





**Lec6**

 

String standard predicates

1. Isname(string) test if the content of the string is name or not

Isname("abc") yes

Isname("123"). No

2.char\_int(char,integer) convert the character to its integer value and the

opposit

 Char\_int('A',X)

 X=65

 Char\_int(X,65)

 X='A'

3. Str\_char(string,char) convert the string (of one char) to char and the

opposit

 Str\_char("A",X)

 X='A'

 Str\_char(X,'A')

 X="A"

4. str\_real (string,real) convert the string (ofreal) to real and the opposit

 Str\_real("0.5",X)

 X=0.5

 Str\_real(X,0.5)

 X="0.5"

5.Fronttoken(string,string,string).

 Take token of word from the string and return the reminder of the string .

 Fronttoken(string,token,rem).

 Fronttoken("ab cd ef",X,Y).

 X="ab" y="cd ef"

 Fronttoken("c def",X,Y)

 X="cd" Y="ef"

6. Frontstring(integer,string,string,string)

 28

Take a string(str) with length specified by the integer value and

return the reminder

 Frontstring(integer,string,str,rem)

 Frontstr(3,"ahmed",X,Y)

 X="ahm" Y="ed"

 Frontstr(2,"abcde",X,Y).

 X="ab" Y="cde"

 Frontstr(3,S,"ahm","ed").

 S="ahmed"

7. Frontchar(string,char,string).

Take one char from a specific string and return the reminder

 Frontchar(string,char,rem).

 Frontchar("ahmed",X,Y)

 X='a' Y="hmed"

 Frontchar(X,'a',"hmed")

 X="ahmed"

8. Str\_len(string,length)

Return the length of specific string

 Str\_len("ahmed",X)

 X=5

 Str\_len("ab",X)

 X=2

 Str\_len("ab",3) no

 Str\_len(X,3) X="---"

9. Concat(string,string,string).

Concat two string together to produce one string

 Concat("ab","cd",X)

 X="abcd"

10. Upper\_lower(string,string)

Convert the string in upper case(in capital letter) to the lower case

(small letter) and the opposite.

 Upper\_lower(capital\_letter,small\_letter)

 Upper\_lower("ABC",X)

 X="abc"

 29

 Upper\_lower("Abc",X)

 X="abc"

 Upper\_lower(,X,"abc")

 X="ABC"

Prolog Programs that deal with string

Ex1:Pogram that read two string and concat them in one string as upper

case.

predicates

start(string)

clauses

start(X):-readln(S),readln(S1),concat(S,S1,S2),upper\_lower(X,S2).

Goal

 Start(X)

 Output:

Ahmed

Ali

X=AHMEDALI yes

Ex2:program that read string of one character then return the integer

value of this char.

predicates

start(integer).

clauses

start(X):-readln(S),str\_char(S,X).

goal

start(X)

 Output:

a

X=97

yes

 30

Ex3: Program that take a string of words and print each word in a line as

upper case.

predicates

start(string).

clauses

start(S):-fronttoken(S,S3,S2), upper\_lower(S1,S3), write(S1),

nl,start(S2).

start("").

Goal

Start("ali is a good boy").

 Output:

ALI

IS

A

GOOD

BOY

yes

Ex4: program that take a string and convert each character it contain to

its corresponding integer value.

Predicates

start(string).

clauses

start(S):-frontstr(1,S,S1,S2), char\_int(S1,I), write(I), nl , start(S2).

start("").

Goal

Start("abc").

 Output:

97

98

99

Yes

Ex5: program that return the number of names in a specific string.

predicates

start(string,INTEGER).

clauses

start(S,X):-fronttoken(S,S1,S2),isname(S1),X1=X+1,start(S2,X1).

start(S,X):-fronttoken(S,\_,S2),start(S2,X).

start("",X):-write("the number of names is", X).

goal

start("ali has 2 cars").

Output:

The no. of names is 3

Yes

Ex6:program that split a specific string to small string with length 3 char.

predicates

start(string).

clauses

start("").

start(S):-str\_len(S,I), I MOD 3=0, frontstr(3,S,S1,S2), write(S1),

nl,start(S2).

start(S):-concat(S," ",S1),start(S1).

Goal

Start("abcdefg").

Output:

abc

def

g

yes

H.W

1- Write a prolog program that do the following: convert the string such as

"abcdef" to 65 66 67 68 69 70.

2-Program tofind the number of tokens and the number of character in a

specific string such as: "ab c def" the output is tokens and 6 character.