# (P) A Matter of Horn Clauses (1/3)

A Horn clause, named for logician Alfred Horn, is a notation used in mathematics and in logic programs such as Prolog. Horn clauses offer a flexible way to write the rules of grammar for a language. This problem will introduce you to Horn clause notation and ask you to use the notation to describe English and Swiss German.

Let's start with English, since you already know it. To capture a simple fragment of English, we might say that a sentence consists of a noun followed by a verb. If we write *S* to mean sentence, *N* to mean noun, and *V* to mean verb, the following Horn clause captures this intuition:

*S*(*xy*) :- *N*(*x*), *V*(*y*).

This rule says that if we have a noun *x* and a verb *y*, we can make a sentence by putting *x* and *y* together in that order. Horn clauses with the ':-' symbol are called rules, and they tell us how to derive the thing on the left side of the ':-' from the things on the right side of the ':-'. Note that the labels *S*, *N*, and *V* don't affect how the rule behaves; they are simply chosen to help us remember what we're representing.

However, so far we haven't given ourselves any nouns or verbs, so we can't make a sentence. The following Horn clauses give us nouns and verbs to work with:

*N*(*Mary*).          *N*(*John*).          *V*(*eats*).          *V*(*sleeps*).

For example, the first clause says that "Mary" is a noun. Horn clauses without the ':-' symbol are called facts, because they tell us things that we know are true without doing any work.

Using our facts and our lone rule, we can derive the following sentences:

*S*(Mary eats).          *S*(Mary sleeps).          *S*(John sleeps).          *S*(John eats).

We can extend our grammar to account for subject-verb agreement in English. *sg* means singular and *pl* means plural.

*S*(*xy*) :- *Nsg*(*x*), *Vsg*(*Y*).          *S*(*xy*) :- *Npl*(*x*), *Vpl*(*Y*).
*Nsg*(Mary).          *Npl*(dogs).          *Vsg*(sleeps).          *Vpl*(sleep).

Note that we can derive the sentences *S*(Mary sleeps) and *S*(dogs sleep), but because we have no way to put an *Nsg* together with a *Vpl*, we can't derive *S*(Mary sleep).

**P1.** The rules above can only generate a fixed, finite number of sentences, but there is no clear upper limit on the length of grammatical English sentences. For example, consider the following sentences:

We helped Mary <u>help John</u> paint the house.
We helped Mary <u>help John help Kim</u> paint the house.
We helped Mary <u>help John help Kim help John</u> paint the house.
We let Mary <u>let John let Kim</u> paint the house.
We let Mary <u>help John let Kim</u> paint the house.
We let Mary <u>help John help Kim let Mary help John let Mary</u> paint the house.
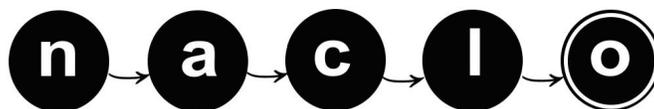
# (P) A Matter of Horn Clauses (2/3)

Clearly we can keep extending these sentences as long as we want; they will still be grammatical, even if they are a bit awkward.

To make things easier for you, we only want you to account for the underlined parts of the sentences. It's easy but tedious to extend the grammar to account for the entire sentences. Write a set of rules and facts that will generate all the possible combinations of "help", "let", "John", and "Kim" that will fit in the sentences above. For example, you should be able to derive *S*(help John let Kim let John).

**P2.** Let's look at similar sentences in Swiss German:

Jan säit das mer em Hans <u>em Jan</u> es huus hälfed <u>hälfe</u> aastriiche.
Jan says that we helped Hans <u>help Jan</u> paint the house.

Jan säit das mer em Hans <u>em Jan em Hans</u> es huus hälfed <u>hälfe hälfe</u> aastriiche.
Jan says that we helped Hans <u>help Jan help Hans</u> paint the house.

Jan säit das mer em Hans <u>em Jan em Hans em Jan</u> es huus hälfed <u>hälfe hälfe hälfe</u> aastriiche.
Jan says that we helped Hans <u>help Jan help Hans help Jan</u> paint the house.

Jan säit das mer de Hans <u>de Jan</u> es huus lönd <u>laa</u> aastriiche.
Jan says that we let Hans <u>let Jan</u> paint the house.

Jan säit das mer de Hans <u>em Jan de Hans</u> es huus lönd <u>hälfe laa</u> aastriiche.
Jan says that we let Hans <u>help Jan let Hans</u> paint the house.

Jan säit das mer de Hans <u>em Jan em Hans de Jan em Hans de Jan</u> es huus lönd <u>hälfe hälfe laa hälfe laa</u> aastriiche.
Jan says that we let Hans <u>help Jan help Hans let Jan help Hans let Jan</u> paint the house.

# (P) A Matter of Horn Clauses (3/3)

It turns out that the formalism described above cannot generate the Swiss German data. However, a simple extension can. Instead of manipulating a single phrase or sentence, we allow ourselves to manipulate a pair of phrases or sentences:

$R(wy, xz)$ :- $T(w, x), T(y, z)$.

This says that if the pair $(w, x)$ is a $T$ (whatever that may be), and the pair $(y, z)$ is also a $T$, then the pair $(wy, xz)$ is an $R$ (whatever that may be). At the end of the day, we can smash the pair into a single sentence:

$S(xy)$ :- $R(x, y)$.

For example, suppose we add the fact $T$(the, cat). Then the first rule lets us derive $R$(the the, cat cat), and the second rule lets us derive $S$(the the cat cat).

Use this extension to describe the Swiss German data. Again, to make your life easier, you only need to generate the underlined part of the sentences. For example, you should be able to derive $S$(de Hans em Jan em Hans laa hälfe hälfe).