**Задача 1.**

а) Да се напише дефиниция на ПРОЛОГ на предикат p(X,L,N), който по даден списък от естествени числа L и естествено число X връща в числото N:

Вариант А
броят на тези елементи на L, които са кратни на X

Вариант Б
сумата на тези елементи на L, които са <= X

б) Да се напише дефиниция на ПРОЛОГ на предикат q(L1,L2,L), където L1=[X1,...,Xn], L2=[Y1,...,Yk] за n,k>=1, който връща в L:

Вариант А
списък, състоящ се от всички двойки [Xi,Mi], където Mi е броят на числата от L2, които са кратни на Xi

Вариант Б
списък, състоящ се от всички двойки [Yi,Ni], където Ni е сумата на числата от L1, които са <=Yi

**Задача 2**

а) Да се напише дефиниция на ПРОЛОГ на предикат p(X,Y,N), който по дадени естествени числа X,Y връща в N:

Вариант А
броят на числата, които делят X, но не делят Y

Вариант Б
броят на общите делители на X и Y

б) Да се напише генератор без повторения на множеството

Вариант А
{ (x,y) | x>=1, y>=1 и броят на числата, който делят X, но не делят Y е точно Y }

Вариант Б
{ (x,y) | x>=y и броят на общите делители на X и Y е точно 5 }

Време за работа - 2 часа.

::BEGIN CODE::
%zadacha 1 grupa A

zad1\_p(\_, [], 0):-!.
zad1\_p(X, [H|T], N):-H mod X =:= 0, !, zad1\_p(X, T, N1), N is N1 + 1.
zad1\_p(X, [H|T], N):-zad1\_p(X, T, N).

zad1\_q([X], L2, [[X, N]]):-zad1\_p(X, L2, N), !.
zad1\_q([H|T], L2, [[H, N]|R]):-zad1\_p(H, L2, N), zad1\_q(T, L2, R).

%zadacha 2 grupa A

int(0).
int(X):-int(Y), X is Y + 1.

between(X, Y, X):- X =< Y.
between(X, Y, Z):- X < Y, X1 is X + 1, between(X1, Y, Z).

ph(X, X, \_, 0):-!.
ph(X, \_, D, 0):- D > X, !.
ph(X, Y, D, N):- D =< X, X mod D =:= 0, Y mod D =\= 0, !, D1 is D + 1, ph(X, Y, D1, N1), N is N1 + 1.
ph(X, Y, D, N):- D =< X, D1 is D + 1, ph(X, Y, D1, N).

zad2\_p(X, Y, N):- ph(X, Y, 2, N).

zad2\_gen\_xy([X, Y]):-int(X), between(1, X, Y), zad2\_p(X, Y, Y).

::END CODE::

Това е моето решение, но не давам 100% гаранция  При моите тестове работи.

Задачи давани на контролни и писмени изпит по ПРОЛОГ от предни години.

  Да се напише генератор без повторения на всички числа z, които се представят във вида z = xy за x,y > 1 естествени числа.

 Да се напише генератор gen\_list(+M,+N,-L) на всички строго растящи редици от естествени числа от M до N включително. Допуска се и празната редица. Задачата има поне три различни по идея решения.

Упътване: подсписъци на списък, използване на between, рекурсивно свеждане към задачата за M+1 и N.

 Да се напише предикат find\_list(+M,+N,-L), който намира списък от всички строго растящи редици от естествени числа от M до N включително. Допуска се и празната редица. Задачата има поне две различни по идея решения. Упътване: последователно пораждане на всички редици, рекурсивно свеждане към задачата за M+1 и N.

Ето решение на задачите:

%::BEGIN CODE::
%zadacha 1
div\_while\_possible(X, X):-!.
div\_while\_possible(Z, X):-Z mod X =:= 0, Z1 is Z // X, div\_while\_possible(Z1, X).

is\_xy(Z):-Z1 is Z - 1, between(2, Z1, X), div\_while\_possible(Z, X), !.

gen\_xy(Z):-int(Z), is\_xy(Z).

%zadacha 2 (resheno chrez svejdane)
gen\_list(A, A, [A]).
gen\_list(A, A, []).
gen\_list(A, B, [A|T]):-B >= A, A1 is A + 1, gen\_list(A1, B, T).
gen\_list(A, B, T):-B >= A, A1 is A + 1, gen\_list(A1, B, T).

%zadacha 3
add\_x(\_, [], []):-!.
add\_x(X, [H|T], [[X|H]|T1]):-add\_x(X, T, T1).

find\_list(X, X, [[], [X]]):-!.
find\_list(X, Y, R):- X add\_x(X, L, L1), append(L1, L, R).
%::END CODE::

Между другото - на упражнението казахме, че при това решение на задача 3, списъкът ще се генерира сортиран (а и аз си мислех, че е така), но всъщност не става... както и да се преподреждах аргументите не успях да го изкарам сортиран. Това не е проблем за решението - не се иска сортиран списък, но все пак да отбележа...