INSTRUCTIONS

This packet contains four programs: `CF`, `HF_SFS`, `CW` and `RIB_OBS`. For each of these, the underlying code (`*.py`) as well as a more user-friendly version with a minimalistic interface is included; the latter can be run by running `python3 *_user_dialogue.py` in a terminal. These programs require Python 3.8, and the Python library `SymPy` should be installed (also see Pipfiles). All files should be located in the same directory.

Instructions for `CF`. The functions `number_to_neg_cont_frac` and `number_to_pos_cont_frac` compute (the coefficients of) the negative and positive continued fraction expansions, respectively, of a rational number. The coefficients that these functions return are at least 2 in absolute value and all of the same sign as the given rational number. The functions `neg_cont_frac_to_number` and `pos_cont_frac_to_number` execute the reverse process, i.e. compute a rational number given (the coefficients of) its negative or positive continued fraction expansion, respectively.

Examples.

- `number_to_neg_cont_frac(18,5) = [4,3,2]` and `neg_cont_frac_to_number(4,3,2) = 18/5`, because
$$\frac{18}{5} = 4 - \cfrac{1}{3 - \cfrac{1}{2}};$$
- `number_to_pos_cont_frac(30,7) = [4,3,2]` and `pos_cont_frac_to_number(4,3,2) = 30/7`, because
$$\frac{30}{7} = 4 + \cfrac{1}{3 + \cfrac{1}{2}};$$
- `number_to_neg_cont_frac(-18,5)= number_to_neg_cont_frac(18,-5) = [-4,-3,-2]`, because
$$-\frac{18}{5} = -4 - \cfrac{1}{-3 - \cfrac{1}{-2}}.$$

Instructions for `HF_SFS`. This program computes HF^+ of Seifert fibered spaces (SFS) that are rational homology spheres. It is largely based on a corresponding code written for MAGMA by Karakurt [Kar], which in turn is an implementation of Némethi's algorithm to compute HF^+ [Ném05]. The main function is `spinc_to_HF`, which outputs $\mathrm{HF}^+(Y)$ of a SFS Y, encoded as a dictionary of the format

`{spinc-structure :` $\mathbb{Z}[U]$`-module-structure of` HF^+`}`

(see example below on how to interpret this dictionary). The function `spinc_to_HF` takes as input a list of integers which should be of the format `[e, a_1, b_1, ..., a_n, b_n]`, where e and the a_i and the b_i are integers specifying Y as in Figure 1 and such that the corresponding plumbing (obtained by expanding the a_i/b_i into continued fractions) is definite.

To compute HF^+ of a lens space $L(p,q)$, a prism manifold $P(p,q)$ or a Brieskorn sphere $\Sigma(a_1,\dots,a_n)$, the input list can be computed using the functions `lens(p,q)`, `prism(p,q)` and `brieskorn(a_1, ..., a_n)`, respectively (see below for orientation conventions). Calling the function `print_HF` will return $\mathrm{HF}^+(Y)$ in a more legible manner. Finally, building on `spinc_to_HF`,

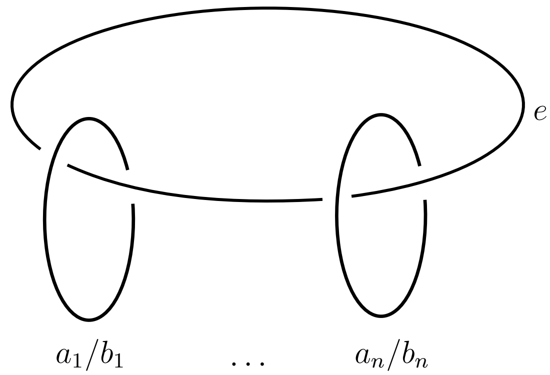

FIGURE 1

the function `correction_terms` returns the list of correction terms of Y, and `is_lspace` checks whether Y is an L-space.

Remark. The input list for $-\Sigma(a_1, \dots, a_n)$ is computed using `brieskorn(-a_1, ..., -a_n)`. The functions `lens(p,q)` and `prism(p,q)` take any pair of non-zero integers as arguments, so that e.g. `lens(8,5)`, `lens(-8,-5)`, `lens(8,13)` and `lens(8,-3)` each return the input list corresponding to the lens space $L(8,5)$.

Example. The Brieskorn sphere $\Sigma(2,3,7)$ bounds a (negative) definite plumbing along the star-shaped tree on four vertices with three branches, where the central vertex has weight -1 and the leaves have weights -2, -3 and -7. The corresponding input list is `[-1,-2,1,-3,1,-7,1]`. Given this input, `spinc_to_HF` returns the dictionary `{(0, 0, 0, 0) :  {0:[0], 1:[-1]}}`. This is to be interpreted as saying that in the (unique) Spin^c-structure labelled `(0,0,0,0)`, $\mathrm{HF}^+(\Sigma(2,3,7))$ consists of an infinite tower whose bottommost element is in grading 0, and a tower of length 1 (i.e. a $\mathbb{Z}$-summand) whose bottommost element is in grading -1. That is,

$$\mathrm{HF}^+(\Sigma(2,3,7)) = \mathcal{T}^+_{(0)} \oplus \mathbb{Z}_{(-1)}.$$

The same output can be obtained by calling `spinc_to_HF(brieskorn(2,3,7))`, and also be displayed in the above, more legible manner by calling `print_HF([-1,-2,1,-3,1,-7,1])`.

The above output shows that the (single) correction term of Y equals 0, which could have been computed by calling e.g. `correction_terms(brieskorn(2,3,7))`.

The above output also shows that $\Sigma(2,3,7)$ is not an L-space; this could have been verified by calling either `is_lspace(brieskorn(2,3,7))` or `is_lspace([-1,-2,1,-3,1,-7,1])`, which returns `False`.

Instructions for `CW`. This program computes the Casson-Walker invariant of Seifert fibered spaces that are rational homology spheres. Generally, $\lambda(Y)$ is computed by calling the function `casson_walker`, whose input should be a list of integers of the same format as is used for `spinc_to_HF` (see above). In the case where Y is a lens space or a prism manifold, the functions `casson_walker_lens` and `casson_walker_prism` can be called instead.

Remark. While the Casson-Walker invariant can be extracted from $\mathrm{HF}^+(Y)$ (see "Conventions" below), closed formulas for $\lambda(Y)$ are used in `CW` in order to increase speed. These underlying formulas can be found in [NN02, Section 5.3] (for the general case and the lens space case) and in [BNOV, Section 2.1] (for the prism manifold case).

Example. $\lambda(\Sigma(2,3,7)) = -1$; this can be obtained by calling `casson_walker(brieskorn(2,3,7))` or `casson_walker([-1,-2,1,-3,1,-7,1])`.

Instructions for `RIB_OBS`. This program can be used to check whether a pair of SFS's Y_1 and Y_2 passes the following obstruction to there being a ribbon rational homology cobordism from Y_1 to Y_2. If Y_1 admits a ribbon rational homology cobordism to Y_2, then

- $|H_1(Y_2;\mathbb{Z})| = u^2 \cdot |H_1(Y_1;\mathbb{Z})|$, for some $u \in \mathbb{Z}$; and
- each correction term of Y_1 appears u times among the correction terms of Y_2 (counted with multiplicity)

The function `ribbon_obstruction_corr` checks whether a pair of rational homology spheres passes this obstruction if the list of their correction terms is known, and takes these two lists as its input.

The function `ribbon_obstruction_sfs` checks whether a pair of SFS's passes this obstruction given definite plumbings filling the two SFS's (where each plumbing is entered as a list of integers as in `spinc_to_HF`). Again, there are shortcut functions `ribbon_obstruction_lens` and `ribbon_obstruction_prism` for the case where both SFS's are lens spaces or prism manifolds, respectively.

Example. Let $Y_1 = L(2,1)$ and $Y_2 = L(8,5)$. The lists of correction terms of Y_1 and Y_2 are [1/4, −1/4] and [1/4, −1/4, −3/8, 5/8, −3/8, −1/4, 5/8, 1/4], respectively. Since

$$|H_1(Y_2;\mathbb{Z})| = 8 = 2^2 \cdot 2 = 2^2 \cdot |H_1(Y_1;\mathbb{Z})|,$$

and because each correction term of Y_1 appears 2 times among those of Y_2, the pair passes the obstruction. This can be verified by calling `ribbon_obstruction_corr([1/4, -1/4],[1/4, -1/4, -3/8, 5/8, -3/8, -1/4, 5/8, 1/4])`, which returns `True`.

Since Y_1 and Y_2 bound plumbings whose input lists are `[-2]` and `[-3,-2,1,-2,1]`, respectively, the same output can be obtained by calling `ribbon_obstruction_sfs([-2],[-3,-2,1,-2,1])`.

Alternatively, since both Y_1 and Y_2 are lens spaces, this output can be obtained by calling `ribbon_obstruction_lens(2,1,8,5)`.

Conventions.

- The lens space $L(p,q)$ is the oriented 3-manifold obtained by performing $-(p/q)$-framed Dehn surgery along the unknot in S^3.
- The prism manifold $P(p,q)$ is the 3-manifold obtained by setting $n = 3$, $a_1/b_1 = a_2/b_2 = -2/1$ and $a_3/b_3 = -p/q$ in Figure 1.
- Brieskorn spheres are oriented such that $\Sigma(2,3,5)$ is the boundary of the negative definite plumbing along the E_8 Dynkin diagram (where each vertex has weight -2).

- The Casson-Walker invariant is defined as in [Rus05], i.e.

$$\lambda(Y) = \frac{1}{|H_1(Y;\mathbb{Z})|} \sum_{\mathfrak{s}\in \mathrm{Spin}^c(Y)} \left(\chi(\mathrm{HF}^+_{\mathrm{red}}(Y)) - \frac{d(Y,\mathfrak{s})}{2} \right).$$

References

[BNOV] William Ballinger, Yi Ni, Tynan Ochse, and Faramarz Vafaee. The prism manifold realization problem iii. (preprint: arXiv:1808.05321 [math.GT]).

[Kar] Çağri Karakurt. MAGMA code. http://web0.boun.edu.tr/cagri.karakurt/Research.html.

[Ném05] András Némethi. On the Ozsváth-Szabó invariant of negative definite plumbed 3-manifolds. *Geom. Topol.*, 9:991–1042, 2005.

[NN02] András Némethi and Liviu I. Nicolaescu. Seiberg-Witten invariants and surface singularities. *Geom. Topol.*, 6:269–328, 2002.

[Rus05] Raif Rustamov. *On Heegaard Floer homology of three-manifolds*. ProQuest LLC, Ann Arbor, MI, 2005. Thesis (Ph.D.)–Princeton University.