% facts

:- dynamic hp/2.

:- dynamic atk/2.

country(ice\_country).

country(fire\_country).

country(water\_country).

country(oil\_country).

character("ice\_king").

character("ice\_prince").

character("ice\_General").

character("ice\_Soldier").

character("fire\_king").

character("fire\_General").

character("fire\_Soldier").

character("fire\_Elf").

character("water\_king").

character("water\_prince").

character("water\_General").

character("water\_Soldier").

character("oil\_queen").

character("oil\_prince").

character("oil\_General").

character("oil\_Soldier").

ice\_country(ice\_king).

ice\_country(ice\_prince).

ice\_country(ice\_General).

ice\_country(ice\_Soldier).

fire\_country(fire\_king).

fire\_country(fire\_General).

fire\_country(fire\_Soldier).

fire\_country(fire\_Elf).

water\_country(water\_king).

water\_country(water\_prince).

water\_country(water\_General).

water\_country(water\_Soldier).

oil\_country(oil\_queen).

oil\_country(oil\_prince).

oil\_country(oil\_General).

oil\_country(oil\_Soldier).

level(ice\_king,97,ice).

level(ice\_prince,56,ice).

level(ice\_General,72,ice).

level(ice\_Soldier,31,ice).

level(fire\_king,98,fire).

level(fire\_General,70,fire).

level(fire\_Soldier,34,fire).

level(fire\_Elf,50,fire).

level(water\_king,95,water).

level(water\_prince,49,water).

level(water\_General,77,water).

level(water\_Soldier,35,water).

level(oil\_queen,91,oil).

level(oil\_prince,58,oil).

level(oil\_General,76,oil).

level(oil\_Soldier,33,oil).

good\_relationship(water,ice).

good\_relationship(ice,water).

good\_relationship(fire,oil).

good\_relationship(oil,fire).

good\_relationship(fire,fire).

good\_relationship(oil,oil).

good\_relationship(ice,ice).

good\_relationship(water,water).

good\_relationship2(water\_country,ice\_country).

good\_relationship2(ice\_country,water\_country).

good\_relationship2(fire\_country,oil\_country).

good\_relationship2(oil\_country,fire\_country).

restrain(water,fire).

restrain(fire,ice).

restrain(ice,oil).

restrain(oil,water).

attributes(water).

attributes(fire).

attributes(ice).

attributes(oil).

hp("ice\_king", 12071).

hp("fire\_king", 10287).

hp("water\_king", 9797).

hp("oil\_queen", 13103).

atk("ice\_king", 701).

atk("fire\_king", 352).

atk("water\_king", 943).

atk("oil\_queen", 323).

atk("Silver\_Sword", 354).

atk("Amos\_Bow", 510).

atk("Rain\_Slasher", 354).

atk("Bloodtainted\_Great\_Sword", 674).

atk("Fillet\_Blade", 565).

weaponType("Silver\_Sword", sword).

weaponType("Amos\_Bow", bow).

weaponType("Rain\_Slasher", spear).

weaponType("Bloodtainted\_Great\_Sword", sword).

weaponType("Fillet\_Blade", sword).

useWeaponType("ice\_king",bow).

useWeaponType("fire\_king",sword).

useWeaponType("water\_king", spear).

useWeaponType("oil\_queen", sword).

% rules

%if level A higher than level,we think A will defeat B

defeated(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),Level1>Level2.

lost(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),Level1<Level2.

tied(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),Level1=Level2.

%when A is restrained by B, but A defeats B(in this situation，The level of A needs to be 5 higher than that of B )

was\_restrained\_but\_defeating(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),restrain(Attribute1,Attribute2),Level1-5>Level2.

%when A restrained B, and A defeats B(in this situation，The level of A can be 5 lower than that of B)

restrained\_and\_defeating(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),restrain(Attribute1,Attribute2),Level1+5>Level2.

%when A is restrained by B, and A lost to B(in this situation，The level of A need to be 5 lower than that of B)

was\_restrained\_and\_lost(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),restrain(Attribute1,Attribute2),Level1-5<Level2.

%when A restrained B, and A lost to B(in this situation，The level of A can be 5 higher than that of B)

restrained\_but\_lost(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),restrain(Attribute1,Attribute2),Level1+5<Level2.

% Master,Grandmaster,and Challenger(Master is from 30 to 55,Grandmaster is from 56 to 75,Challenger is from 76 to 99)

is\_Master(A):- level(A,Level1,Attribute1),attributes(Attribute1),Level1>29,Level1<56.

is\_Grandmaster(A):- level(A,Level1,Attribute1),attributes(Attribute1),Level1>55,Level1<76.

is\_Challenger(A):- level(A,Level1,Attribute1),attributes(Attribute1),Level1>75,Level1<100.

% is\_superiors means it means your superiors, you accept their jurisdiction

is\_superiors(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),Attribute1=Attribute2,Level1>Level2.

country\_is\_friends(A,B):-country(A),country(B),good\_relationship2(A,B).

people\_is\_friends(A,B):-level(A,Level1,Attribute1),level(B,Level2,Attribute2),A\=B,Level1+Level2<999,good\_relationship(Attribute1,Attribute2).

canUseWeapon(CharacterNAme, WeaponName) :-

useWeaponType(CharacterNAme, Type),

weaponType(WeaponName, Type).

hit(ObjectA, ObjectB) :-

hp(ObjectA, HP\_A),

hp(ObjectB, HP\_B),

atk(ObjectA, ATK\_A),

atk(ObjectB, ATK\_B),

write(ObjectA),

write(' cause '),

write(ATK\_A),

write(' damage '),

write('to '),

write(ObjectB).

equip(CharacterName, WeaponName) :-

canUseWeapon(CharacterName, WeaponName) ->

( atk(CharacterName, X),

write('The ATK of '), write(CharacterName), write(' is: '), write(X), nl,

atk(WeaponName, Y),

NewX is X + Y,

write('New ATK of '), write(CharacterName), write(' is: '), write(NewX), nl,

retract(atk(CharacterName, X)),

asserta(atk(CharacterName, NewX))

).

printStatus(Object) :-

hp(ObjectA, HP\_A),

hp(ObjectB, HP\_B),

atk(ObjectA, ATK\_A),

atk(ObjectB, ATK\_B),

write('Status of '),

write(Object),

write(':'), nl,

hp(Object, HP),

atk(Object, ATK),

write('HP: '),

NewHP is HP-ATK\_A,

write(NewHP), nl,

write('ATK: '),

write(ATK), nl.

attack(CharacterName, EnemyName) :-

character(CharacterName)\=character(EnemyName) ->

( write(CharacterName),

write(' attack '),

write(EnemyName),

write('!'), nl,

write('Choose a equipment:'),

read(Equipment),

canUseWeapon(CharacterName, Equipment) ->

( equip(CharacterName, Equipment),

hit(CharacterName, EnemyName),nl,

printStatus(EnemyName)

);

write(CharacterName),

write(' can not use '),

write(Equipment),

write(', Choose another!'), nl,

attack(CharacterName, EnemyName)

), !;

( write(CharacterName),

write(' will not attack '),

write(EnemyName),

write(', because they are not hostiles.'), nl

).