The *trill* family of functions is used to play notes with trill (a cyclical varying of pitch at the end of a note). The *trill* family follows the \{n,r,d\} convention.

```c
void trill(double beats,int, instrument,int octave,int pitch,
             double startBeats,double delta,double down,double up,int count);

void ntrill(double beats,int instrument,int numberedNote,
              double startBeats,double delta,double down,double up,int count);

void rtrill(double beats,RRA *r,
              double startBeats,double delta,double down,double up,int count);

void dtrill(double beats,int *data,int length,
             double startBeats,double delta,double down,double up,int count);
```

The note plays for `startBeats` and then oscillates 'count' times around `down` and `up`. Each down phase lasts `delta` beats and each up phase lasts `delta` beats. Down and up are specified as offsets. For example, if the specified note is C3, `down` is -STEP and `up` is STEP, then the note will start at C3 and then oscillate between B2 and Cs3.

A trill is usually considered a rapid alternation between adjacent notes; the trill function provides more generality than that.