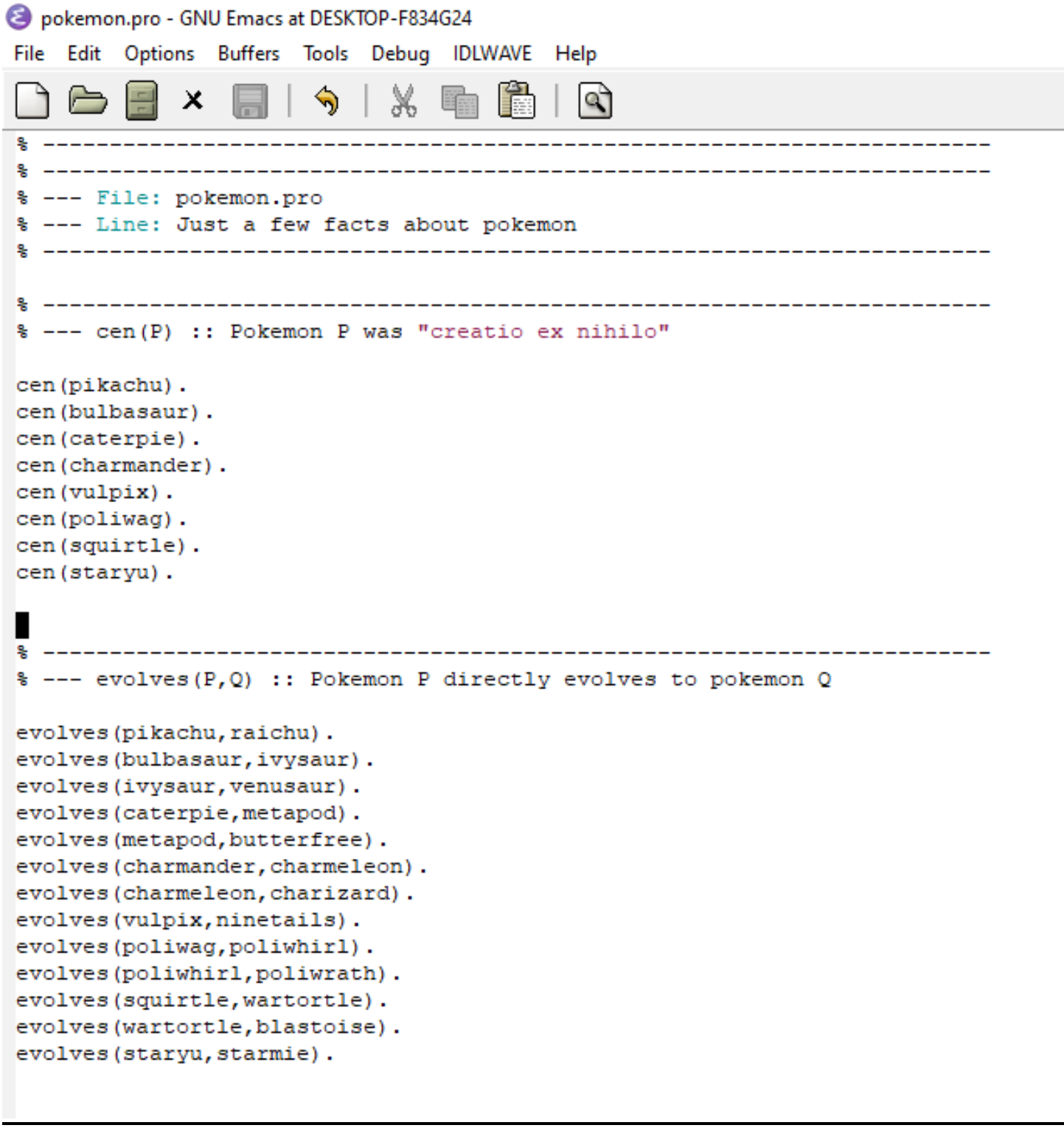
Joseph Scollo

Abstract:

In this assignment one can learn how to interact with a prolog knowledge base by implementing specific queries. List processing prolog is also covered in this assignment. Extracting specific information from a pokemon knowledge base with the use of various created prolog queries is used to give an implementation example with part 1 and a simple heads/tails coin flip is used to illustrate part 2.

Task 1 – Pokemon

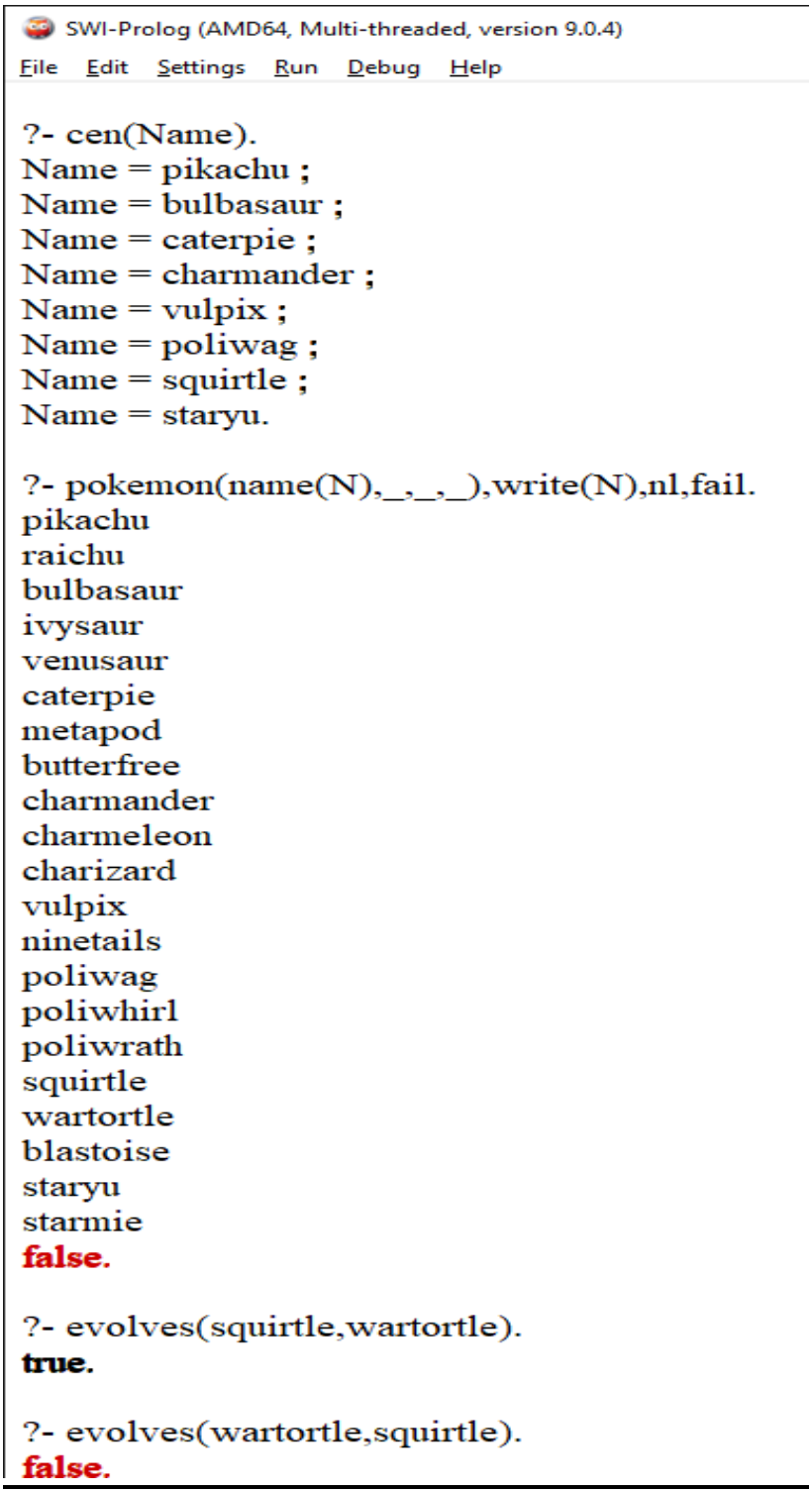
Part 1: Initial Pokemon KB:




```
% ------------------------------------------------------------------------
% ------------------------------------------------------------------------
% --- File: pokemon.pro
% --- Line: Just a few facts about pokemon
% ------------------------------------------------------------------------

% ------------------------------------------------------------------------
% --- cen(P) :: Pokemon P was "creatio ex nihilo"

cen(pikachu).
cen(bulbasaur).
cen(caterpie).
cen(charmander).
cen(vulpix).
cen(poliwag).
cen(squirtle).
cen(staryu).

% ------------------------------------------------------------------------
% --- evolves(P,Q) :: Pokemon P directly evolves to pokemon Q

evolves(pikachu,raichu).
evolves(bulbasaur,ivysaur).
evolves(ivysaur,venusaur).
evolves(caterpie,metapod).
evolves(metapod,butterfree).
evolves(charmander,charmeleon).
evolves(charmeleon,charizard).
evolves(vulpix,ninetails).
evolves(poliwag,poliwhirl).
evolves(poliwhirl,poliwrath).
evolves(squirtle,wartortle).
evolves(wartortle,blastoise).
evolves(staryu,starmie).
```

```
% ------------------------------------------------------------------------
% --- pokemon(name(N),T,hp(H),attach(A,D)) :: There is a pokemon with
% --- name N, type T, hit point value H, and attach named A that does
% --- damage D.

pokemon(name(pikachu), electric, hp(60), attack(gnaw, 10)).
pokemon(name(raichu), electric, hp(90), attack(thunder-shock, 90)).

pokemon(name(bulbasaur), grass, hp(40), attack(leech-seed, 20)).
pokemon(name(ivysaur), grass, hp(60), attack(vine-whip, 30)).
pokemon(name(venusaur), grass, hp(140), attack(poison-powder, 70)).

pokemon(name(caterpie), grass, hp(50), attack(gnaw, 20)).
pokemon(name(metapod), grass, hp(70), attack(stun-spore, 20)).
pokemon(name(butterfree), grass, hp(130), attack(whirlwind, 80)).

pokemon(name(charmander), fire, hp(50), attack(scratch, 10)).
pokemon(name(charmeleon), fire, hp(80), attack(slash, 50)).
pokemon(name(charizard), fire, hp(170), attack(royal-blaze, 100)).

pokemon(name(vulpix), fire, hp(60), attack(confuse-ray, 20)).
pokemon(name(ninetails), fire, hp(100), attack(fire-blast, 120)).

pokemon(name(poliwag), water, hp(60), attack(water-gun, 30)).
pokemon(name(poliwhirl), water, hp(80), attack(amnesia, 30)).
pokemon(name(poliwrath), water, hp(140), attack(dashing-punch, 50)).

pokemon(name(squirtle), water, hp(40), attack(bubble, 10)).
pokemon(name(wartortle), water, hp(80), attack(waterfall, 60)).
pokemon(name(blastoise), water, hp(140), attack(hydro-pump, 60)).
```

Part 2: Interaction Demo with the Initial KB:



```
?- evolves(X,Y),evolves(Y,Z).
X = bulbasaur,
Y = ivysaur,
Z = venusaur ;
X = caterpie,
Y = metapod,
Z = butterfree ;
X = charmander,
Y = charmeleon,
Z = charizard ;
X = poliwag,
Y = poliwhirl,
Z = poliwrath ;
X = squirtle,
Y = wartortle,
Z = blastoise ;
false.

?- evolves(X,Y),evolves(Y,Z),write(X),write(-->),write(Z),nl,fail.
bulbasaur-->venusaur
caterpie-->butterfree
charmander-->charizard
poliwag-->poliwrath
squirtle-->blastoise
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- pokemon(name(N),_,_,_),write(N),nl,fail.
pikachu
raichu
bulbasaur
ivysaur
venusaur
caterpie
metapod
butterfree
charmander
charmeleon
charizard
vulpix
ninetails
poliwag
poliwhirl
poliwrath
squirtle
wartortle
blastoise
staryu
starmie
false.

?- pokemon(name(N),fire,_,_),write(N),nl,fail.
charmander
charmeleon
charizard
vulpix
ninetails
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- pokemon(Name,Kind,_,_),write('nks('),write(Name),write(', kind('),write(Kind),write('
nks(name(pikachu), kind(electric))
nks(name(raichu), kind(electric))
nks(name(bulbasaur), kind(grass))
nks(name(ivysaur), kind(grass))
nks(name(venusaur), kind(grass))
nks(name(caterpie), kind(grass))
nks(name(metapod), kind(grass))
nks(name(butterfree), kind(grass))
nks(name(charmander), kind(fire))
nks(name(charmeleon), kind(fire))
nks(name(charizard), kind(fire))
nks(name(vulpix), kind(fire))
nks(name(ninetails), kind(fire))
nks(name(poliwag), kind(water))
nks(name(poliwhirl), kind(water))
nks(name(poliwrath), kind(water))
nks(name(squirtle), kind(water))
nks(name(wartortle), kind(water))
nks(name(blastoise), kind(water))
nks(name(staryu), kind(water))
nks(name(starmie), kind(water))
false.

?- pokemon(name(N),_,_,attack(waterfall,_)),write(N),nl,fail.
wartortle
false.

?- pokemon(name(N),_,_,attack(poison-powder,_)),write(N),nl,fail.
venusaur
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- pokemon(_,water,_,attack(A,_)),write(A),nl,fail.
water-gun
amnesia
dashing-punch
bubble
waterfall
hydro-pump
slap
star-freeze
false.

?- pokemon(name(poliwhirl),_,hp(D),_),write('HP'),write(" " = " "),write(D),nl,fail.
HP = 80
false.

?- pokemon(name(butterfree),_,hp(D),_),write('HP'),write(" " = " "),write(D),nl,fail.
HP = 130
false.

?- pokemon(name(Name),_,hp(HP),_),HP > 85,write(Name),nl,fail.
raichu
venusaur
butterfree
charizard
ninetails
poliwrath
blastoise
false.

?- pokemon(name(Name),_,_,attack(_,Damage)),Damage > 60,write(Name),nl,fail.
raichu
venusaur
butterfree
charizard
ninetails
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

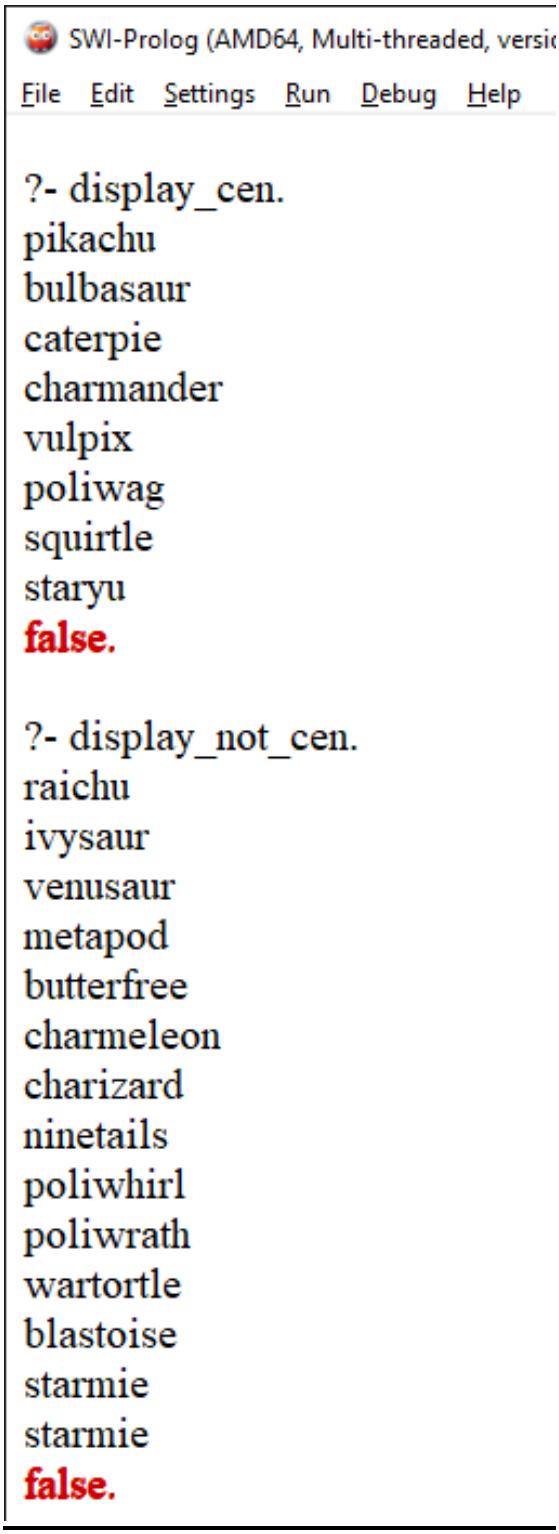
File Edit Settings Run Debug Help

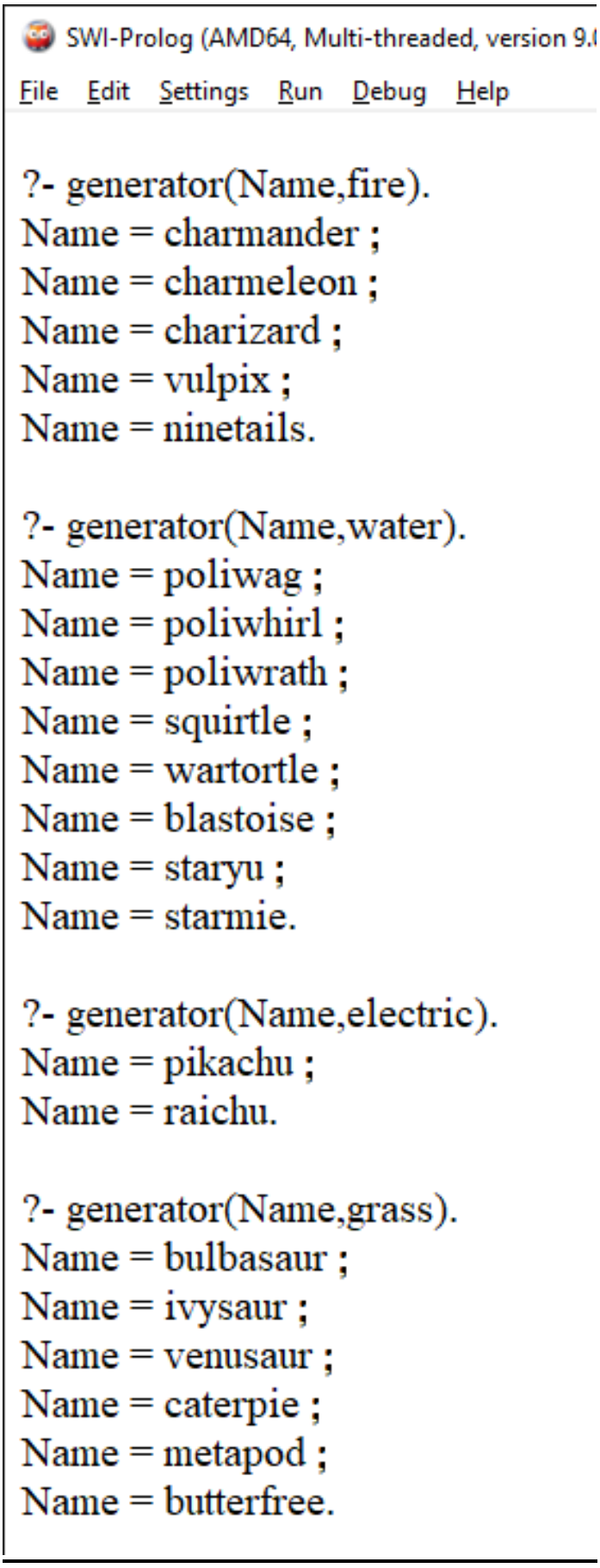
```
?- pokemon(name(Name),_,hp(HP),_),write(Name),write(' : '),write(HP),nl,f
pikachu : 60
raichu : 90
bulbasaur : 40
ivysaur : 60
venusaur : 140
caterpie : 50
metapod : 70
butterfree : 130
charmander : 50
charmeleon : 80
charizard : 170
vulpix : 60
ninetails : 100
poliwag : 60
poliwhirl : 80
poliwrath : 140
squirtle : 40
wartortle : 80
blastoise : 140
staryu : 40
starmie : 60
false.

?-
```

Part 3: KB Extension:

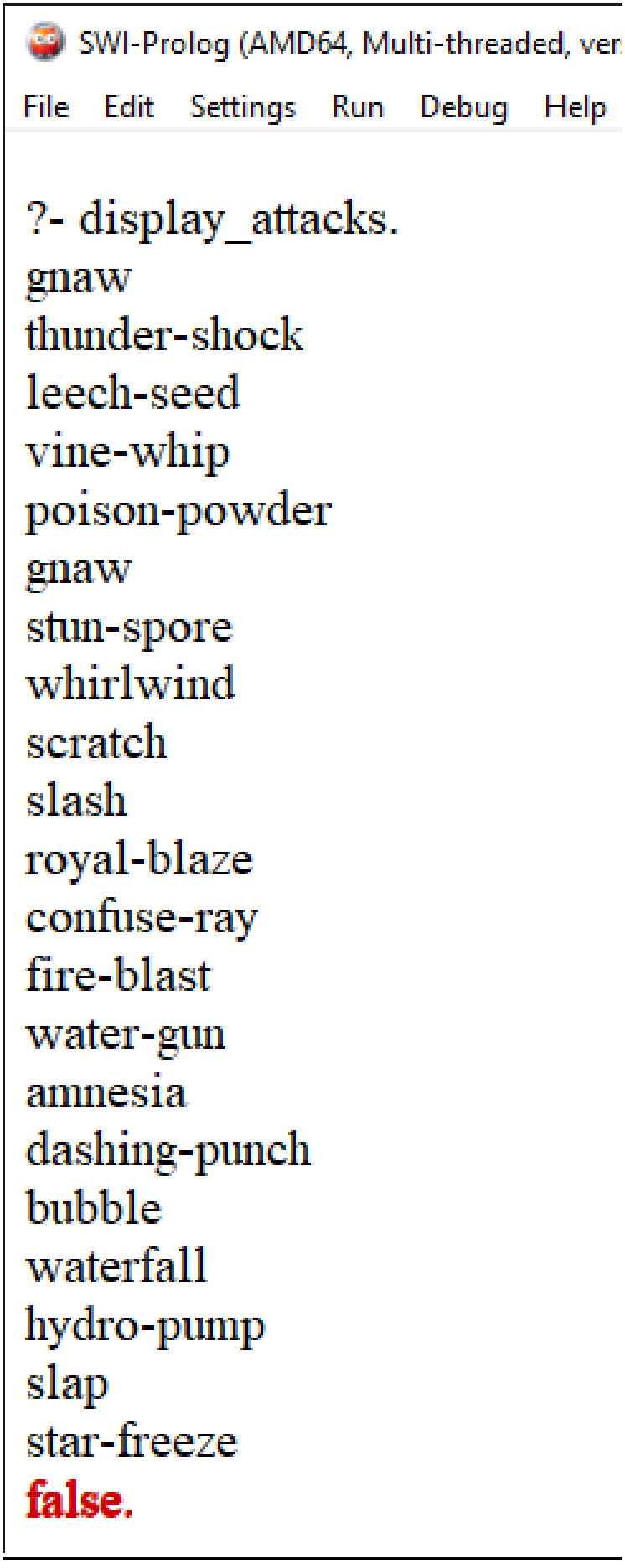
```
%%%-------------------------------------------------------------------------------------------------------------------------------------------%%%
%- Various commands(queries) that return information from the above pokemon knoweldge base.                                                  -%
%-------------------------------------------------------------------------------------------------------------------------------------------%
%- display_cen :: displays all "creatio ex nihilo" pokemon by name.                                                                          -%
%- display_not_cen :: displays all pokemon of which are not "creatio ex nihilo" by name.                                                    -%
%- generator(N,T) :: takes parameters (N) pokermon name and (T)pokemon Type and returns true if name and type coincides.                    -%
%- display_names :: dislays the names of all pokemon in the knowledge base.                                                                 -%
%- display_attacks :: lists all of the attacks in the knowledge base.                                                                       -%
%- display_cen_attacks :: lists all "creatio ex nihilo" pokemon attacks.                                                                    -%
%- indicate_attack(N) :: Takes a pokemon name (N) as a parameter and returns its attack.                                                    -%
%- indicate_attacks :: list all pokemon with there speicific attacks in the kwoledge base.                                                 -%
%- powerful(N):: Takes a polemon name(N) as a parameter and returns true if the pokemons attack damage points is above 55.                 -%
%- tough(N) :: Takes a pokon name(N) as a paramater and returns true is the pokomons HP is greater than 100.                               -%
%- powerful_but_vulnerable(N) :: Takes a polemon name(N) as a parameter and returns true if --                                             -%
%- the pokemons attack damage points is above 55 and  pokomons HP is less than 101.                                                         -%
%- type(N,T) :: takes two parameters, pokemon name(N) and typt(T). Returns true if the type assosated with pokemon matches.                -%
%- dump_kind(T) :: takes a parameter type(T) and dumps all the pokemon of thats type and all of their attributes.                          -%
%- family(Name) ::  takes a cen name  as a parameter and returns the evolutioary family.                                                    -%
%- familiess :: returns all of the cen pokemon and thier evolutions.                                                                        -%
%- lineage(Name) :: takes a parameter (Name) and returns that pokemon, all its evolutoios, and the metadata assoctiated with each one      -%
%-------------------------------------------------------------------------------------------------------------------------------------------%

display_cen :- (cen(Name),write(Name),nl,fail).
display_not_cen :- (evolves(_,Q),write(Q),nl,fail).
generator(N,T) :- pokemon(name(N),T,_,_).
display_names :- pokemon(name(N),_,_,_),write(N),nl,fail.
display_attacks :- pokemon(_,_,_,attack(A,_)),write(A),nl,fail.
display_cen_attacks :- (cen(Name),pokemon(name(Name),_,_,attack(A,_)),write(A),nl,fail).
indicate_attack(N) :- pokemon(name(N),_,_,attack(A,_)),write(N),write(' --> '),write(A),nl,fail.
indicate_attacks :- pokemon(name(N),_,_,attack(A,_)),write(N),write(' --> '),write(A),nl,fail.
powerful(N) :- pokemon(name(N),_,_,attack(_,D)),D > 55.
tough(N) :- pokemon(name(N),_,hp(HP),_),HP > 100.
awesome(N) :- powerful(N),tough(N).
powerful_but_vulnerable(N) :- powerful(N), pokemon(name(N),_,hp(HP),_),HP < 101.
type(N,T) :- pokemon(name(N),T,_,_).
dump_kind(T) :- pokemon(name(N),T,hp(HP),attack(A,D)),write('pokemon(name('),write(N),write('),'),write(T),write(',hp('),write(HP),write(')
,attack('),write(A),write(','),write(D),write('))'),nl,fail.
family(Name) :- cen(Name), (evolves(Name,Y), write(Name), write(' '), write(Y)), evolves(Y,Z), write(' '), write(Z).
families :- cen(Name),nl,family(Name),fail.
lineage(Name) :- pokemon(name(Name),Type,HP,Attack), write('pokemon(name('),write(Name), write('),'),write(Type),write(','),write(HP),write(',')
,write(Attack),write(')'), evolves(Name,Evolution),nl,lineage(Evolution).
```

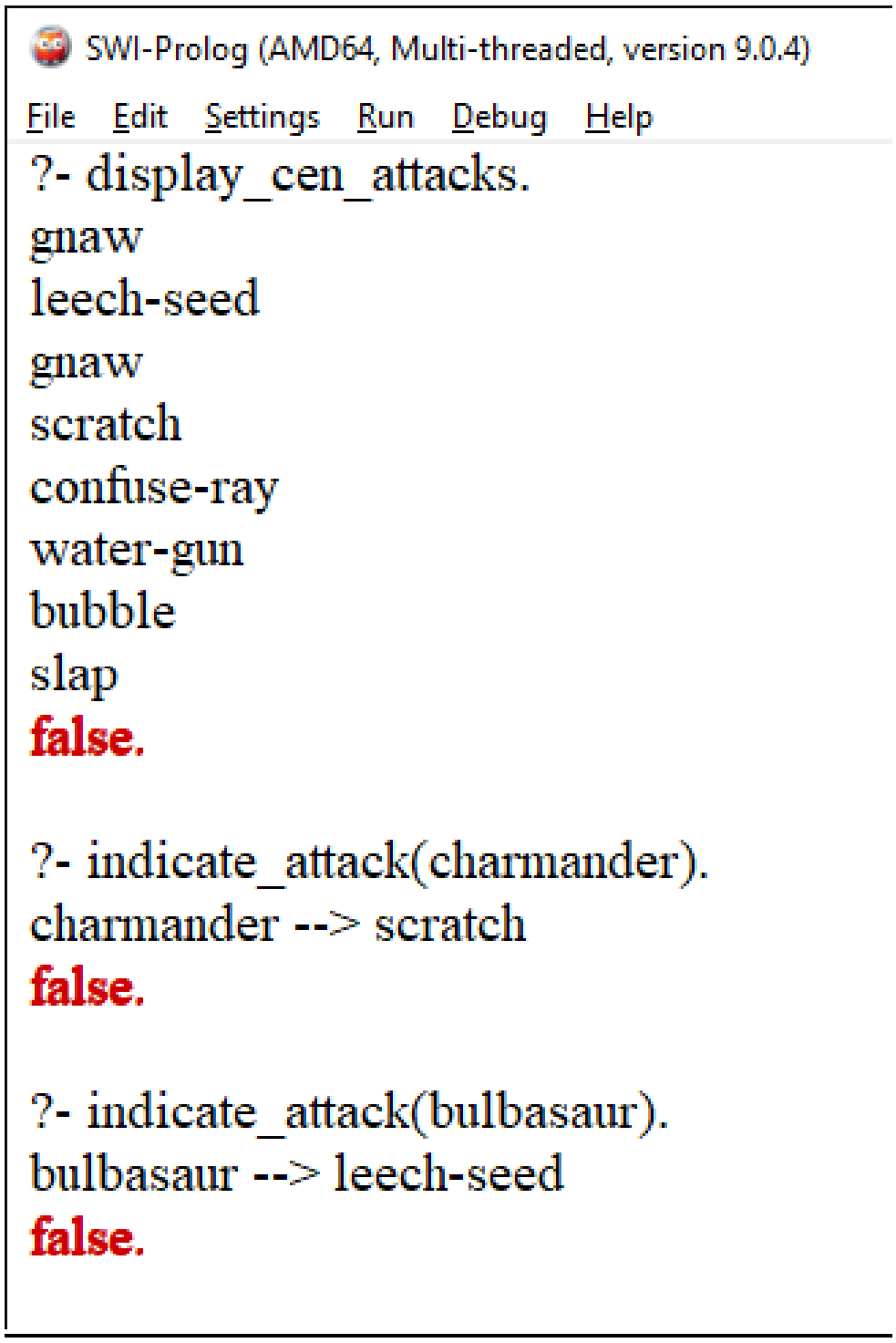
Part 4: Interaction demo with the Augmented KB

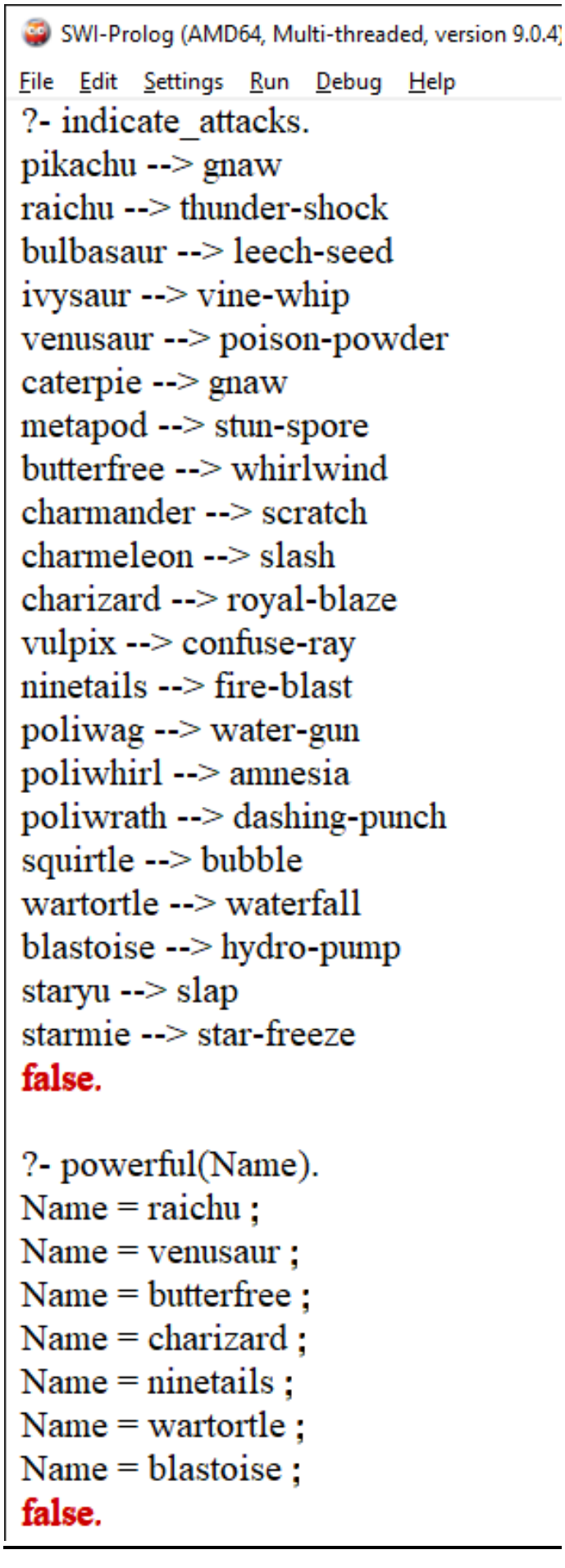



SWI-Prolog (AMD64, Multi-threaded, version 9.
File Edit Settings Run Debug Help
?- generator(Name,fire).
Name = charmander ;
Name = charmeleon ;
Name = charizard ;
Name = vulpix ;
Name = ninetails.
?- generator(Name,water).
Name = poliwag ;
Name = poliwhirl ;
Name = poliwrath ;
Name = squirtle ;
Name = wartortle ;
Name = blastoise ;
Name = staryu ;
Name = starmie.
?- generator(Name,electric).
Name = pikachu ;
Name = raichu.
?- generator(Name,grass).
Name = bulbasaur ;
Name = ivysaur ;
Name = venusaur ;
Name = caterpie ;
Name = metapod ;
Name = butterfree.


SWI-Prolog (AMD64, Multi-t
File Edit Settings Run De
?- display_names.
pikachu
raichu
bulbasaur
ivysaur
venusaur
caterpie
metapod
butterfree
charmander
charmeleon
charizard
vulpix
ninetails
poliwag
poliwhirl
poliwrath
squirtle
wartortle
blastoise
staryu
starmie
false.


SWI-Prolog (AMD64, Multi-threaded, ver
File Edit Settings Run Debug Help
?- display_attacks.
gnaw
thunder-shock
leech-seed
vine-whip
poison-powder
gnaw
stun-spore
whirlwind
scratch
slash
royal-blaze
confuse-ray
fire-blast
water-gun
amnesia
dashing-punch
bubble
waterfall
hydro-pump
slap
star-freeze
false.


SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)
File Edit Settings Run Debug Help
?- display_cen_attacks.
gnaw
leech-seed
gnaw
scratch
confuse-ray
water-gun
bubble
slap
false.
?- indicate_attack(charmander).
charmander --> scratch
false.
?- indicate_attack(bulbasaur).
bulbasaur --> leech-seed
false.


SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)
File Edit Settings Run Debug Help
?- indicate_attacks.
pikachu --> gnaw
raichu --> thunder-shock
bulbasaur --> leech-seed
ivysaur --> vine-whip
venusaur --> poison-powder
caterpie --> gnaw
metapod --> stun-spore
butterfree --> whirlwind
charmander --> scratch
charmeleon --> slash
charizard --> royal-blaze
vulpix --> confuse-ray
ninetails --> fire-blast
poliwag --> water-gun
poliwhirl --> amnesia
poliwrath --> dashing-punch
squirtle --> bubble
wartortle --> waterfall
blastoise --> hydro-pump
staryu --> slap
starmie --> star-freeze
false.
?- powerful(Name).
Name = raichu ;
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = ninetails ;
Name = wartortle ;
Name = blastoise ;
false.

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- tough(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = poliwrath ;
Name = blastoise ;
false.

?- awesome(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = blastoise ;
false.

?- powerful_but_vulnerable(Name).
Name = raichu ;
Name = ninetails ;
Name = wartortle ;
false.

?- type(squirtle,Type).
Type = water.

?- type(caterpie,Type).
Type = grass.

?- type(Name,fire),write(Name),nl,fail.
charmander
charmeleon
charizard
vulpix
ninetails
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

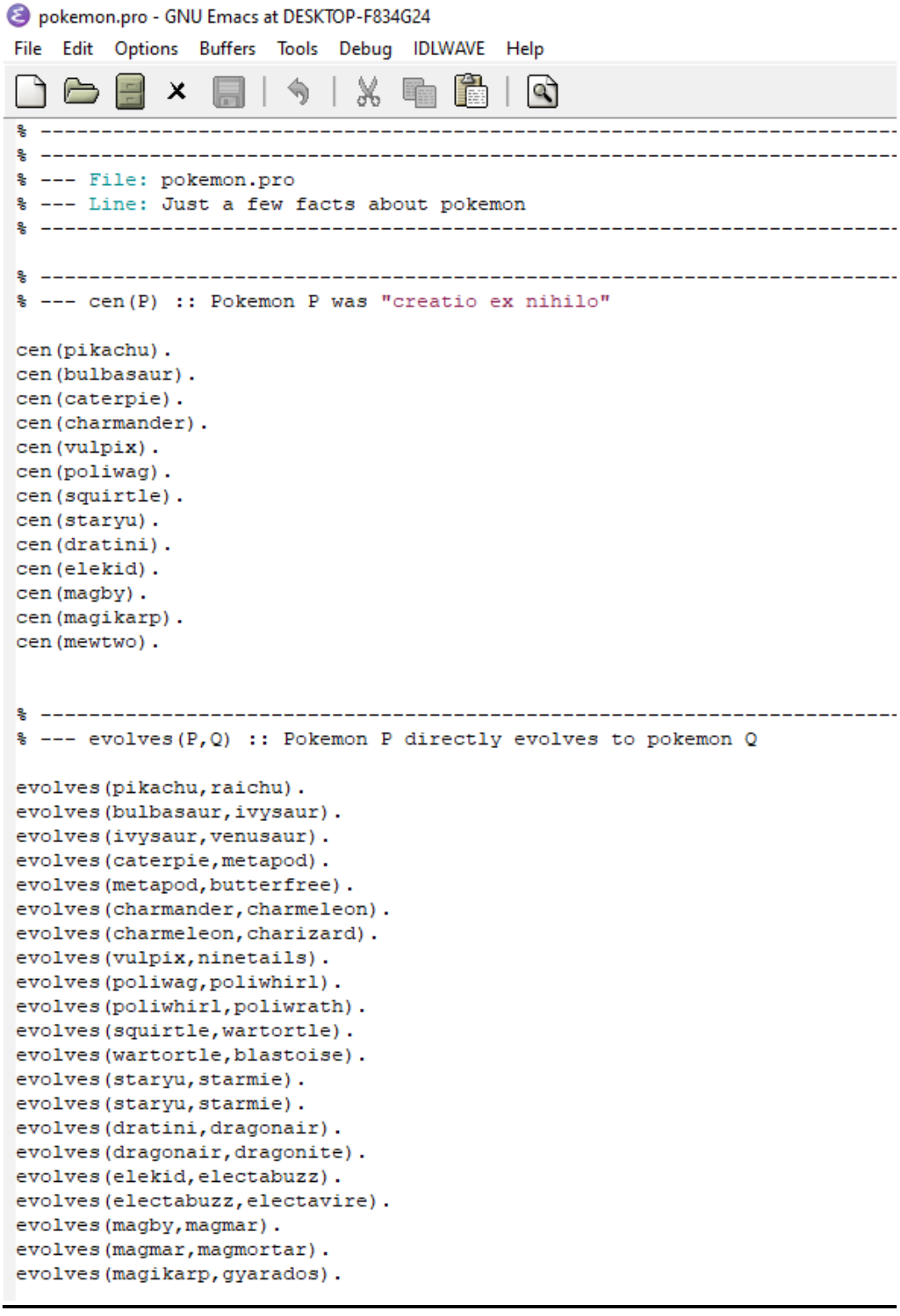
```
?- dump_kind(water).
pokemon(name(poliwag),water,hp(60),attack(water-gun,30))
pokemon(name(poliwhirl),water,hp(80),attack(amnesia,30))
pokemon(name(poliwrath),water,hp(140),attack(dashing-punch,50))
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
pokemon(name(staryu),water,hp(40),attack(slap,20))
pokemon(name(starmie),water,hp(60),attack(star-freeze,20))
false.

?- dump_kind(grass).
pokemon(name(bulbasaur),grass,hp(40),attack(leech-seed,20))
pokemon(name(ivysaur),grass,hp(60),attack(vine-whip,30))
pokemon(name(venusaur),grass,hp(140),attack(poison-powder,70))
pokemon(name(caterpie),grass,hp(50),attack(gnaw,20))
pokemon(name(metapod),grass,hp(70),attack(stun-spore,20))
pokemon(name(butterfree),grass,hp(130),attack(whirlwind,80))
false.

?- family(pikachu).
pikachu raichu
false.

?- family(bulbasaur).
bulbasaur ivysaur venusaur
true.

?- family(caterpie).
caterpie metapod butterfree
true.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- families.

pikachu raichu
bulbasaur ivysaur venusaur
caterpie metapod butterfree
charmander charmeleon charizard
vulpix ninetails
poliwag poliwhirl poliwrath
squirtle wartortle blastoise
staryu starmiestaryu starmie
false.

?- lineage(pikachu).
pokemon(name(pikachu),electric,hp(60),attack(gnaw,10))
pokemon(name(raichu),electric,hp(90),attack(thunder-shock,90))
false.

?- lineage(squirtle).
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
false.

?- lineage(wartortle).
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
false.

?- lineage(blastoise).
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
false.

?- lineage(charmander).
pokemon(name(charmander),fire,hp(50),attack(scratch,10))
pokemon(name(charmeleon),fire,hp(80),attack(slash,50))
pokemon(name(charizard),fire,hp(170),attack(royal-blaze,100))
false.
```

Part 5: KB Augmented by 12 Pokemon:



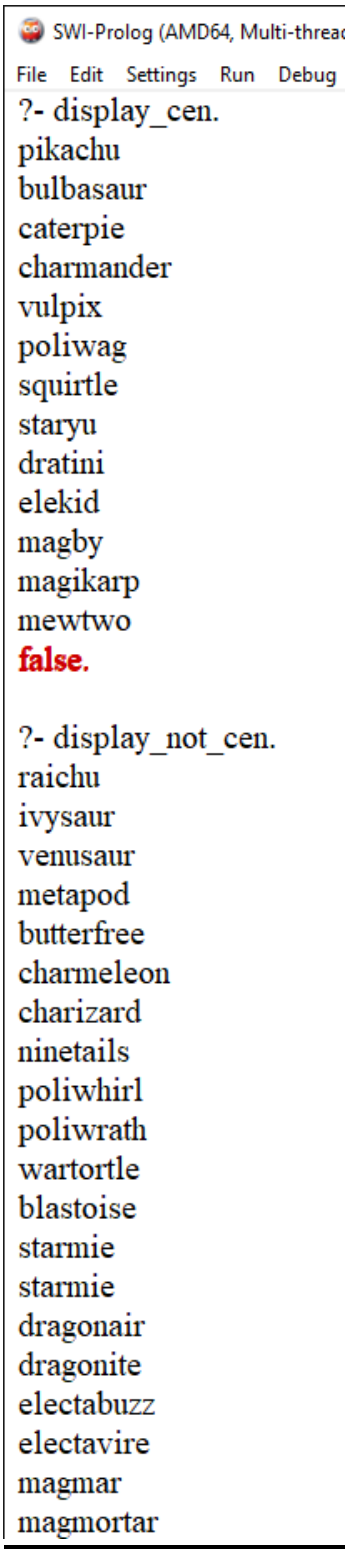

```
% ------------------------------------------------------------------------
% ------------------------------------------------------------------------
% --- File: pokemon.pro
% --- Line: Just a few facts about pokemon
% ------------------------------------------------------------------------

% ------------------------------------------------------------------------
% --- cen(P) :: Pokemon P was "creatio ex nihilo"

cen(pikachu).
cen(bulbasaur).
cen(caterpie).
cen(charmander).
cen(vulpix).
cen(poliwag).
cen(squirtle).
cen(staryu).
cen(dratini).
cen(elekid).
cen(magby).
cen(magikarp).
cen(mewtwo).


% ------------------------------------------------------------------------
% --- evolves(P,Q) :: Pokemon P directly evolves to pokemon Q

evolves(pikachu,raichu).
evolves(bulbasaur,ivysaur).
evolves(ivysaur,venusaur).
evolves(caterpie,metapod).
evolves(metapod,butterfree).
evolves(charmander,charmeleon).
evolves(charmeleon,charizard).
evolves(vulpix,ninetails).
evolves(poliwag,poliwhirl).
evolves(poliwhirl,poliwrath).
evolves(squirtle,wartortle).
evolves(wartortle,blastoise).
evolves(staryu,starmie).
evolves(staryu,starmie).
evolves(dratini,dragonair).
evolves(dragonair,dragonite).
evolves(elekid,electabuzz).
evolves(electabuzz,electavire).
evolves(magby,magmar).
evolves(magmar,magmortar).
evolves(magikarp,gyarados).
```

pokemon.pro - GNU Emacs at DESKTOP-F834G24

File Edit Options Buffers Tools Debug IDLWAVE Help

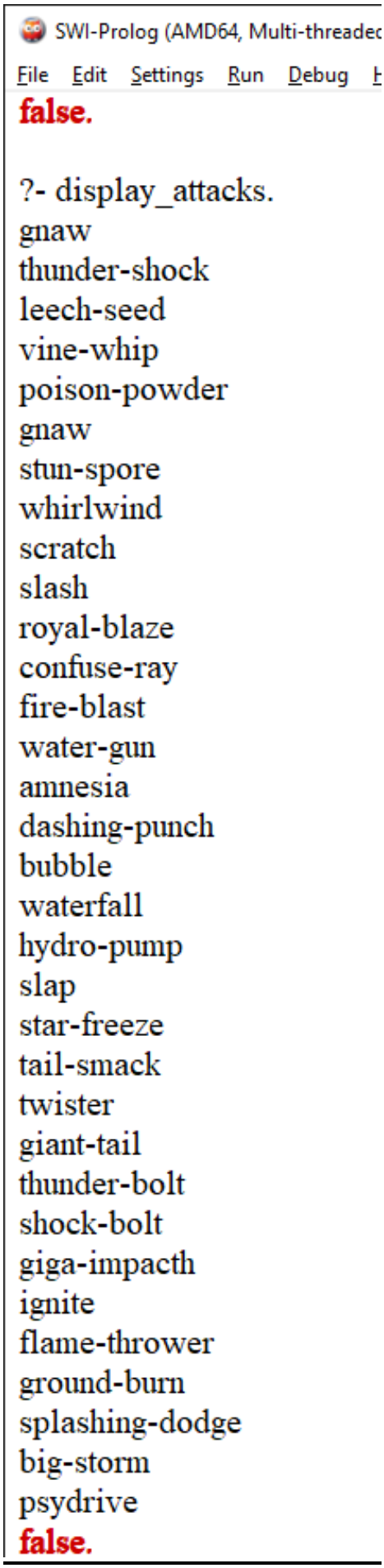
```
% ------------------------------------------------------------------
% --- pokemon(name(N),T,hp(H),attach(A,D)) :: There is a pokemon with
% --- name N, type T, hit point value H, and attach named A that does
% --- damage D.

pokemon(name(pikachu), electric, hp(60), attack(gnaw, 10)).
pokemon(name(raichu), electric, hp(90), attack(thunder-shock, 90)).

pokemon(name(bulbasaur), grass, hp(40), attack(leech-seed, 20)).
pokemon(name(ivysaur), grass, hp(60), attack(vine-whip, 30)).
pokemon(name(venusaur), grass, hp(140), attack(poison-powder, 70)).

pokemon(name(caterpie), grass, hp(50), attack(gnaw, 20)).
pokemon(name(metapod), grass, hp(70), attack(stun-spore, 20)).
pokemon(name(butterfree), grass, hp(130), attack(whirlwind, 80)).

pokemon(name(charmander), fire, hp(50), attack(scratch, 10)).
pokemon(name(charmeleon), fire, hp(80), attack(slash, 50)).
pokemon(name(charizard), fire, hp(170), attack(royal-blaze, 100)).

pokemon(name(vulpix), fire, hp(60), attack(confuse-ray, 20)).
pokemon(name(ninetails), fire, hp(100), attack(fire-blast, 120)).

pokemon(name(poliwag), water, hp(60), attack(water-gun, 30)).
pokemon(name(poliwhirl), water, hp(80), attack(amnesia, 30)).
pokemon(name(poliwrath), water, hp(140), attack(dashing-punch, 50)).

pokemon(name(squirtle), water, hp(40), attack(bubble, 10)).
pokemon(name(wartortle), water, hp(80), attack(waterfall, 60)).
pokemon(name(blastoise), water, hp(140), attack(hydro-pump, 60)).

pokemon(name(staryu), water, hp(40), attack(slap, 20)).
pokemon(name(starmie), water, hp(60), attack(star-freeze, 20)).

pokemon(name(dratini), dragon, hp(60), attack(tail-smack, 10)).
pokemon(name(dragonair), dragon, hp(80), attack(twister, 40)).
pokemon(name(dragonite), dragon, hp(140), attack(giant-tail, 200)).

pokemon(name(elekid), electric, hp(60), attack(thunder-bolt, 30)).
pokemon(name(electabuzz), electric, hp(70), attack(shock-bolt, 60)).
pokemon(name(electavire), electric, hp(140), attack(giga-impacth, 170)).

pokemon(name(magby), fire, hp(40), attack(ignite, 10)).
pokemon(name(magmar), fire, hp(80), attack(flame-thrower, 80)).
pokemon(name(magmortar), fire, hp(140), attack(ground-burn, 160)).

pokemon(name(magikarp), water, hp(30), attack(splashing-dodge, 10)).
pokemon(name(gyarados), water, hp(180), attack(big-storm, 200)).

pokemon(name(mewtwo), psychic, hp(200), attack(psydrive, 180)).
```

Part 6: Interaction Demo With KB Augmented by 12 Pokemon:



```
?- display_names.
pikachu
raichu
bulbasaur
ivysaur
venusaur
caterpie
metapod
butterfree
charmander
charmeleon
charizard
vulpix
ninetails
poliwag
poliwhirl
poliwrath
squirtle
wartortle
blastoise
staryu
starmie
dratini
dragonair
dragonite
elekid
electabuzz
electavire
magby
magmar
magmortar
magikarp
gyarados
mewtwo
false.
```

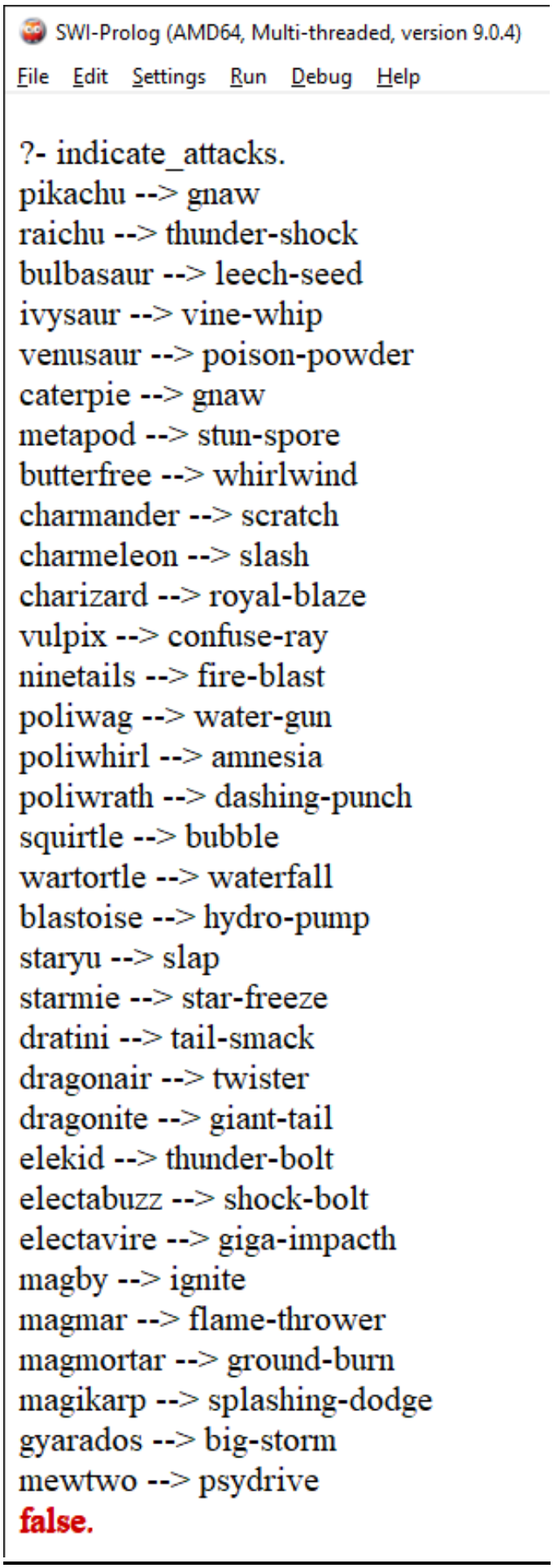
```
?- generator(Name,fire).
Name = charmander ;
Name = charmeleon ;
Name = charizard ;
Name = vulpix ;
Name = ninetails ;
Name = magby ;
Name = magmar ;
Name = magmortar.

?- generator(Name,water).
Name = poliwag ;
Name = poliwhirl ;
Name = poliwrath ;
Name = squirtle ;
Name = wartortle ;
Name = blastoise ;
Name = staryu ;
Name = starmie ;
Name = magikarp ;
Name = gyarados.
```

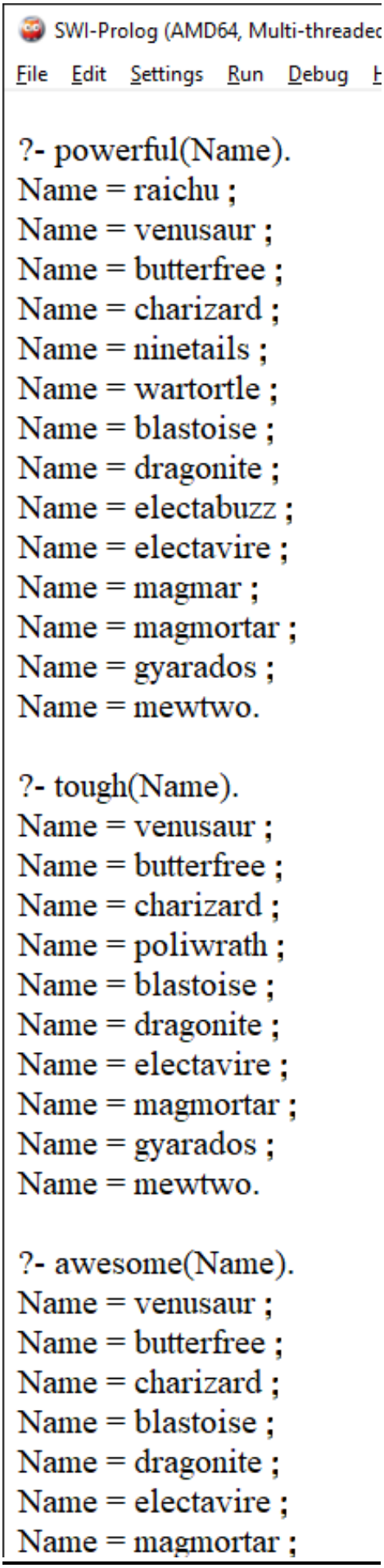

SWI-Prolog (AMD64, Multi-threaded
File Edit Settings Run Debug
false.
?- display_attacks.
gnaw
thunder-shock
leech-seed
vine-whip
poison-powder
gnaw
stun-spore
whirlwind
scratch
slash
royal-blaze
confuse-ray
fire-blast
water-gun
amnesia
dashing-punch
bubble
waterfall
hydro-pump
slap
star-freeze
tail-smack
twister
giant-tail
thunder-bolt
shock-bolt
giga-impacth
ignite
flame-thrower
ground-burn
splashing-dodge
big-storm
psydrive
false.

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- display_cen_attacks.
gnaw
leech-seed
gnaw
scratch
confuse-ray
water-gun
bubble
slap
tail-smack
thunder-bolt
ignite
splashing-dodge
psydrive
false.

?- indicate_attack(gyarados).
gyarados --> big-storm
false.

?- indicate_attack(mewtwo).
mewtwo --> psydrive
false.
```


SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)
File Edit Settings Run Debug Help
?- indicate_attacks.
pikachu --> gnaw
raichu --> thunder-shock
bulbasaur --> leech-seed
ivysaur --> vine-whip
venusaur --> poison-powder
caterpie --> gnaw
metapod --> stun-spore
butterfree --> whirlwind
charmander --> scratch
charmeleon --> slash
charizard --> royal-blaze
vulpix --> confuse-ray
ninetails --> fire-blast
poliwag --> water-gun
poliwhirl --> amnesia
poliwrath --> dashing-punch
squirtle --> bubble
wartortle --> waterfall
blastoise --> hydro-pump
staryu --> slap
starmie --> star-freeze
dratini --> tail-smack
dragonair --> twister
dragonite --> giant-tail
elekid --> thunder-bolt
electabuzz --> shock-bolt
electavire --> giga-impacth
magby --> ignite
magmar --> flame-thrower
magmortar --> ground-burn
magikarp --> splashing-dodge
gyarados --> big-storm
mewtwo --> psydrive
false.


SWI-Prolog (AMD64, Multi-threadec
File Edit Settings Run Debug
?- powerful(Name).
Name = raichu ;
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = ninetails ;
Name = wartortle ;
Name = blastoise ;
Name = dragonite ;
Name = electabuzz ;
Name = electavire ;
Name = magmar ;
Name = magmortar ;
Name = gyarados ;
Name = mewtwo.
?- tough(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = poliwrath ;
Name = blastoise ;
Name = dragonite ;
Name = electavire ;
Name = magmortar ;
Name = gyarados ;
Name = mewtwo.
?- awesome(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = blastoise ;
Name = dragonite ;
Name = electavire ;
Name = magmortar ;

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- powerful_but_vulnerable(Name).
Name = raichu ;
Name = ninetails ;
Name = wartortle ;
Name = electabuzz ;
Name = magmar ;
false.

?- type(mewtwo,Type).
Type = psychic.

?- type(elekid,Type).
Type = electric.

?- type(Name,fire),write(Name),nl,fail.
charmander
charmeleon
charizard
vulpix
ninetails
magby
magmar
magmortar
false.

?- dump_kind(water).
pokemon(name(poliwag),water,hp(60),attack(water-gun,30))
pokemon(name(poliwhirl),water,hp(80),attack(amnesia,30))
pokemon(name(poliwrath),water,hp(140),attack(dashing-punch,50))
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
pokemon(name(staryu),water,hp(40),attack(slap,20))
pokemon(name(starmie),water,hp(60),attack(star-freeze,20))
pokemon(name(magikarp),water,hp(30),attack(splashing-dodge,10))
pokemon(name(gyarados),water,hp(180),attack(big-storm,200))
false.
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- dump_kind(grass).
pokemon(name(bulbasaur),grass,hp(40),attack(leech-seed,20))
pokemon(name(ivysaur),grass,hp(60),attack(vine-whip,30))
pokemon(name(venusaur),grass,hp(140),attack(poison-powder,70))
pokemon(name(caterpie),grass,hp(50),attack(gnaw,20))
pokemon(name(metapod),grass,hp(70),attack(stun-spore,20))
pokemon(name(butterfree),grass,hp(130),attack(whirlwind,80))
false.

?- family(magikarp).
magikarp gyarados
false.

?- family(elekid).
elekid electabuzz electavire
true.
```

```
?- family(magby).
magby magmar magmortar
true.

?-
```

?- families.

pikachu raichu
bulbasaur ivysaur venusaur
caterpie metapod butterfree
charmander charmeleon charizard
vulpix ninetails
poliwag poliwhirl poliwrath
squirtle wartortle blastoise
staryu starmiestaryu starmie
dratini dragonair dragonite
elekid electabuzz electavire
magby magmar magmortar
magikarp gyarados
false.

?- lineage(magikarp).
pokemon(name(magikarp),water,hp(30),attack(splashing-dodge,10))
pokemon(name(gyarados),water,hp(180),attack(big-storm,200))
false.

?- lineage(elekid).
pokemon(name(elekid),electric,hp(60),attack(thunder-bolt,30))
pokemon(name(electabuzz),electric,hp(70),attack(shock-bolt,60))
pokemon(name(electavire),electric,hp(140),attack(giga-impacth,170))
false.

?- lineage(magby).
pokemon(name(magby),fire,hp(40),attack(ignite,10))
pokemon(name(magmar),fire,hp(80),attack(flame-thrower,80))
pokemon(name(magmortar),fire,hp(140),attack(ground-burn,160))
false.

?- lineage(dratini).
pokemon(name(dratini),dragon,hp(60),attack(tail-smack,10))
pokemon(name(dragonair),dragon,hp(80),attack(twister,40))
pokemon(name(dragonite),dragon,hp(140),attack(giant-tail,200))

Task 2 – List Processing

Head/Tail Exercises:

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)
File Edit Settings Run Debug Help

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- [H|T] = [red, yellow, blue, green].
H = red,
T = [yellow, blue, green].

?- [H, T] = [red, yellow, blue, green].
false.

?- [F|_] = [red, yellow, blue, green].
F = red.

?- [_|[S|_]] = [red, yellow, blue, green].
S = yellow.

?- [F|[S|R]] = [red, yellow, blue, green].
F = red,
S = yellow,
R = [blue, green].

?- List = [this|[and, that]].
List = [this, and, that].

?- List = [this, and, that].
List = [this, and, that].

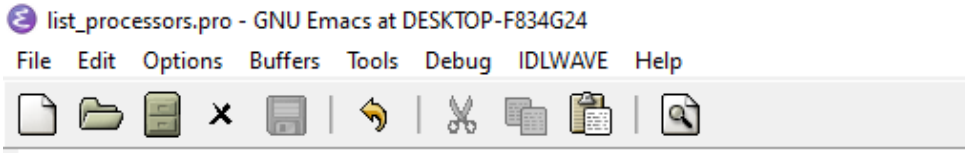
?- [a,[b, c]] = [a, b, c].
false.

?- [a|[b, c]] = [a, b, c].
true.

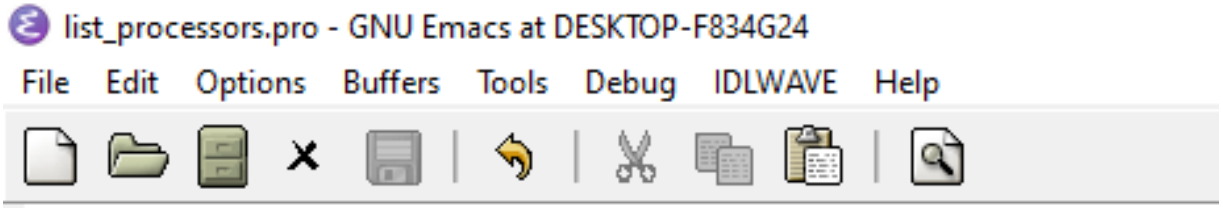
?- [cell(Row,Column)|Rest] = [cell(1,1), cell(3,2), cell(1,3)].
Row = Column, Column = 1,
Rest = [cell(3, 2), cell(1, 3)].

?- [X|Y] = [one(un, uno), two(dos, deux), three(trois, tres)].
X = one(un, uno),
Y = [two(dos, deux), three(trois, tres)].

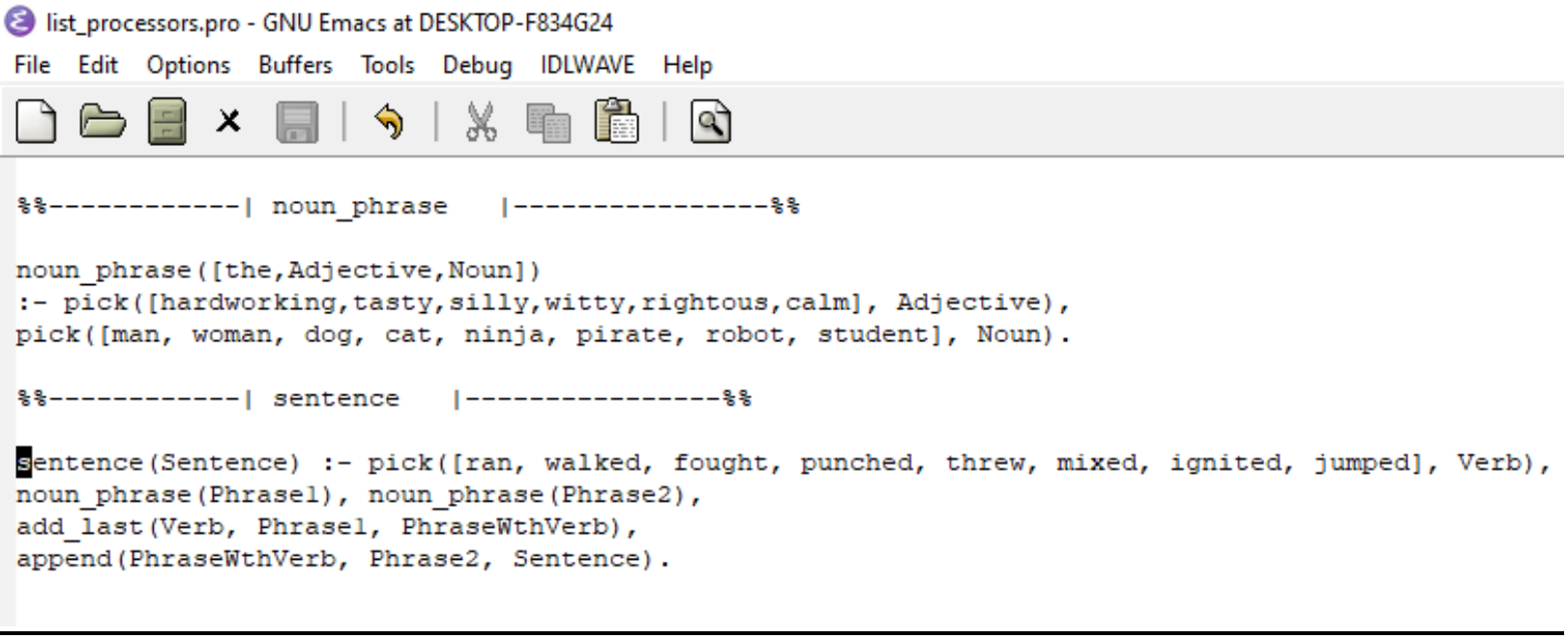
List Processing Code:




```
%%------------------------------------------------------------%%
%%- list_processers.pro |                                     %%
%%----------------------                                      %%
%% Some code to process some lists in prolog.                 %%
%%                                                            %%
%%------------------------------------------------------------%%

%%-------------------------------------------------%%
%%----------------|   Code   |-------------------%%
%%-------------------------------------------------%%


%-----------------|  First   |-------------------%%

first([H|_], H).

%-----------------|   Rest   |-------------------%%

rest([_|T], T).

%-----------------|   Last   |-------------------%%

last([H|[]], H).
last([_|T], Result) :- last(T, Result).

%-----------------|    Nth   |-------------------%%

nth(0,[H|_],H).
nth(N,[_|T],E) :- K is N - 1, nth(K,T,E).

%-----------------| Writelist |-----------------%%

writelist([]).
writelist([H|T]) :- write(H), nl, writelist(T).

%-----------------|    Sum    |-----------------%%

sum([],0).
sum([Head|Tail],Sum) :-
sum(Tail,SumOfTail),
Sum is Head + SumOfTail.

%-----------------|  Add_first |----------------%%

add_first(X,L,[X|L]).

%-----------------|  Add_last  |----------------%%

add_last(X,[],[X]).
add_last(X,[H|T],[H|TX]) :- add_last(X,T,TX).

%-----------------|   Iota     |----------------%%

iota(0,[]).
iota(N,IotaN) :-
K is N - 1,
iota(K,IotaK),
```




```
%-----------------|   Pick     |----------------%%

pick(L,Item) :-
length(L,Length),
random(0,Length,RN),
nth(RN,L,Item).

%-----------------|  Make_set  |----------------%%

make_set([],[]).
make_set([H|T],TS) :-
member(H,T),
make_set(T,TS).
make_set([H|T],[H|TS]) :-
make_set(T,TS).

%-----------------|  Product   |----------------%%

product([],1).
product([Head|Tail],Product) :-
product(Tail,ProductOfTail),
Product is Head * ProductOfTail.


%-----------------| Factotial |-------------------%%

factorial(N,R) :-
iota(N,K),
product(K,R).

%%---------------| make_list |-------------------%%

make_list(0, _, []).
make_list(N, E, [E|ResultR]) :-
K is N - 1,
make_list(K, E, ResultR).
```


list_processors.pro - GNU Emacs at DESKTOP-F834G24
File Edit Options Buffers Tools Debug IDLWAVE Help
%%------------| noun_phrase |----------------%%
noun_phrase([the,Adjective,Noun])
:- pick([hardworking,tasty,silly,witty,rightous,calm], Adjective),
pick([man, woman, dog, cat, ninja, pirate, robot, student], Noun).
%%------------| sentence |----------------%%
sentence(Sentence) :- pick([ran, walked, fought, punched, threw, mixed, ignited, jumped], Verb),
noun_phrase(Phrase1), noun_phrase(Phrase2),
add_last(Verb, Phrase1, PhraseWthVerb),
append(PhraseWthVerb, Phrase2, Sentence).

Demo for Example List Processors:

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.4)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- consult('C:/Users/User/Documents/Prolog/prolog/list_processors.pro').
true.

?- first([apple],First).
First = apple.

?- first([c,d,e,f,g,a,b],P).
P = c.

?- rest([apple],Rest).
Rest = [].

?-  rest([c,d,e,f,g,a,b],Rest).
Rest = [d, e, f, g, a, b].

?-  last([peach],Last).
Last = peach .

?- last([c,d,e,f,g,a,b],P).
P = b .

?- nth(0,[zero,one,two,three,four],Element).
Element = zero .

?- nth(3,[four,three,two,one,zero],Element).
Element = one .
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

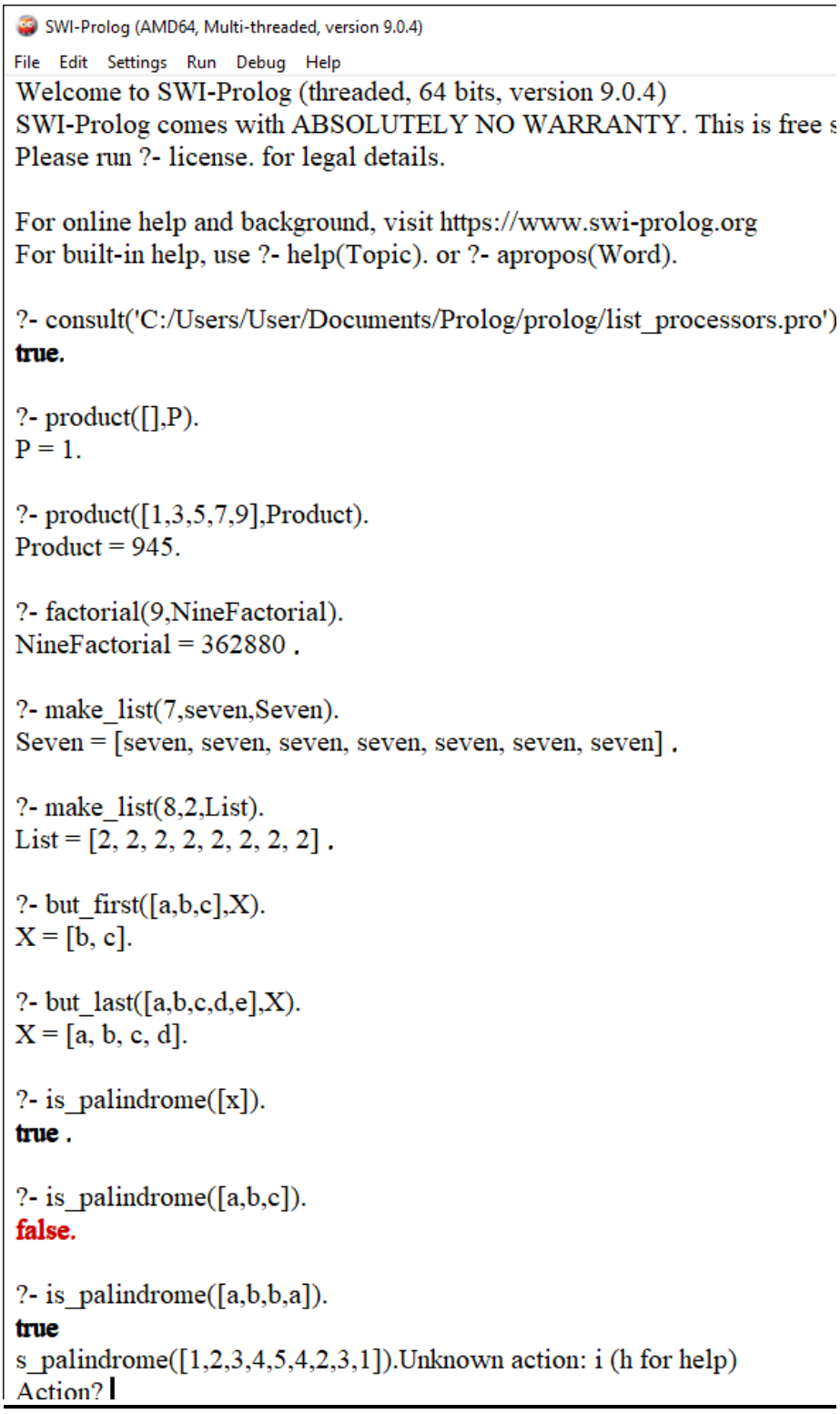
```
?- sum([],Sum).
Sum = 0.

?-  sum([2,3,5,7,11],SumOfPrimes).
SumOfPrimes = 28.

?- add_first(thing,[],Result).
Result = [thing].

?- add_first(racket,[prolog,haskell,rust],Languages).
Languages = [racket, prolog, haskell, rust].

?- add_last(thing,[],Result).
Result = [thing] .

?- add_last(rust,[racket,prolog,haskell],Languages).
Languages = [racket, prolog, haskell, rust] .

?- iota(5,Iota5).
Iota5 = [1, 2, 3, 4, 5] .

?- iota(9,Iota9).
Iota9 = [1, 2, 3, 4, 5, 6, 7, 8, 9] .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = blueberry .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = cherry [print]
ick([cherry,peach,apple,blueberry],Pie).Pie = cherry
Unknown action: (h for help)
Action?
Unknown action: (h for help)
Action?
Unknown action: (h for help)
Action? .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = cherry .
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- pick([cherry,peach,apple,blueberry],Pie).
Pie = cherry .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = peach .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = cherry .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = apple .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = apple .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = cherry .

?- pick([cherry,peach,apple,blueberry],Pie).
Pie = blueberry .

?-  make_set([1,1,2,1,2,3,1,2,3,4],Set).
Set = [1, 2, 3, 4] .

?- make_set([bit,bot,bet,bot,bot,bit],B).
B = [bet, bot, bit]
```

Demo List Processing Exercises:



SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- is_palindrome([1,2,3,4,5,4,2,3,1]).
false.

?- is_palindrome([c,o,f,f,e,e,e,e,f,f,o,c]).
true .

?- noun_phrase(NP).
NP = [the, hardworking, man] .

?- noun_phrase(NP).
NP = [the, silly, pirate] .

?- noun_phrase(NP).
NP = [the, silly, pirate] .

?- noun_phrase(NP).
NP = [the, silly, pirate] .

?- sentence(S).
S = [the, silly, pirate, threw, the, calm, man] .

?- sentence(S).
S = [the, silly, robot, ignited, the, silly, student] .

?- sentence(S).
S = [the, hardworking, woman, punched, the, tasty, woman] .

?- sentence(S).
S = [the, tasty, cat, fought, the, rightous, robot] .

?- sentence(S).
S = [the, tasty, robot, ignited, the, silly, man] .

?- sentence(S).
S = [the, rightous, man, ignited, the, hardworking, man] .

?- sentence(S).
S = [the, tasty, student, walked, the, tasty, dog] .

?- sentence(S).
```

SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

File Edit Settings Run Debug Help

```
?- sentence(S).
S = [the, rightous, cat, fought, the, tasty, student] .

?- sentence(S).
S = [the, witty, robot, jumped, the, silly, dog] .

?- sentence(S).
S = [the, calm, robot, jumped, the, silly, student] .

?- sentence(S).
S = [the, tasty, ninja, punched, the, silly, man] .

?- sentence(S).
S = [the, witty, dog, jumped, the, calm, woman] .

?- sentence(S).
S = [the, silly, dog, fought, the, rightous, woman] .

?- sentence(S).
S = [the, calm, man, mixed, the, tasty, robot] .

?- sentence(S).
S = [the, silly, student, jumped, the, tasty, woman] .

?-
```