

```
person(wolf).
person(nafnaf).
person(nifnif).
person(nufnuf).
```

```
theme(fauna).
theme(flora).
theme(sport).
theme(space).
```

```
collects(wolf, fauna). % Волк собирает марки с фауной
```

```
collects(nafnaf, space). % Наф Наф собирает марки с космосом (по условию)
collects(nifnif, sport). % Ниф Ниф собирает марки со спортом (по условию)
collects(nufnuf, flora). % Нуф Нуф собирает марки с флорой (по условию)
```

```
seated_order([Left, Center, Right, Opposite]) :-
    Left = wolf,      % Волк слева от Наф Нафа (условие 1)
    Center = nafnaf,  % Центр — Наф Наф
    Right = nifnif,   % Справа от Центра сидит Ниф Ниф
    Opposite = nufnuf. % Напротив Наф Нафа сидит Нуф Нуф (условие 3)
```

```
find_theme_collectors([Left, Center, Right, Opposite]) :-
    collects(Center, space), % Центр (Наф Наф) собирает марки с космосом
    Right = nifnif.         % Справа от него сидит Ниф Ниф
```

```
no_sport_for_nufnuf :-
    collects(nufnuf, Theme),
    Theme \= sport.
```

```
solve :-
    seated_order(Seating),
    find_theme_collectors(Seating),
    no_sport_for_nufnuf,
    format("Порядок за столом по часовой стрелке: ~w", [Seating]),
    nl,
    collects(nafnaf, NafNafTheme),
    collects(nifnif, NifNifTheme),
    collects(nufnuf, NufNufTheme),
    format("Наф Наф собирает марки: ~w", [NafNafTheme]),
    nl,
    format("Ниф Ниф собирает марки: ~w", [NifNifTheme]),
    nl,
    format("Нуф Нуф собирает марки: ~w", [NufNufTheme]).
```

```
:- solve.
```