A series $\sum_{k\geq 0} s_k$ is called *hypergeometric* when $s_0 = 1$ and the ratio $\frac{s_{k+1}}{s_k}$ is a rational function of k, that is

$$\frac{s_{k+1}}{s_k} = \frac{P(k)}{Q(k)}, \tag{1}$$

for some polynomials P and Q in k. In this case the terms s_k are called *hypergeometric terms.*

> From [1]: It is important to recognize when a given series is hypergeometric, if it is, because the general theory of hypergeometric functions is very powerful, and we may gain a lot of insight into a function that concerns us by first recognizing that it is hypergeometric, then identifying precisely which hyper-geometric function it is, and finally by using known results about such functions.

Suppose that the polynomials P and Q have been completely factored, in the form

$$\frac{s_{k+1}}{s_k} = \frac{P(k)}{Q(k)} = \frac{(a_1 + k)\cdots(a_p + k)}{(b_1 + k)\cdots(b_q + k)(1 + k)}\, x \tag{2}$$

where x is a constant, and normalized such that $s_0 = 1$, then the hypergeometric series $\sum_{k\geq 0} s_k\, x^k$ is denoted by the symbolic form

$${}_pF_q \left[\begin{matrix} a_1 & a_2 & \cdots & a_p \\ b_1 & b_2 & \cdots & b_q \end{matrix}; x\right]. \tag{3}$$

The factor $(1 + k)$ is for historical reasons and there is no loss of generality if it is artificially inserted in case of absence as long as it cancels out.

The hypergeometric series lookup algorithm. The algortithm is described in [1], and is essentially based on factoring the polynomials in order ot read off the symbol pF_q.

Exported functions from package `HYPGEOM`.

```
-- Example 3.6.2
f8(k) == (-1)^k * binomial(2*n,k)*binomial(2*k,k)*binomial(4*n-2*k,2*n-k)

                                                            Type: Void
ff8:=factoredForm(f8)$HYPGEOM

   Compiling function f8 with type Expression(Fraction(Integer)) ->
      Expression(Fraction(Integer))

               2      1
        (? - 2 n) (? + -)
                      2
   (27)  ---------------------
```

```
                 1        2
         (? - 2 n + -)(? + 1)
                 2
Type: Fraction(Factored(SparseUnivariatePolynomial
(SparseMultivariatePolynomial(Fraction(Integer),
Kernel(Expression(Fraction(Integer)))))))

pFq(f8) -->


              1                                  1
   (28)  [ap = [-, - 2 n, - 2 n], bq = [1, - 2 n + -], fac = 1]
              2                                  2
Type: Record(ap: List(Expression(Fraction(Integer))),

bq: List(Expression(Fraction(Integer))),fac: Expression(Fraction(Integer)))
(29) -> display %

              +1                   +
              |-    - 2 n    - 2 n|
              |2                   |
   (29)   F   |                    | [1]
         3  2 |            1       |
              |1  - 2 n + -     *  |
              +            2       +
                                                      Type: OutputForm
(30) -> latex %
```

$$ {}_3F_2 \left[\begin{array}{ccc} \frac{1}{2} & -2\ n & -2\ n \\ 1 & -2\ n + \frac{1}{2} & * \end{array} \right] [1] $$

References

[1] Marko Petkovsek, Herbert Wilf, and Doron Zeilberger, *A=B*, Routledge & CRC Press, URL:https://www2.math.upenn.edu/˜wilf/AeqB.pdf.