Artificial Intelligence Lab

Code: PMDS601P

Digital Assignment 3

Name: Soumyadeep Ganguly

Reg. No.: 24MDT0082

Course: M.Sc in Data Science

1. **Write a Prolog program defining a binary arithmetic predicate blahblah/2, which takes as its first argument a number X, and returns in its second argument the number $X^X + 2$.**

Code:

```
blahblah(X, Y):-
    XX is X**X,
    Y is XX + 2.
```

Output:

```
% c:/Users/sambh/OneDrive/Documents/Prolog/AIDA3_1.pl compiled (
c, 1 clauses
?- blahblah(3, X).
X = 29.

?- blahblah(3, 27).
false.

?-
```

Convert blahblah/2 into a binary operator in the same file. For instance:

Code:

```
:- op(500, xfx, blahblah).

blahblah(X, Y):-
    XX is X**X,
    Y is XX + 2.
```

Output:

```
% c:/Users/sambh/OneDri
c, 0 clauses
?- 3 blahblah X.
X = 29.

?- 3 blahblah 27.
false.
```

2. Convert your previously defined predicates of set union and set inter□section to appropriately defined operators.

Code:

```
:- op(500, xfy, union).
:- op(500, xfy, intersection).

Set1 union Set2 :-
   union(Set1, Set2, Result),
   write('Union: '), write(Result).

Set1 intersection Set2 :-
   intersection(Set1, Set2, Result),
   write('Intersection: '), write(Result).

union([], B, B).
union([H|T], B, [H|U]) :-
   \+ member(H, B),
   union(T, B, U).
union([H|T], B, U) :-
   member(H, B),
   union(T, B, U).

intersection([], _, []).
intersection([H|T], B, [H|I]) :-
   member(H, B),
```

```
    intersection(T, B, I).
intersection([H|T], B, I) :-
    \+ member(H, B),
    intersection(T, B, I).
```

Output:

```
% c:/Users/sambh/OneDrive/Documents/Prolog/AID
c, 8 clauses
?- [1,2,3] union [4,5,6].
Union: [1,2,3,4,5,6]
true ;
false.

?- [1,2,3] intersection [3,5,6].
Intersection: [3]
true ;
false.

?-
```