

```
1 dlugosc([],0).
2 dlugosc([_|0],N) :- dlugosc(0,M),N is M+1.
```

```
≡ ?- dlugosc([1,2,3,4,5],X).
```

Odcięcie

! Przerywa proces nawracania

```
1 f(X,2):-X=<2,! .
2 f(X,1):-2<X,X=<4,! .
3 f(X,0):-4<X,! .
```

```
≡ ?- f(6,Y).
```

```
≡ ?- f(3,Y).
```

```
≡ ?- f(1,Y).
```

Zadanie 3

Rozważmy następującą bazę danych:

```
1 teaches(dr_fred, history).
2 teaches(dr_fred, english).
3 teaches(dr_fred, drama).
4 teaches(dr_fiona, physics).
5 studies(alice, english).
6 studies(angus, english).
7 studies(amelia, drama).
8 studies(alex, physics).
```

```
≡ ?- teaches(dr_fred, Course),studies(Student, Course).
```

```
≡ ?- !,teaches(dr_fred, Course),studies(Student, Course).
```

```
≡ ?- teaches(dr_fred, Course),!,studies(Student, Course).
```

```
≡ ?- teaches(dr_fred, Course),!,studies(Student, Course),!.
```

Zadanie 4

Zdefiniuj funkcję:

```
f(x)= -4 dla x>3
f(x)= 2 dla 3 ≥ x > 0
f(x)= 11 dla x ≤ 0
```

W definicji wykorzystaj odcięcie (!). Przetestuj działanie funkcji dla 3 różnych argumentów.

```
1 f(X,-4):-X>3,!.
2 f(X,2):- X=<3,X>0,!.
3 f(X,11):- X=<0,!.

```

```
≡ ?- f(10,Y).
```

```
≡ ?- f(2,Y).
```

Zadanie 5

```
1 silnia(0,1).
2 silnia(N,S):-
3   NewN is N-1,
4   silnia(NewN,NewS),
5   S is NewS*N,!.

```

```
≡ ?- silnia(3,6).
```

Create a

Program	Query	Markdown	HTML
---------	-------	----------	------

 cell here

```
≡ ?- silnia(3,X).
```

```
≡ ?- silnia(4,X).
```

```
≡ ?- silnia(5,X).
```

```
≡ ?- silnia(6,X).
```

Drzewa

```
1 rodzic(a,b).
2 rodzic(a,c).
3 rodzic(a,d).
4 rodzic(b,e).
5 rodzic(b,f).
6 rodzic(c,g).
7 rodzic(c,h).
8 rodzic(c,i).
9 rodzic(d,j).
10 rodzic(f,k).
11 rodzic(f,l).

```

```
12
13 rodzenstwo(A,B) :- rodzic(C,A),rodzic(C,B),A\==B.
```

```
≡ ?- rodzenstwo(a,b).
```



```
≡ ?- rodzenstwo(a,X).
```



```
≡ ?- rodzenstwo(b,X).
```



Wieszchołki są na tym samym poziomie gdy ich rodzice są na tym samym poziomie.

```
1 ten_sam_poziom(X,X).
2 ten_sam_poziom(X,Y):-rodzic(A,X),rodzic(B,Y),X\==Y,ten_sam_poziom(A,B).
```



```
≡ ?- ten_sam_poziom(X,e).
```



```
1 poziom(X,0):-not(rodzic(_,X)),!.
2 poziom(X,N):-rodzic(Y,X),poziom(Y,M),N is M+1.
```



```
≡ ?- poziom(e,N).
```



```
≡ ?- poziom(X,2).
```



```
1 path(a):-write('o-').
2 path(X):-rodzic(A,X),path(A),write(A),write(' ->').
3 znajdz(X):-path(X),write(X),!.
```



```
≡ ?- znajdz(e).
```



```
1 lisc(X):-not(rodzic(X,_)).
```



```
≡ ?- lisc(X).
```

