPUTS OF ALL 8 TESTS OF TOKENIZING:

TEST 1

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

hello this is tokenizing test

word #1 is 'hello'

word #2 is 'this'

word #3 is 'is'

word #4 is 'tokenizing'

word #5 is 'test'

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 2

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

this is test number 2 to check if gives expected output for string of 30 words and no invalid data input. there are tota

l of thirty words in this string.

word #1 is 'this'

word #2 is 'is'

word #3 is 'test'

word #4 is 'number'

word #5 is '2'

word #6 is 'to'

word #7 is 'check'

word #8 is 'if'

word #9 is 'gives'

word #10 is 'expected'

word #11 is 'output'

word #12 is 'for'

word #13 is 'string'

word #14 is 'of'

word #15 is '30'

word #16 is 'words'

word #17 is 'and'

word #18 is 'no'

word #19 is 'invalid'

word #20 is 'data'

word #21 is 'input.'

word #22 is 'there'

word #23 is 'are'

word #24 is 'total'

word #25 is 'of'

word #26 is 'thirty'

word #27 is 'words'

word #28 is 'in'

word #29 is 'this'

word #30 is 'string.'

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 3

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

hello this is tokenizing test

word #1 is 'hello '

word #2 is 'this '

word #3 is 'is '

word #4 is 'tokenizing '

word #5 is 'test '

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 4

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

a1 a2 a3 a4 a5 a6 a7 a8 a9 a0 b1 b2 b3 b4 b5 b6 b7 b8 b9 b0 c1 c2 c3 c4 c5 c6 c7 c8 c9 c0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d0 e1 e2 e3 e4 e5 e6 e7 e8 e9 e0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f0 g1 g2 g3 g4 g5 g6 g7 g8 g9 g0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h0 i1 i2 i3 i4 i5 i6 i7 i8 i9 i0 j1 j2 j3 j4 j5 j6 j7 j8 j9 j

word #1 is 'a1'

word #2 is 'a2'

word #3 is 'a3'

word #4 is 'a4'

word #5 is 'a5'

word #6 is 'a6'

word #7 is 'a7'

word #8 is 'a8'

word #9 is 'a9'

word #10 is 'a0'

word #11 is 'b1'

word #12 is 'b2'

word #13 is 'b3'

word #14 is 'b4'

word #15 is 'b5'

word #16 is 'b6'

word #17 is 'b7'

word #18 is 'b8'

word #19 is 'b9'

word #20 is 'b0'

word #21 is 'c1'

word #22 is 'c2'

word #23 is 'c3'

word #24 is 'c4'

word #25 is 'c5'

word #26 is 'c6'

word #27 is 'c7'

word #28 is 'c8'

word #29 is 'c9'

word #30 is 'c0'

word #31 is 'd1'

word #32 is 'd2'

word #33 is 'd3'

word #34 is 'd4'

word #35 is 'd5'

word #36 is 'd6'

word #37 is 'd7'

word #38 is 'd8'

word #39 is 'd9'

word #40 is 'd0'

word #41 is 'e1'

word #42 is 'e2'

word #43 is 'e3'

word #44 is 'e4'

word #45 is 'e5'

word #46 is 'e6'

word #47 is 'e7'

word #48 is 'e8'

word #49 is 'e9'

word #50 is 'e0'

word #51 is 'f1'

word #52 is 'f2'

word #53 is 'f3'

word #54 is 'f4'

word #55 is 'f5'

word #56 is 'f6'

word #57 is 'f7'

word #58 is 'f8'

word #59 is 'f9'

word #60 is 'f0'

word #61 is 'g1'

word #62 is 'g2'

word #63 is 'g3'

word #64 is 'g4'

word #65 is 'g5'

word #66 is 'g6'

word #67 is 'g7'

word #68 is 'g8'

word #69 is 'g9'

word #70 is 'g0'

word #71 is 'h1'

word #72 is 'h2'

word #73 is 'h3'

word #74 is 'h4'

word #75 is 'h5'

word #76 is 'h6'

word #77 is 'h7'

word #78 is 'h8'

word #79 is 'h9'

word #80 is 'h0'

word #81 is 'i1'

word #82 is 'i2'

word #83 is 'i3'

word #84 is 'i4'

word #85 is 'i5'

word #86 is 'i6'

word #87 is 'i7'

word #88 is 'i8'

word #89 is 'i9'

word #90 is 'i0'

word #91 is 'j1'

word #92 is 'j2'

word #93 is 'j3'

word #94 is 'j4'

word #95 is 'j5'

word #96 is 'j6'

word #97 is 'j7'

word #98 is 'j8'

word #99 is 'j9'

word #100 is 'j'

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 5

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 6

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

a1 a2 a3 a4 a5 a6 a7 a8 a9 a0 b1 b2 b3 b4 b5 b6 b7 b8 b9 b0 c1 c2 c3 c4 c5 c6 c7 c8 c9 c0 d1 d2 d3 d4 d5 d6 d7 d8 d9 d0 e1 e2 e3 e4 e5 e6 e7 e8 e9 e0 f1 f2 f3 f4 f5 f6 f7 f8 f9 f0 g1 g2 g3 g4 g5 g6 g7 g8 g9 g0 h1 h2 h3 h4 h5 h6 h7 h8 h9 h0 i1 i2 i3 i4 i5 i6 i7 i8 i9 i0 j1 j2 j3 j4 j5 j6 j7 j8 j9 j0 k1 k2 k3 k4 k5 k6 k7 k8 k9 k0

word #1 is 'a1'

word #2 is 'a2'

word #3 is 'a3'

word #4 is 'a4'

word #5 is 'a5'

word #6 is 'a6'

word #7 is 'a7'

word #8 is 'a8'

word #9 is 'a9'

word #10 is 'a0'

word #11 is 'b1'

word #12 is 'b2'

word #13 is 'b3'

word #14 is 'b4'

word #15 is 'b5'

word #16 is 'b6'

word #17 is 'b7'

word #18 is 'b8'

word #19 is 'b9'

word #20 is 'b0'

word #21 is 'c1'

word #22 is 'c2'

word #23 is 'c3'

word #24 is 'c4'

word #25 is 'c5'

word #26 is 'c6'

word #27 is 'c7'

word #28 is 'c8'

word #29 is 'c9'

word #30 is 'c0'

word #31 is 'd1'

word #32 is 'd2'

word #33 is 'd3'

word #34 is 'd4'

word #35 is 'd5'

word #36 is 'd6'

word #37 is 'd7'

word #38 is 'd8'

word #39 is 'd9'

word #40 is 'd0'

word #41 is 'e1'

word #42 is 'e2'

word #43 is 'e3'

word #44 is 'e4'

word #45 is 'e5'

word #46 is 'e6'

word #47 is 'e7'

word #48 is 'e8'

word #49 is 'e9'

word #50 is 'e0'

word #51 is 'f1'

word #52 is 'f2'

word #53 is 'f3'

word #54 is 'f4'

word #55 is 'f5'

word #56 is 'f6'

word #57 is 'f7'

word #58 is 'f8'

word #59 is 'f9'

word #60 is 'f0'

word #61 is 'g1'

word #62 is 'g2'

word #63 is 'g3'

word #64 is 'g4'

word #65 is 'g5'

word #66 is 'g6'

word #67 is 'g7'

word #68 is 'g8'

word #69 is 'g9'

word #70 is 'g0'

word #71 is 'h1'

word #72 is 'h2'

word #73 is 'h3'

word #74 is 'h4'

word #75 is 'h5'

word #76 is 'h6'

word #77 is 'h7'

word #78 is 'h8'

word #79 is 'h9'

word #80 is 'h0'

word #81 is 'i1'

word #82 is 'i2'

word #83 is 'i3'

word #84 is 'i4'

word #85 is 'i5'

word #86 is 'i6'

word #87 is 'i7'

word #88 is 'i8'

word #89 is 'i9'

word #90 is 'i0'

word #91 is 'j1'

word #92 is 'j2'

word #93 is 'j3'

word #94 is 'j4'

word #95 is 'j5'

word #96 is 'j6'

word #97 is 'j7'

word #98 is 'j8'

word #99 is 'j9'

word #100 is 'j'

Type a few words separated by space (q - to quit) :

word #1 is 'k1'

word #2 is 'k2'

word #3 is 'k3'

word #4 is 'k4'

word #5 is 'k5'

word #6 is 'k6'

word #7 is 'k7'

word #8 is 'k8'

word #9 is 'k9'

word #10 is 'k0'

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 7

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

hello this is test 7

word #1 is 'hello'

word #2 is 'this'

word #3 is 'is'

word #4 is 'test'

word #5 is '7'

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*

TEST 8

\*\*\* Start of Tokenizing Words Demo \*\*\*

Type a few words separated by space (q - to quit) :

Type a few words separated by space (q - to quit) :

Type a few words separated by space (q - to quit) :

Type a few words separated by space (q - to quit) :

Type a few words separated by space (q - to quit) :

Type a few words separated by space (q - to quit) :

q

\*\*\* End of Tokenizing Words Demo \*\*\*