More ties than we thought

DAN HIRSCH
Upstanding Hackers LLC
thequux@upstandinghackers.com

MEREDITH L. PATTERSON
Upstanding Hackers LLC
mlp@upstandinghackers.com

ANDERS SANDBERG
Oxford University
anders.sandberg@philosophy.ox.ac.uk

MIKAEL VEJDEMO-JOHANSSON
KTH Royal Institute of Technology, Stockholm
Jožef Štefan Institute, Ljubljana
Institute for Mathematics and its Applications, Minneapolis
mvj@kth.se

Abstract

We extend the existing enumeration of neck tie knots to include tie knots with a textured front, tied with the narrow end of a tie. These tie knots have gained popularity in recent years, based on reconstructions of a costume detail from The Matrix Reloaded, and are explicitly ruled out in the enumeration by Fink and Mao (2000).

We show that the relaxed tie knot description language that comprehensively describes these extended tie knot classes is either context sensitive or context free. It has a sub-language that covers all the knots that inspired the work, and that is regular. From this regular sub-language we enumerate 177,147 distinct tie knots that seem tieable with a normal necktie. These are found through an enumeration of 2,046 winding patterns that can be varied by tucking the tie under itself at various points along the winding.
I. INTRODUCTION

There are several different ways to tie a necktie. Classically, knots such as the four-in-hand, the half windsor and the full windsor have been commonly taught to new tie-wearers. In a sequence of papers and a book, Fink and Mao [1–3] defined a formal language for describing tie knots, encoding the topology and geometry of the knot tying process into the formal language, and then used this language to enumerate all tie knots that could reasonably be tied with a normal-sized necktie.

The enumeration of Fink and Mao crucially depends on dictating a particular finishing sequence for tie knots: a finishing sequence that forces the front of the knot – the façade – to be a flat stretch of fabric. With this assumption in place, Fink and Mao produce a list of 85 distinct tie knots, and determine several novel knots that extend the previously commonly known list of tie knots.

In recent years, however, interest has been growing for a new approach to tie knots. In The Matrix Reloaded [12], the character of “The Merovingian” has a sequence of particularly fancy tie knots. Attempts by fans of the movie to recreate the tie knots from the Merovingian have led to a collection of new tie knot inventions, all of which rely on tying the tie with the thin end of the tie – the thin blade. Doing this allows for a knot with textures or stylings of the front of the knot, producing symmetric and pleasing patterns.

Knorr [4] tells the story of the main participants and their additions to the conversation:

On 21 June 2003 Luke edeity Housego invents the inverse tie-knots. The day before he had seen ‘Matrix Reloaded’ at the cinema and wanted to have a tie-knot as cool as the one the character ‘Merovingian’ sports in the movie.

On 28 September 2003 Luke publishes a .pdf-tutorial for his knot on the Internet. He calls his invention ‘edeity’s knot.’

On 03 February 2006 Victor Allen Lord Whimsy Crawford III publishes a .pdf-tutorial for a tie-knot he calls ‘The Merovingian.’ In fact it is edeity’s sequence, but rendered much more clearly than in edeity’s original .pdf. Whimsy had the idea from said .pdf, but was not sure, if he had matched the sequence.

On 16 February 2007 Henry SimplyJustHen Hu publishes a video on YouTube wherein he shows how to tie a knot he calls the ‘Hen Tie.’ In the video Henry makes clear that he has the idea from edeity’s .pdf-tutorial, but that he was not sure if he had matched the sequence. In fact Henry’s sequence slightly differs from edeity’s.

On 18 February 2007 the knot called ‘Merovingian’ appears in the German version of the Wikipedia, linking to Lord Whimsy’s tutorial.

On 04 May 2008 Jeffrey cwtrain Eldredge publishes a video on YouTube, demonstrating how to tie an even larger inverse tie-knot he calls the ‘Eldredge.’ Luke edeity Housego gave the world the inverse tie-knots, and Jeffrey Eldredge invented a subterfuge in tie-knotting not to be found in the literature so far: He simply tucks away the rest of the tie’s narrow end under the collar, thereby making possible the largest tie-knot known. This move rightfully can be called ‘the Eldredge tuckaway.’ But there is a problem with Jeffrey’s knot: It’s not a knot, but more a ‘wrapping.’

On 19 October 2008 Alexander zephyrin_xirdal Knorr publishes the description and sequence of the ‘Eldredge Variant’ in his weblog, making the ‘Eldredge’ into a true knot.

On 19 June 2010 Jeffrey Eldredge publishes the video ‘The Eldredge Knot: Revisited’ on YouTube, demonstrating how to tie the sequence of the ‘Eldredge Variant.’

In this paper, we present a radical simplification of the formal language proposed by Fink and Mao, together with an analysis of the asymptotic complexity class of the tie knots language. We produce a novel enumeration of necktie knots tied with the thin blade, and compare it to the results of Fink and Mao.
Fig. 1. The parts of a necktie, and the division of the wearer’s torso with the regions (Left, Center Right) and the winding directions (Turnwise, Widdershins) marked out for reference.

A. Formal languages

The work in this paper relies heavily on the language of formal languages, as used in theoretical computer science and in mathematical linguistics. For a comprehensive reference, we recommend the textbook by Sipser [11].

Recall that given a finite set \( L \) called an alphabet, the set of all sequences of any length of items drawn (with replacement) from \( L \) is denoted by \( L^* \). A formal language on the alphabet \( L \) is some subset \( A \) of \( L^* \). Depending on how difficult it is to accurately determine membership in a formal language \( A \), it places in one of several complexity classes. Languages that are described by finite state automata are regular; languages that require a pushdown automaton are context free; languages that require a linear bounded automaton are context sensitive and languages that require a full Turing machine to determine are called recursively enumerable. This sequence builds an increasing hierarchy of expressibility and computational complexity for syntactic rules for strings of some arbitrary sort of tokens.

One way to describe a language is to give a grammar – a set of production rules that decompose some form of abstract tokens into sequences of abstract or concrete tokens, ending with a sequence of elements in some alphabet. The standard notation for such grammars is the Backus-Naur form, which uses ::= to denote the production rules and (some name) to denote the abstract tokens. Further common symbols are * – the Kleene star, that denotes an arbitrary number of repetitions of the previous token (or group in brackets), and | denoting a choice of one of the adjoining options.

II. THE ANATOMY OF A NECKTIE

In the following, we will be referring quite a lot to various parts and constructions with a necktie. We call the ends of a necktie blades, and distinguish between the broad blade and the thin blade – see Figure 1 for these names. The tie knot can be divided up into a body, consisting of all the twists and turns that are not directly visible in the final knot, and a façade, consisting of the parts of the tie actually visible in the end. In Figure 2 we demonstrate this distinction. The body builds up the overall shape of the tie knot, while the façade gives texture to the front of the knot. The enumeration of Fink and Mao only considers knots with trivial façades, while these later inventions all consider more interesting façades. As a knot is in place around a wearer, the Y-shape of the tie divides the torso into 3 regions: Left, Center and Right – as shown to the right in Figure 1.

A tie knot has to be tied by winding and tucking one of the two blades around the other: if both blades are active, then the tie can no longer be adjusted in place for a comfortable fit. We shall refer to the blade used in tying the knot as the leading blade or the active blade. Each time the active blade is moved across the tie knot – in front or in back – we call the part of the tie laid on top of the knot a bow.

1There are neckties without a width difference between the ends. We ignore this distinction for this paper.
III. A LANGUAGE FOR TIE KNOTS

Fink and Mao [2] observe that once the first crossing has been made, the wrapping sequence of a classical tie knot is completely decided by the sequence of regions into which the broad blade is moved. Adorning the region specifications with a direction — is the tie moving away from the wearer or towards the wearer — they establish a formal alphabet for describing tie knots with 7 symbols. We reproduce their construction here, using \( U \) for the move to tuck the blade under the tie itself\(^2\). The notation proposed by Fink and Mao [2] interprets repetitions \( U^k \) of \( U \) as tucking the blade \( k \) bows under the top. It turns out that the complexity analysis is far simpler if we instead write \( U^k \) for tucking the blade under the bow that was produced \( 2k \) windings ago. This produces a language on the alphabet:

\[
\{L_\odot, L_\kappa, C_\odot, C_\kappa, R_\odot, R_\kappa, U\}
\]

They then introduce relations and restrictions on these symbols:

- **\( T\text{ie}_1 \)**: No region \((L, C, R)\) shall repeat. \( U \) moves do not influence this.
- **\( T\text{ie}_2 \)**: No direction \((\kappa, \odot)\) shall repeat. \( U \) moves do not influence this.
- **\( T\text{ie}_3 \)**: Tucks \((U)\) are valid after an outward move.
- **\( T\text{ie}_4 \)**: A tie knot can end only on one of \( C_\odot, C_\kappa \) or \( U \). In fact, almost all classical knots end on \( U \).
- **\( T\text{ie}_5 \)**: A \( k \)-fold tuck \( U^k \) is only valid after at least \( 2k \) preceding moves. Fink and Mao [2] do not pay much attention to the conditions on \( k \)-fold tucks, since these show up in their enumeration as stylistic variations, exclusively at the end of a knot.

This collection of rules allow us to drastically shrink the tie language, both in alphabet and axioms. Fink and Mao are careful to annotate whether tie knot moves go outwards or inwards at any given point. We note that the inwards/outwards distinction follows as a direct consequence of axioms **\( T\text{ie}_2, T\text{ie}_3 \) and **\( T\text{ie}_4 \). Since non-tuck moves must alternate between inwards and outwards, and the last non-tuck move must be outwards, the orientation of any sequence of moves follows by backtracking from the end of the string.

Hence, when faced with a non-annotated string like

\[
RCLCRCLCRCLRURCLU
\]

we can immediately trace from the tail of the knot string: the last move before the final tuck must be outwards, so that \( L \) must be a \( L_\odot \). So it must be preceded by \( R_\odot C_\odot \). Tracing backwards, we can specify

\(^2\)Fink and Mao used \( T \) for Tuck

\(^3\)The exemption here being the Onassis style knot, favored by the eponymous shipping magnate, where after a classical knot the broad blade is brought up with a \( C_\odot \) move to fall in front of the knot, hiding the knot completely.
the entire string above to
\[ R ⊗ C ⊗ L ⊗ C ⊗ R ⊗ C ⊗ L ⊗ C ⊗ R ⊗ C ⊗ L ⊗ C ⊗ R ⊗ C \]

Next, the axiom \( \text{Tie}_1 \) means that a sequence will not contain either of \( LU^*L, CU^*C, RU^*R \) as subsequences.\(^4\) Hence, the listing of regions is less important than the direction of transition: any valid transition is going to go either clockwise or counterclockwise. Writing \( T \) for clockwise\(^5\) and \( W \) for counterclockwise,\(^6\) we can give a strongly reduced tie language on the alphabet \( T, W, U \). To completely determine a tie knot, the sequence needs a starting state: an annotation on whether the first crossing of a tie knot goes across to the right or to the left. In such a sequence, a \( U \) instruction must be followed by either \( T \) or \( W \) dictating which direction the winding continues after the tuck, unless it is the last move of the tie: in this case, the blade is assumed to continue straight ahead – down in front for most broad-blade tie knots, tucked in under the collar for most thin-blade knots.

Position of the leading blade after a sequence of \( W/T \) windings is a direct result of \( \#W - \#T \equiv 2 \pmod{3} \). This observation allows us to gain control over several conditions determining whether a distribution of \( U \) symbols over a sequence of \( W/T \) produces a physically viable tie knot.

**Theorem 1:** A position in a winding sequence is valid for a \( k \)-fold tuck if the position is preceded by \( 2k \) winding symbols (\( W \) or \( T \)) that either

1) starts with \( W \) and satisfies \( \#W - \#T = 2 \pmod{3} \)
2) starts with \( T \) and satisfies \( \#T - \#W = 2 \pmod{3} \)

**Proof:** The initial symbol produces the bow under which the tuck will go. If the initial symbol goes, say, from \( R \) to \( L \), then the tuck move needs to come from \( C \) in order to go under the bow. In general, a tuck needs to come from the one region not involved in the covering bow. Every other bow goes in front of the knot, and the others go behind the knot. Hence, there are \( 2k - 1 \) additional winding symbols until the active blade returns to the right side of the knot. During these \( 2k - 1 \) symbols, we need to transition one more step around the sequence of regions. The transitions \( W \) and \( T \) are generator and inverse for the cyclic group of order 3, concluding the proof.

We may notice that with the usual physical constraints on a tie – where we have experimentally established that broad blade ties tend to be bounded by 9 moves, and thin blade ties by 15 moves, we can expect that no meaningful tuck deeper than 7 will ever be relevant; 4 for the broad blade ties. The bound of 4 is achieved in the enumeration by Fink and Mao\(^3\).

### IV. LANGUAGE COMPLEXITY

In this section, we examine the complexity features of the tie knot language. Due to the constraints we have already observed on the cardinality of \( W \) and \( T \), we will define a grammar for this language using the attribute grammar formalism of Knuth\(^5\). We will write this grammar with an annotated Backus-Naur form – symbols may have a finite number of numeric or boolean attributes that can be used to validate a potentially correct string in the grammar. Although in practice it is only possible to realise finite strings in the tie knot language due to the physical properties of fabric, we assume an arbitrarily long (but finite), infinitely thin tie.

**A. Single-depth tucks**

The following regular grammar describes all valid winding sequences for knots in which tucks only pass the active blade under the most recent bow made over the knot (which we will call **depth-1-tuckable**).

\(^4\)Recall that the Kleene star \( F^* \) is used to denote sequences of 0 or more repetitions of the string \( F \).

\(^5\)\( T \) for Turnwise

\(^6\)\( W \) for Widdershins
This subclass can be described by a simple grammar; if we want to allow deeper tucks – going under earlier bows – then an attribute grammar as in Section IV-B is called for.

\[
\begin{align*}
  \langle \text{prefix} \rangle & ::= \text{T} | \text{W} | \epsilon \\
  \langle \text{tuck} \rangle & ::= \text{TTU} | \text{WWU} \\
  \langle \text{pair} \rangle & ::= \text{WT} | \text{WW} | \text{TT} | \text{TW} \\
  \langle \text{tie} \rangle & ::= \langle \text{prefix} \rangle \langle \text{pair} \rangle \langle \text{tuck} \rangle^* \langle \text{tuck} \rangle \\
\end{align*}
\]

As section V elaborates, the distribution of T and W varies by type of knot: for classical knots, \( \#W - \#T = 2 \pmod{3} \); for modern knots that tuck to the right, \( \#W - \#T = 1 \pmod{3} \); and for modern knots that tuck to the left, \( \#W - \#T = 0 \pmod{3} \). This grammar does not discriminate between these three subclasses.

B. Recursive tucks

We allow for greater tuck depth by making the \( \langle \text{tuck} \rangle \) rule recursive:

\[
\langle \text{tuck} \rangle ::= \langle \text{pair} \rangle \langle \text{tuck} \rangle' \cup' \epsilon
\]

We then add attributes to the \( \langle \text{pair} \rangle \) and \( \langle \text{tuck} \rangle \) rules as follows:

\[
\begin{align*}
  \langle \text{pair} \rangle & ::= \text{TT}' | \text{TW}' | \text{WT}' | \text{WW}' \\
  \text{pair}.t & = 2, \text{pair}.w = 0 \\
\langle \text{tuck} \rangle & ::= \langle \text{pair} \rangle \langle \text{tuckable} \rangle' \cup' \\
  \text{tuck}.t & = \text{pair}.t + \text{tuckable}.t \\
  \text{tuck}.w & = \text{pair}.w + \text{tuckable}.w \\
  \text{tuck}.valid & = \text{tuck}.t - \text{tuck}.w \pmod{3} \text{ if } \text{tuck}[0] = \text{T'} \\
\langle \text{tuckable} \rangle & ::= \epsilon \\
  \text{tuckable}.t & = 0, \text{tuckable}.w = 0 \\
\langle \text{tuck} \rangle & ::= \langle \text{tuck} \rangle' \text{tuck}\text{valid} = \text{True if all } \text{tuck}.valid = 1 \\
\end{align*}
\]

Note that the validity of a tuck depends only on the count of T and W in the entire sequence comprising the tuck, and not the validity of any tucks recursively embedded into it. For instance, TWTTUU is a valid depth-2-tuckable sequence, as is its embedded depth-1-tuckable sequence TTU. However, TTWTUU is also a valid depth-2-tuckable sequence, even though WTTU is not a valid depth-1-tuckable sequence.

Finally, we add one last attribute to the top-level \( \langle \text{tie} \rangle \) rule:

\[
\langle \text{tie} \rangle ::= \langle \text{prefix} \rangle \langle \text{pair} \rangle \langle \text{tuck} \rangle^* \langle \text{tuck} \rangle \quad \text{[tie.valid} = \text{True if all } \text{tuck}.valid = 1\text{]}
\]

Since all the semantic information passes up the parse tree, the grammar uses only synthesized attributes and is therefore an S-attributed grammar, amenable to both top-down and bottom-up parsing approaches.
C. Classification of the tie-knot language

If we limit our attention to only the single depth tie knots described in Section IV-A, then the grammar is regular, proving that this tie language is a regular language and can be described by a finite automaton. In fact, an automaton accepting these tie knots is given by:

![Automaton Diagram]

Execution starts at the middle node, but has to go outside and return before the machine will accept input.

As for the deeper tucked language in Section IV-B, classification is significantly harder: the presence of recursion means it is at least context-free, whereas the expressibility with an attribute grammar implies at most context-sensitive. The language does not seem immediately amenable to pumping lemma arguments, nor to treatment with Ogden’s lemma or Parikh’s theorem or interchange lemmas[8–10, 13].

V. Enumeration

We can cut down the enumeration work by using some apparent symmetries. Without loss of generality we can assume that a tie knot starts by putting the active blade in region R: any knot starting in the region L is the mirror image of a knot that starts in R and swaps all W to T and vice versa.

A. The classical case

Knots finishing in region C include all classical knots as classified by Fink and Mao [2]: these are all tied with the broad blade and finish with C. Hence, they are classified completely by W/T sequences with \( \#W - \#T = 1 \pmod{3} \), by not counting the initial crossing as one of the moves –

For a tie knot with \( k \) moves, the W and T moves need to divide so that \( \#W - \#T = 1 \pmod{3} \). This is possible in \( \lceil k/3 \rceil \) ways. For each allotment of W and T moves, then, a classical tie knot must end with at least a single order tuck. If it weren’t for this condition, we could count the number of ties for each allotment as \( (\#T)_k \), by choosing the positions that receive a T symbol. However, we need to disqualify the choices that place different symbols at the two last positions of the string.

These disqualified versions end with either WT or TW and are preceded by an arbitrary sequence of \( k-2 \) W and T symbols. They are only possible if there are at least 1 of each symbol available to begin with.

**Theorem 2:** The number of classical ties with exactly \( k \) winding steps is

\[
\sum_{t-w=k \atop t-w=1 \pmod{3}} \binom{k}{t} - 2\binom{k-2}{t-1}
\]

**Proof:** Out of the \( k \) winding steps, we need 1 more T than W to reach the C position for the finishing move. For any choice of \( \#T, \#W \), there are \( \binom{k}{\#T} \) of these. Out of all ways to place the requisite number of T in the winding sequence, we remove those that use up one each of T and W for the last two moves.
Such windings are counted by \( (k-2)_{#T-1} \), and for each such sequence, both TW and WT are forbidden finishing sequences.

Hence, for each candidate sequence of \( k-2 \) moves we need to exclude two sequences.

In particular, the cases enumerated by Fink and Mao \([2]\) are

\[
\begin{array}{ccccccccc}
\text{Winding length} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \text{total} \\
\# tie knots & 1 & 1 & 3 & 5 & 11 & 21 & 43 & 85 \\
\end{array}
\]

**B. The modern case**

A knot with the thick blade active will cover up the entire knot with each new bow. As such, all thick blade active tie knots will fall within the classification by Fink and Mao \([2]\).

The modern case, thus, deals with thin blade active knots. As evidenced by the Trinity and the Eldredge knots, thin blade knots have a wider range of interesting façades and of interesting tuck patterns. Where for thick blade knots, it was enough to assume that the tuck happens last, and from the C region, the thin blade knots have a far wider variety.

The case remains that unless the last move is a tuck – or possibly finishes in the C region – the knot will unravel from gravity. We can thus expect this to be a valid requirement for the enumeration. There are often more valid tuck sites than the final position in a knot, and the tuck need no longer come from the C region: R and L are at least as valid.

To enumerate windings that end with a single depth tuck move, the formula from Theorem 2 will serve our needs, with some slight modification. More interesting, however, is to enumerate all tie knots with all their choices of possible tuck sites.

Winding sequences can be generated with the same building blocks we used for the classical case above.

**Theorem 3:** The number of tie winding sequences ending with a single depth tuck from the left after \( k \) winding steps is

\[
\sum_{t+w=k \pmod{3}} \binom{k}{t} - 2 \binom{k-2}{t-1}
\]

The number of winding sequences ending with a single depth tuck from the right after \( k \) winding steps is

\[
\sum_{t+w=k \pmod{3}} \binom{k}{t} - 2 \binom{k-2}{t-1}
\]

**Proof:** The proof closely follows that of Theorem 2. In each case, we need to enforce repetition of the last winding symbol to allow for a single depth tuck at the end. In each case, we also need to end our winding sequence in the correct region for the tuck to take place. The region placement is handled by the \( \pmod{3} \) residue.

While a good closed form expression for the total number of tie knots, especially including all the tuck combinations of different depths, seems out of reach at the current time, the descriptions we have given so far provide accessible routes to algorithmically enumerating possible tie knots. In the appendix, we include a listing of single depth tuck knots, as well as python code that will list and print knots and potential tuck sites for any depth and winding length.

To extend the table above, we can use the formulae in Theorems 2 and 3 to establish the following:

\[
\begin{array}{ccccccccc}
\text{Winding length} & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & \text{total} \\
\# left single tuck windings & 0 & 2 & 2 & 6 & 10 & 22 & 42 & 86 & 170 & 324 & 682 \\
\# right single tuck windings & 1 & 1 & 3 & 5 & 11 & 21 & 43 & 85 & 171 & 341 & 682 \\
\# center single tuck windings & 1 & 1 & 3 & 5 & 11 & 21 & 43 & 85 & 171 & 341 & 682 \\
\end{array}
\]
The reason for the similarity between the right and the center counts is that if \( t - w = 1 \) (mod 3), then \( w - t = 2 \) (mod 3). Hence, a winding sequence for a center tuck can be mirrored to a winding sequence for a right tuck.

By counting the number of detected tuck sites, we have calculated the total number of tie knots using only single depth tucks to be 177,147. Of these, 59,016 each end with a right or a center tuck, and 59,115 end with a left tuck.

VI. AESTHETICS

Fink and Mao [2] propose several measures to quantify the aesthetic qualities of a necktie knot; notably symmetry and balance, corresponding to the quantities \(#R - #L\) and the number of transitions from a streak of W to a streak of T or vice versa.

By considering the popular thin-blade neck tie knots: the Eldredge and the Trinity, as described in [6, 7], we can immediately note that balance no longer seems to be as important for the look of a tie knot as is the shape of its façade. Symmetry still plays an important role in knots, and is easy to calculate using the CLR notation for tie knots.

| Knot  | TW-string       | CLR-string          | Balance | Symmetry |
|-------|----------------|---------------------|---------|----------|
| Eldredge | TTTWWTTUTTWU   | LCRLRCRLUCRCLU | 3       | 0        |
| Trinity | TWWWTTTTUTTU | LCLRCRLCURLU      | 2       | 1        |

We do not in this paper attempt to optimize any numeric measures of aesthetics, as this would require us to have a formal and quantifiable measure of the knot façades. This seems difficult with our currently available tools.

VII. CONCLUSION

In this paper, we have extended the enumeration methods originally used by Fink and Mao [2] to provide a larger enumeration of necktie knots, including those knots tied with the thin blade of a necktie to produce ornate patterns in the knot façade.

We have found 2,046 winding patterns that take up to 11 moves to tie and are anchored by a final single depth tuck, and thus are reasonable candidates for use with a normal necktie. We chose the number of moves by examining popular thin-blade tie knots – the Eldredge tie knot uses 11 moves. Most of these winding patterns allow several possible tuck patterns, and thus the 2,046 winding sequences generate 177,147 tie knots with single depth tucks.

We have further shown that in the limit, the language describing neck tie knots is either context sensitive or context free, with a regular sub-language describing the 177,147 knots above.

Questions that remain open to our mind include:
- Settle the language complexity class of the full tie knot language.
- Find a way to algorithmically divide a knot description string into a body/façade distinction.
- Using such a distinction, classify all possible knot façades with reasonably short necktie lengths.
REFERENCES

[1] T. Fink and Y. Mao. *The 85 ways to tie a tie*. Fourth Estate, 2001.

[2] T. Fink and Y. Mao. “Tie knots, random walks and topology”. In: *Physica A: Statistical Mechanics and its Applications* 276.1 (2000), pp. 109–121.

[3] T.M. Fink and Y. Mao. “Designing tie knots by random walks”. In: *Nature* 398.6722 (1999), pp. 31–32.

[4] A. Knorr. *eldredge reloaded*. Blog post on [http://xirdalium.net](http://xirdalium.net). Accessed on 26 December 2012. June 2010. URL: [http://xirdalium.net/2010/06/20/eldredge-reloaded/](http://xirdalium.net/2010/06/20/eldredge-reloaded/).

[5] D. Knuth. “Semantics of context-free languages”. In: *Mathematical Systems Theory* 2.2 (1967), pp. 127–145.

[6] A. Krasny. *Eldredge Tie Knot - How to Tie a Eldredge Necktie Knot*. Blog post on [http://agreeordie.com](http://agreeordie.com). Accessed on 26 December 2012. Nov. 2012. URL: [http://agreeordie.com/blog/musings/545-how-to-tie-a-necktie-eldredge-knot](http://agreeordie.com/blog/musings/545-how-to-tie-a-necktie-eldredge-knot).

[7] A. Krasny. *Trinity Tie Knot - How to Tie a Trinity Necktie Knot*. Blog post on [http://agreeordie.com](http://agreeordie.com). Accessed on 26 December 2012. Dec. 2012. URL: [http://agreeordie.com/blog/musings/553-how-to-tie-a-necktie-trinity-knot](http://agreeordie.com/blog/musings/553-how-to-tie-a-necktie-trinity-knot).

[8] William Ogden. “A helpful result for proving inherent ambiguity”. In: *Theory of Computing Systems* 2.3 (1968), pp. 191–194.

[9] William Ogden, Rockford J Ross, and Karl Winklmann. “An “interchange lemma” for context-free languages”. In: *SIAM Journal on Computing* 14.2 (1985), pp. 410–415.

[10] Rohit J Parikh. “On context-free languages”. In: *Journal of the ACM (JACM)* 13.4 (1966), pp. 570–581.

[11] Michael Sipser. *Introduction to the Theory of Computation*. Vol. 2. Thomson Course Technology Boston, 2006.

[12] A. Wachowski et al. *The matrix reloaded*. Warner Bros. Pictures, 2003.

[13] David S Wise. “A strong pumping lemma for context-free languages”. In: *Theoretical Computer Science* 3.3 (1976), pp. 359–369.
**Classical ties in WTU notation**

The 85 ties enumerated by Fink and Mao [2] are given by the following winding strings in our WTU notation for tie knots:

| Index | Tie string | Index | Tie string |
|-------|------------|-------|------------|
| 1     | WWU        | 43    | WWTWTTWWU  |
| 2     | WTTU       | 44    | WTWNWWTTU  |
| 3     | WTWWU      | 45    | WTTTTWTTU  |
| 4     | TTTTU      | 46    | WTTTTTTU   |
| 5     | TWWU       | 47    | WWWNTWTTU  |
| 6     | WTWWWU     | 48    | WTWNTWTTU  |
| 7     | WWWWWU     | 49    | WTTWWTTU   |
| 8     | WWTTTU     | 50    | WWTNTWNU   |
| 9     | TTTWU      | 51    | WTTTTWWU   |
| 10    | TWTTU      | 52    | TTNTWTTU   |
| 11    | WTWTWWU    | 53    | TWNTWTTU   |
| 12    | WTTTTU     | 54    | WWWWWWWU   |
| 13    | WWWTTU     | 55    | WTTTTTTU   |
| 14    | WTWWWWU    | 56    | WWWNTWTTU  |
| 15    | WTTWWU     | 57    | TTTWWWWU   |
| 16    | TTTWTTU    | 58    | TTTTTTTU   |
| 17    | TWTWWU     | 59    | WTTTTWWU   |
| 18    | TTWWWU     | 60    | TTTWTWTTU  |
| 19    | TTWWTU     | 61    | TTTWWWWU   |
| 20    | TWTTTU     | 62    | WWTNTWNU   |
| 21    | TWTWWU     | 63    | WTTTTTTU   |
| 22    | WTWTWWWU   | 64    | WTTWWTTU   |
| 23    | WTTTTWU    | 65    | TWNTWTTU   |
| 24    | WTWWWWU    | 66    | TWWWWTWWU  |
| 25    | WWWWTWU    | 67    | TTWTWTTU   |
| 26    | WTWWWTTU   | 68    | TWWNTTWTU  |
| 27    | WTTWWTTU   | 69    | TWWNTWTTU  |
| 28    | TTTWTWU    | 70    | TTTWTWTTU  |
| 29    | WTTWTTU    | 71    | TWWNTWWU   |
| 30    | TWTTWWWU   | 72    | TTTWTWTTU  |
| 31    | TTTTTTU    | 73    | WWNTWTTU   |
| 32    | TWWWWWU    | 74    | WWTWTWWU   |
| 33    | TTTWWWWU   | 75    | WWTWTWTTU  |
| 34    | WWWTTTTU   | 76    | TTWTWTUU   |
| 35    | TWWTTTU    | 77    | TWTWTWWU   |
| 36    | WWWTTWWU   | 78    | TWTWTWTTU  |
| 37    | WWWTTWWWU  | 79    | TWTWTWTTU  |
| 38    | TWTTWWWU   | 80    | TTWTWTWWU  |
| 39    | WTWTWWWU   | 81    | TTWTWTWWU  |
| 40    | TWTWTTU    | 82    | TTWTWTU    |
| 41    | TTWTWWWU   | 83    | TTWTWTU    |
| 42    | TWTWWTTU   | 84    | TTWTWTU    |
| 85    | TWTWTWWU   | 85    | TTWTWTU    |
Thin blade active tie knots

The knots that started this project were the Eldredge and the Trinity. Our transcriptions of these ties as demonstrated by Krasny [6, 7] are:

| Knot     | Knot Transcription   |
|----------|----------------------|
| Eldredge | TTTWWTUUTTWWU         |
| Trinity  | TWWWTUUTTU           |

Note that these show up in the enumerations below as L-373 (uuUu) and L-110 (uU).

In the next several pages of this appendix, we shall list enumerations for up to 12 winding moves – including the Eldredge and the Trinity in our enumeration. Throughout, we write u for a potential tuck site, where a single depth tuck is allowed, and U for the final tuck that keeps the tie knot in place. We split the enumeration based on where the final tuck takes place: from L, from R or from C.
We write the TWU strings with a lowercase u for any additionally feasible front tuck site. First ties knots that tuck from the left.

| Index | Tie string |
|-------|------------|
| L-1   | TTTU       |
| L-2   | WWWW       |
| L-3   | TTuWWU     |
| L-4   | WWuTTU     |
| L-5   | TWTTTU     |
| L-6   | TTWTUU     |
| L-7   | WTWT TU     |
| L-8   | TWWuWWU     |
| L-9   | WWTWU       |
| L-10  | WTWWU       |
| L-11  | TTuTTuTU   |
| L-12  | TTuWTWU     |
| L-13  | TTuTWUwWU   |
| L-14  | WTTuWWU     |
| L-15  | WTTWTU     |
| L-16  | TWTTuWWU    |
| L-17  | TWWuTTU     |
| L-18  | WUuWTUU     |
| L-19  | WUuWTTU     |
| L-20  | WUuWTWU     |
| L-21  | TTuTTuWWU   |
| L-22  | TTuTTuWUwWU |
| L-23  | TTuTTuWTU   |
| L-24  | TTuTTuWUWU  |
| L-25  | TTuTTuWTTU  |
| L-26  | TTuTTuWTWU  |
| L-27  | TWWuTTuTU   |
| L-28  | WTTuWTUU     |
| L-29  | WTTuWTTU    |
| L-30  | WTTuWTWU    |
| L-31  | WTTuWTWU    |
| L-32  | WTTuWTWU    |
| L-33  | TWWuWTWU     |
| L-34  | TWWuWTWU    |
| L-35  | TWWuWTWU    |
| L-36  | TWWuWTWU    |
| L-37  | WTTuWTWU    |
| L-38  | WTTuWTWU    |
| L-39  | WTTuWTWU    |
| L-40  | WTTuWTWU    |
| L-41  | WTTuWTWU    |
| L-42  | WTTuWTWU    |
| L-43  | WTTuWTWU    |
| L-44  | WTTuWTWU    |
| L-45  | WTTuWTWU    |
| L-46  | WTTuWTWU    |
| L-47  | WTTuWTWU    |
| L-48  | WTTuWTWU    |
| L-49  | WTTuWTWU    |
| L-50  | WTTuWTWU    |

Index
Tie string
L-41 | WTTuWTWU |
L-42 | WTTuWTWU |
L-43 | WTTuWTWU |
L-44 | WTTuWTWU |
L-45 | WTTuWTWU |
L-46 | WTTuWTWU |
L-47 | WTTuWTWU |
L-48 | WTTuWTWU |
L-49 | WTTuWTWU |
L-50 | WTTuWTWU |
L-51 | WTTuWTWU |
L-52 | WTTuWTWU |
L-53 | WTTuWTWU |
L-54 | WTTuWTWU |
L-55 | WTTuWTWU |
L-56 | WTTuWTWU |
L-57 | WTTuWTWU |
L-58 | WTTuWTWU |
L-59 | WTTuWTWU |
L-60 | WTTuWTWU |
L-61 | WTTuWTWU |
L-62 | WTTuWTWU |
L-63 | WTTuWTWU |
L-64 | WTTuWTWU |
L-65 | WTTuWTWU |
L-66 | WTTuWTWU |
L-67 | WTTuWTWU |
L-68 | WTTuWTWU |
L-69 | WTTuWTWU |
L-70 | WTTuWTWU |
L-71 | WTTuWTWU |
L-72 | WTTuWTWU |
L-73 | WTTuWTWU |
L-74 | WTTuWTWU |
L-75 | WTTuWTWU |
L-76 | WTTuWTWU |
L-77 | WTTuWTWU |
L-78 | WTTuWTWU |
L-79 | WTTuWTWU |
L-80 | WTTuWTWU |
L-81 | WTTuWTWU |
L-82 | WTTuWTWU |
L-83 | WTTuWTWU |
L-84 | WTTuWTWU |
L-85 | TTTuTTuTTuTTU |
L-86 | TTTuTTuTTuTTU |
L-87 | TTTuTTuTTuTTU |
L-88 | TTTuTTuTTuTTU |
L-89 | TTTuTTuTTuTTU |
L-90 | TTTuTTuTTuTTU |
L-91 | TTTuTTuTTuTTU |
L-92 | TTTuTTuTTuTTU |
L-93 | TTTuTTuTTuTTU |
L-94 | TTTuTTuTTuTTU |
L-95 | TTTuTTuTTuTTU |
L-96 | TTTuTTuTTuTTU |
L-97 | TTTuTTuTTuTTU |
L-98 | TTTuTTuTTuTTU |
L-99 | TTTuTTuTTuTTU |
L-100 | TTTuTTuTTuTTU |
L-101 | TTTuTTuTTuTTU |
L-102 | TTTuTTuTTuTTU |
L-103 | TTTuTTuTTuTTU |
L-104 | TTTuTTuTTuTTU |
L-105 | TTTuTTuTTuTTU |
L-106 | TTTuTTuTTuTTU |
L-107 | TTTuTTuTTuTTU |
L-108 | TTTuTTuTTuTTU |
L-109 | TTTuTTuTTuTTU |
L-110 | TTTuTTuTTuTTU |
L-111 | TTTuTTuTTuTTU |
L-112 | TTTuTTuTTuTTU |
L-113 | TTTuTTuTTuTTU |
L-114 | TTTuTTuTTuTTU |
L-115 | TTTuTTuTTuTTU |
L-116 | TTTuTTuTTuTTU |
L-117 | TTTuTTuTTuTTU |
L-118 | TTTuTTuTTuTTU |
L-119 | TTTuTTuTTuTTU |
L-120 | TTTuTTuTTuTTU |
| Index | Tie string | Index | Tie string | Index | Tie string |
|-------|------------|-------|------------|-------|------------|
| L-121 | WWTWWTTuTU | L-171 | TTuTTuTTuTTuWWu | L-221 | TTuWWuWTWWuTTu |
| L-122 | WWTWWTTuTU | L-172 | TTuTTuTTuWWuTTu | L-222 | TTuWWuWTWWuTTu |
| L-123 | WWTWWTTuTU | L-173 | TTuTTuWTWWuTTu | L-223 | TTuWWuWTWWuTTu |
| L-124 | WWTWWTTuTU | L-174 | TTuTTuWTWWuTTu | L-224 | TTuWWuWTWWuTTu |
| L-125 | WWTWWTTuTU | L-175 | TTuTTuTWTTuTTu | L-225 | TTuWWuWTWWuTTu |
| L-126 | WWTWWTTuTU | L-176 | TTuTTuTWTTuTTu | L-226 | TTuWWuWTWWuTTu |
| L-127 | WWTWWTTuTU | L-177 | TTuTTuWTWWuTTu | L-227 | TTuWTWWuWTWWu |
| L-128 | TTuWuWuWWuWWu | L-178 | TTuWTWWuWTWWu | L-228 | TTuWTWWuWTWWu |
| L-129 | TWWTWuWuWWu | L-179 | TTuWTWWuWTWWu | L-229 | TTuWTWWuWTWWu |
| L-130 | TWWTWuWuWWu | L-180 | TTuWTWWuWTWWu | L-230 | TTuWTWWuWTWWu |
| L-131 | TWWTWuWuWWu | L-181 | TTuWTWWuWTWWu | L-231 | TTuWTWWuWTWWu |
| L-132 | TWWTWuWuWWu | L-182 | TTuWTWWuWTWWu | L-232 | TTuWTWWuWTWWu |
| L-133 | TWWTWuWuWWu | L-183 | TTuWTWWuWTWWu | L-233 | TTuWTWWuWTWWu |
| L-134 | TWWTWuWuWWu | L-184 | TTuWTWWuWTWWu | L-234 | TTuWTWWuWTWWu |
| L-135 | TWWTWuWuWWu | L-185 | TTuWTWWuWTWWu | L-235 | TTuWTWWuWTWWu |
| L-136 | TWWTWuWuWWu | L-186 | TTuWTWWuWTWWu | L-236 | TTuWTWWuWTWWu |
| L-137 | TWWTWuWuWWu | L-187 | TTuWTWWuWTWWu | L-237 | TTuWTWWuWTWWu |
| L-138 | TWWTWuWuWWu | L-188 | TTuWTWWuWTWWu | L-238 | TTuWTWWuWTWWu |
| L-139 | TWWTWuWuWWu | L-189 | TTuWTWWuWTWWu | L-239 | TTuWTWWuWTWWu |
| L-140 | TWWTWuWuWWu | L-190 | TTuWTWWuWTWWu | L-240 | TTuWTWWuWTWWu |
| L-141 | TWWTWuWuWWu | L-191 | TTuWTWWuWTWWu | L-241 | TTuWTWWuWTWWu |
| L-142 | TWWTWuWuWWu | L-192 | TTuWTWWuWTWWu | L-242 | TTuWTWWuWTWWu |
| L-143 | TWWTWuWuWWu | L-193 | TTuWTWWuWTWWu | L-243 | TTuWTWWuWTWWu |
| L-144 | TWWTWuWuWWu | L-194 | TTuWTWWuWTWWu | L-244 | TTuWTWWuWTWWu |
| L-145 | TWWTWuWuWWu | L-195 | TTuWTWWuWTWWu | L-245 | TTuWTWWuWTWWu |
| L-146 | TWWTWuWuWWu | L-196 | TTuWTWWuWTWWu | L-246 | TTuWTWWuWTWWu |
| L-147 | TWWTWuWuWWu | L-197 | TTuWTWWuWTWWu | L-247 | TTuWTWWuWTWWu |
| L-148 | TWWTWuWuWWu | L-198 | TTuWTWWuWTWWu | L-248 | TTuWTWWuWTWWu |
| L-149 | TWWTWuWuWWu | L-199 | TTuWTWWuWTWWu | L-249 | TTuWTWWuWTWWu |
| L-150 | TWWTWuWuWWu | L-200 | TTuWTWWuWTWWu | L-250 | TTuWTWWuWTWWu |
| L-151 | TWWTWuWuWWu | L-201 | TTuWTWWuWTWWu | L-251 | TTuWTWWuWTWWu |
| L-152 | TWWTWuWuWWu | L-202 | TTuWTWWuWTWWu | L-252 | TTuWTWWuWTWWu |
| L-153 | TWWTWuWuWWu | L-203 | TTuWTWWuWTWWu | L-253 | TTuWTWWuWTWWu |
| L-154 | TWWTWuWuWWu | L-204 | TTuWTWWuWTWWu | L-254 | TTuWTWWuWTWWu |
| L-155 | TWWTWuWuWWu | L-205 | TTuWTWWuWTWWu | L-255 | TTuWTWWuWTWWu |
| L-156 | TWWTWuWuWWu | L-206 | TTuWTWWuWTWWu | L-256 | TTuWTWWuWTWWu |
| L-157 | TWWTWuWuWWu | L-207 | TTuWTWWuWTWWu | L-257 | TTuWTWWuWTWWu |
| L-158 | TWWTWuWuWWu | L-208 | TTuWTWWuWTWWu | L-258 | TTuWTWWuWTWWu |
| L-159 | TWWTWuWuWWu | L-209 | TTuWTWWuWTWWu | L-259 | TTuWTWWuWTWWu |
| L-160 | TWWTWuWuWWu | L-210 | TTuWTWWuWTWWu | L-260 | TTuWTWWuWTWWu |
| L-161 | TWWTWuWuWWu | L-211 | TTuWTWWuWTWWu | L-261 | TTuWTWWuWTWWu |
| L-162 | TWWTWuWuWWu | L-212 | TTuWTWWuWTWWu | L-262 | TTuWTWWuWTWWu |
| L-163 | TWWTWuWuWWu | L-213 | TTuWTWWuWTWWu | L-263 | TTuWTWWuWTWWu |
| L-164 | TWWTWuWuWWu | L-214 | TTuWTWWuWTWWu | L-264 | TTuWTWWuWTWWu |
| L-165 | TWWTWuWuWWu | L-215 | TTuWTWWuWTWWu | L-265 | TTuWTWWuWTWWu |
| L-166 | TWWTWuWuWWu | L-216 | TTuWTWWuWTWWu | L-266 | TTuWTWWuWTWWu |
| L-167 | TWWTWuWuWWu | L-217 | TTuWTWWuWTWWu | L-267 | TTuWTWWuWTWWu |
| L-168 | TWWTWuWuWWu | L-218 | TTuWTWWuWTWWu | L-268 | TTuWTWWuWTWWu |
| L-169 | TWWTWuWuWWu | L-219 | TTuWTWWuWTWWu | L-269 | TTuWTWWuWTWWu |
| L-170 | TWWTWuWuWWu | L-220 | TTuWTWWuWTWWu | L-270 | TTuWTWWuWTWWu |
| Index | Tie string               | Index | Tie string               |
|-------|--------------------------|-------|--------------------------|
| L-271 | TWTWTTuWTWWU             | L-321 | TWWWuWTWWuWWU            |
| L-272 | TWTWTTuWTWWU             | L-322 | TWWWuWTWWuWWU            |
| L-273 | TWTWTTTTuWTWWU           | L-323 | TWWWuWTWWuWTWWU          |
| L-274 | TWTWWTWWuTTU             | L-324 | TWWWuWuWTWWu             |
| L-275 | TWTWTTTTuWTWWU           | L-325 | WWWWuWTWWuWWuWWW         |
| L-276 | TWTWTTWuTTuTTU           | L-326 | WWWWuWTWWuWWW            |
| L-277 | TWTWuWuWTTTTU            | L-327 | WWWWuWTWWuWW             |
| L-278 | TWTWuWWuWTTTTU           | L-328 | WWWWuWTWWuW              |
| L-279 | TWWuWTuTTuTTuWWU         | L-329 | WWWWuWTWWuWWW            |
| L-280 | TWWuWTuWTuWTuTTU         | L-330 | WWWWuWTWWuWW             |
| L-281 | TWWuWTuWTUWTTU           | L-331 | WWWWuWTWWuW              |
| L-282 | TWWuWTuWTUWTTU           | L-332 | WWWWuWTWWu               |
| L-283 | TWWuWTuWTTTTU            | L-333 | WWWWuWTWWu               |
| L-284 | TWWuWTuWTTTTU            | L-334 | WPPPuWTWWuWWW             |
| L-285 | TWWuWTuWdTuTTuTTU        | L-335 | WWWWuWTWWuWWW             |
| L-286 | WWWWuWTuTuTTuWTWu        | L-336 | WWWWuWTWuWTT             |
| L-287 | WWWWuWTuTTuTTuWu         | L-337 | WWWWuWTWuWW              |
| L-288 | WWWWuWTuTTuTTuWu         | L-338 | WWWWuWTWuWW              |
| L-289 | WWWWuWTuTTuWTWu         | L-339 | WWWWuWTWuWu               |
| L-290 | WWWWuWTuWTuW            | L-340 | WWWWuWTWuWu               |
| L-291 | WWWWuWTuWTuWu            | L-341 | WWWWuWTWuWu               |
| L-292 | WWWWuWTuWTuWu            | L-342 | WWWWuWTWuWu               |
| L-293 | WWWWuWTuWTuWu            | L-343 | WWWWuWTWuWu               |
| L-294 | WWWWuWTuWTuWu            | L-344 | WWWWuWTWuWu               |
| L-295 | WWWWuWTuWTuWu            | L-345 | WWWWuWTWuWu               |
| L-296 | WWWWuWTuWTuWu            | L-346 | WWWWuWTWuWu               |
| L-297 | WWWWuWTuWTuWu            | L-347 | WWWWuWTWuWu               |
| L-298 | WWWWuWTuWTuWu            | L-348 | WWWWuWTWuWu               |
| L-299 | WWWWuWTuWTuWu            | L-349 | WWWWuWTWuWu               |
| L-300 | WWWWuWTuTTuTTuWu         | L-350 | WWWWuWTuTTuTTuWu         |
| L-301 | WWWWuWTuTTuWTWu          | L-351 | WWWWuWTuTTuWTWu          |
| L-302 | WWWWuWTWuTTuWTWu         | L-352 | WWWWuWTuTTuWTWu          |
| L-303 | WWWWuWTWuTTuWTWu         | L-353 | WWWWuWTuTTuWTWu          |
| L-304 | WWWWuWTWuTTuWTWu         | L-354 | WWWWuWTuTTuWTWu          |
| L-305 | WWWWuWTWuTTuWTWu         | L-355 | WWWWuWTuTTuWTWu          |
| L-306 | WWWWuWTWuTTuWTWu         | L-356 | WWWWuWTuTTuWTWu          |
| L-307 | WWWWuWTWuTTuWTWu         | L-357 | WWWWuWTuTTuWTWu          |
| L-308 | WWWWuWTuTTuWTWu          | L-358 | WWWWuWTuTTuWTWu          |
| L-309 | WWWWuWTuTTuWTWu          | L-359 | WWWWuWTuTTuWTWu          |
| L-310 | WWWWuWTuTTuWTWu          | L-360 | WWWWuWTuTTuWTWu          |
| L-311 | WWWWuWTuTTuWTWu          | L-361 | WWWWuWTuTTuWTWu          |
| L-312 | WWWWuWTuTTuWTWu          | L-362 | WWWWuWTuTTuWTWu          |
| L-313 | WWWWuWTuTTuWTWu          | L-363 | WWWWuWTuTTuWTWu          |
| L-314 | WWWWuWTuTTuWTWu          | L-364 | WWWWuWTuTTuWTWu          |
| L-315 | WWWWuWTuTTuWTWu          | L-365 | WWWWuWTuTTuWTWu          |
| L-316 | WWWWuWTuTTuWTWu          | L-366 | WWWWuWTuTTuWTWu          |
| L-317 | WWWWuWTuTTuWTWu          | L-367 | WWWWuWTuTTuWTWu          |
| L-318 | WWWWuWTuTTuWTWu          | L-368 | WWWWuWTuTTuWTWu          |
| L-319 | WWWWuWTuTTuWTWu          | L-369 | WWWWuWTuTTuWTWu          |
| L-320 | WWWWuWTuTTuWTWu          | L-370 | WWWWuWTuTTuWTWu          |
| Index | Tie string                  | Index | Tie string                  |
|-------|-----------------------------|-------|-----------------------------|
| L-371 | TTuTWWuWuWTTTU              | L-421 | TTTWTTuTTuWWU              |
| L-372 | TTuTWWuWTWTTU              | L-422 | TTTWTTuWWuTTU              |
| L-373 | TTuWuTTuTTuWWuWuWuTTU      | L-423 | TTTWWTWTTTU                |
| L-374 | TTuWuTTuTuuWuWuTTU         | L-424 | TTTWWTWTTTU                |
| L-375 | TTuWuWTWTTTU               | L-425 | TTTWWTWTTTU                |
| L-376 | TTuWuWTWTTTU               | L-426 | TTTWWTWTTTU                |
| L-377 | TTuWuTTuWTTTU              | L-427 | TTTWWTWuTTuTTU             |
| L-378 | TTuWuWTTuWTTTU             | L-428 | TTTWWTuTTuWuTTU            |
| L-379 | TTuWuWuWuuTuTTuTTuTTU     | L-429 | TTTWWTuTTuWuWuTTU          |
| L-380 | TTuTTuTTuWTWuWuWu          | L-430 | TTTWWTuTTuWuuTTuWuWu       |
| L-381 | TTuTTuTTuWWuWTWu           | L-431 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-382 | TTuTTuTTuWTWTTuWuWuWu      | L-432 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-383 | TTuTTuWTWuWTTuWuWu          | L-433 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-384 | TTuTTuWTWuWTTuWuWu          | L-434 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-385 | TTuTTuWTWuWTTuWuWu          | L-435 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-386 | TTuTTuWTWuWTTuWuWu          | L-436 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-387 | TTuTTuWTWuWTTuWuWu          | L-437 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-388 | TTuTTuWTWuWTTuWuWu          | L-438 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-389 | TTuTTuWTWuWTTuWuWu          | L-439 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-390 | TTuTTuWTWuWTTuWuWu          | L-440 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-391 | TTuTTuWTWuWTTuWuWu          | L-441 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-392 | TTuTTuWTWuWTTuWuWu          | L-442 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-393 | TTuTTuWTWuWTTuWuWu          | L-443 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-394 | TTuTTuWTWuWTTuWuWu          | L-444 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-395 | TTuTTuWTWuWTTuWuWu          | L-445 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-396 | TTuTTuWTWuWTTuWuWu          | L-446 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-397 | TTuTTuWTWuWTTuWuWu          | L-447 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-398 | TTuTTuWTWuWTTuWuWu          | L-448 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-399 | TTuTTuWTWuWTTuWuWu          | L-449 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-400 | TTuTTuWTWuWTTuWuWu          | L-450 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-401 | TTuTTuWTWuWuWTWuWuWu        | L-451 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-402 | TTuTTuWTWuWuWTWuWuWu        | L-452 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-403 | TTuTTuWTWuWuWTWuWuWu        | L-453 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-404 | TTuTTuWTWuWuWTWuWuWu        | L-454 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-405 | TTuTTuWTWuWuWTWuWuWu        | L-455 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-406 | TTuTTuWTWuWuWTWuWuWu        | L-456 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-407 | TTuTTuWTWuWuWTWuWuWu        | L-457 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-408 | TTuTTuWTWuWuWTWuWuWu        | L-458 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-409 | TTuTTuWTWuWuWTWuWuWu        | L-459 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-410 | TTuTTuWTWuWuWTWuWuWu        | L-460 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-411 | TTuTTuWTWuWuWTWuWuWu        | L-461 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-412 | TTuTTuWTWuWuWTWuWuWu        | L-462 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-413 | TTuTTuWTWuWuWTWuWuWu        | L-463 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-414 | TTuTTuWTWuWuWTWuWuWu        | L-464 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-415 | TTuTTuWTWuWuWTWuWuWu        | L-465 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-416 | TTuTTuWTWuWuWTWuWuWu        | L-466 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-417 | TTuTTuWTWuWuWTWuWuWu        | L-467 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-418 | TTuTTuWTWuWuWTWuWuWu        | L-468 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-419 | TTuTTuWTWuWuWTWuWuWu        | L-469 | TTTWWTuTTuWuuTTuWuuTTu    |
| L-420 | TTuTTuWTWuWuWTWuWuWu        | L-470 | TTTWWTuTTuWuuTTuWuuTTu    |
| Index | Tie string               | Index   | Tie string               |
|-------|-------------------------|---------|-------------------------|
| L-471 | WTTuWuTTuTWuTTu        | L-521   | TWTuTWuWTuWTWu          |
| L-472 | WTTuWuWTTuTWuTTu       | L-522   | TWTuTWuWTuWuW          |
| L-473 | WTTuWuWTTuTWuTTu       | L-523   | TWTuTWuWTuWuW          |
| L-474 | WTTuWTuTTuTuTTuWuWu   | L-524   | TWTuTWuWTuWTWu          |
| L-475 | WTTuTuTuWuWuTTu        | L-525   | TWTuWTuTuWuWuW          |
| L-476 | WTTuTWuWTWTTuTu        | L-526   | TWTuWTuTuWuWuW          |
| L-477 | WTTuTWuWTWTTuTu        | L-527   | TWTuTuTuTuWTuWuWu       |
| L-478 | WTTuWTuTWuTTuTu        | L-528   | TWTuWTuTuWuWuW          |
| L-479 | WTTuTuWTWTTuTu         | L-529   | TWTuTuWuWuWuW           |
| L-480 | WTTuWTuWTuTuTuTu       | L-530   | TWTuWTuWTuTuTuTuTu     |
| L-481 | WTTuWTuWTuTuTuTuTu     | L-531   | TWTuWTuWTuTuTuTuTu     |
| L-482 | WTTuWTuWTuTuTuTuTuTu   | L-532   | TWTuWTuWTuTuTuTuTuTu   |
| L-483 | WTTuWTuWTuTuTuTuTuTu   | L-533   | TWTuWTuWTuTuTuTuTuTu   |
| L-484 | WTTuWTuWTuTuTuTuTuTu   | L-534   | TWTuWTuWTuTuTuTuTuTu   |
| L-485 | WTTuWTuWTuTuTuTuTuTu   | L-535   | TWTuWTuWTuTuTuTuTuTu   |
| L-486 | WTTuWTuWTuTuTuTuTuTu   | L-536   | TWTuWTuWTuTuTuTuTuTu   |
| L-487 | WTTuWTuWTuTuTuTuTuTu   | L-537   | TWTuWTuWTuTuTuTuTuTu   |
| L-488 | WTTuWTuWTuTuTuTuTuTu   | L-538   | TWTuWTuWTuTuTuTuTuTu   |
| L-489 | WTTuWTuWTuTuTuTuTuTu   | L-539   | TWTuWTuWTuTuTuTuTuTu   |
| L-490 | WTTuWTuWTuTuTuTuTuTu   | L-540   | TWTuWTuWTuTuTuTuTuTu   |
| L-491 | WTTuWTuWTuTuTuTuTuTu   | L-541   | TWTuWTuWTuTuTuTuTuTu   |
| L-492 | WTTuWTuWTuTuTuTuTuTu   | L-542   | TWTuWTuWTuTuTuTuTuTu   |
| L-493 | WTTuWTuWTuTuTuTuTuTu   | L-543   | TWTuWTuWTuTuTuTuTuTu   |
| L-494 | WTTuWTuWTuTuTuTuTuTu   | L-544   | TWTuWTuWTuTuTuTuTuTu   |
| L-495 | WTTuWTuWTuTuTuTuTuTu   | L-545   | TWTuWTuWTuTuTuTuTuTu   |
| L-496 | WTTuWTuWTuTuTuTuTuTu   | L-546   | TWTuWTuWTuTuTuTuTuTu   |
| L-497 | WTTuWTuWTuTuTuTuTuTu   | L-547   | TWTuWTuWTuTuTuTuTuTu   |
| L-498 | WTTuWTuWTuTuTuTuTuTu   | L-548   | TWTuWTuWTuTuTuTuTuTu   |
| L-499 | WTTuWTuWTuTuTuTuTuTu   | L-549   | TWTuWTuWTuTuTuTuTuTu   |
| L-500 | WTTuWTuWTuTuTuTuTuTu   | L-550   | TWTuWTuWTuTuTuTuTuTu   |
| L-501 | WTTuWTuWTuTuTuTuTuTu   | L-551   | TWTuWTuWTuTuTuTuTuTu   |
| L-502 | WTTuWTuWTuTuTuTuTuTu   | L-552   | TWTuWTuWTuTuTuTuTuTu   |
| L-503 | WTTuWTuWTuTuTuTuTuTu   | L-553   | TWTuWTuWTuTuTuTuTuTu   |
| L-504 | WTTuWTuWTuTuTuTuTuTu   | L-554   | TWTuWTuWTuTuTuTuTuTu   |
| L-505 | WTTuWTuWTuTuTuTuTuTu   | L-555   | TWTuWTuWTuTuTuTuTuTu   |
| L-506 | WTTuWTuWTuTuTuTuTuTu   | L-556   | TWTuWTuWTuTuTuTuTuTu   |
| L-507 | WTTuWTuWTuTuTuTuTuTu   | L-557   | TWTuWTuWTuTuTuTuTuTu   |
| L-508 | WTTuWTuWTuTuTuTuTuTu   | L-558   | TWTuWTuWTuTuTuTuTuTu   |
| L-509 | WTTuWTuWTuTuTuTuTuTu   | L-559   | TWTuWTuWTuTuTuTuTuTu   |
| L-510 | WTTuWTuWTuTuTuTuTuTu   | L-560   | TWTuWTuWTuTuTuTuTuTu   |
| L-511 | WTTuWTuWTuTuTuTuTuTu   | L-561   | TWTuWTuWTuTuTuTuTuTu   |
| L-512 | WTTuWTuWTuTuTuTuTuTu   | L-562   | TWTuWTuWTuTuTuTuTuTu   |
| L-513 | WTTuWTuWTuTuTuTuTuTu   | L-563   | TWTuWTuWTuTuTuTuTuTu   |
| L-514 | WTTuWTuWTuTuTuTuTuTu   | L-564   | TWTuWTuWTuTuTuTuTuTu   |
| L-515 | WTTuWTuWTuTuTuTuTuTu   | L-565   | TWTuWTuWTuTuTuTuTuTu   |
| L-516 | WTTuWTuWTuTuTuTuTuTu   | L-566   | TWTuWTuWTuTuTuTuTuTu   |
| L-517 | WTTuWTuWTuTuTuTuTuTu   | L-567   | TWTuWTuWTuTuTuTuTuTu   |
| L-518 | WTTuWTuWTuTuTuTuTuTu   | L-568   | TWTuWTuTuTuTuTuTuTu   |
| L-519 | WTTuWTuWTuTuTuTuTuTu   | L-569   | TWTuWTuTuTuTuTuTuTu   |
| L-520 | WTTuWTuWTuTuTuTuTuTu   | L-570   | TWTuWTuTuTuTuTuTuTu   |
| Index | Tie string       | Index | Tie string       |
|-------|------------------|-------|------------------|
| L-571 | TWWuWWuTWuWTuTTu | L-621 | WTTTTuWuTWuTTu  |
| L-572 | TWWuWWuTWTuWWu  | L-622 | WTTTTuWuWuWuWu  |
| L-573 | TWWuWWuTWuWTuTTu | L-623 | WTTWTuTWuTWuWu  |
| L-574 | TWWuWWuWuWuWuWu  | L-624 | WTTWTuTWuTWuWu  |
| L-575 | TWWuWWuWuWuTWWu  | L-625 | WTTWTuTWuTWuWu  |
| L-576 | WTTuWTuWuWuWuWu  | L-626 | WTTWTuTWuTWuWu  |
| L-577 | WTTuWTWTuWuWuWu  | L-627 | WTTWTuTWuWuTTu  |
| L-578 | WTTuWTWWTuWuWuWu | L-628 | WTTWTuTWuWuTTu  |
| L-579 | WTTuWTWuWuWuWuWu | L-629 | WTTWTuTWuWuTTu  |
| L-580 | WTTuWTWTuWuWuWu  | L-630 | WTTWTuTWuWuTTu  |
| L-581 | WTTuWTWTuWuWuWu  | L-631 | WTTWTuTWuWuTTu  |
| L-582 | WTTuWTWTuWuWuWu  | L-632 | WTTWTuTWuWuTTu  |
| L-583 | WTTuWTWTuWuWuWu  | L-633 | WTTWTuTWuWuTTu  |
| L-584 | WTTuWTWTuWuWuWu  | L-634 | WTTWTuTWuWuTTu  |
| L-585 | WTTuWTWTuWuWuWu  | L-635 | WTTWTuTWuWuTTu  |
| L-586 | WTTuWTWTuWuWuWu  | L-636 | WTTWTuTWuWuTTu  |
| L-587 | WTTuWTWTuWuWuWu  | L-637 | WTTWTuTWuWuTTu  |
| L-588 | WTTuWTWTuWuWuWu  | L-638 | WTTWTuTWuWuTTu  |
| L-589 | WTTuWTWTuWuWuWu  | L-639 | WTTWTuTWuWuTTu  |
| L-590 | WTTuWTWTuWuWuWu  | L-640 | WTTWTuTWuWuTTu  |
| L-591 | WTTuWTWTuWuWuWu  | L-641 | WTTWTuTWuWuTTu  |
| L-592 | WTTuWTWTuWuWuWu  | L-642 | WTTWTuTWuWuTTu  |
| L-593 | WTTuWTWTuWuWuWu  | L-643 | WTTWTuTWuWuTTu  |
| L-594 | WTTuWTWTuWuWuWu  | L-644 | WTTWTuTWuWuTTu  |
| L-595 | WTTuWTWTuWuWuWu  | L-645 | WTTWTuTWuWuTTu  |
| L-596 | WTTuWTWTuWuWuWu  | L-646 | WTTWTuTWuWuTTu  |
| L-597 | WTTuWTWTuWuWuWu  | L-647 | WTTWTuTWuWuTTu  |
| L-598 | WTTuWTWTuWuWuWu  | L-648 | WTTWTuTWuWuTTu  |
| L-599 | WTTuWTWTuWuWuWu  | L-649 | WTTWTuTWuWuTTu  |
| L-600 | WTTuWTWTuWuWuWu  | L-650 | WTTWTuWuWuTTu  |
| L-601 | WTTuWTWTuWuWuWu  | L-651 | WTTWTuWuWuTTu  |
| L-602 | WTTuWTWTuWuWuWu  | L-652 | WTTWTuWuWuTTu  |
| L-603 | WTTuWTWTuWuWuWu  | L-653 | WTTWTuWuWuTTu  |
| L-604 | WTTuWTWTuWuWuWu  | L-654 | WTTWTuWuWuTTu  |
| L-605 | WTTuWTWTuWuWuWu  | L-655 | WTTWTuWuWuTTu  |
| L-606 | WTTuWTWTuWuWuWu  | L-656 | WTTWTuWuWuTTu  |
| L-607 | WTTuWTWTuWuWuWu  | L-657 | WTTWTuWuWuTTu  |
| L-608 | WTTuWTWTuWuWuWu  | L-658 | WTTWTuWuWuTTu  |
| L-609 | WTTuWTWTuWuWuWu  | L-659 | WTTWTuWuWuTTu  |
| L-610 | WTTuWTWTuWuWuWu  | L-660 | WTTWTuWuWuTTu  |
| L-611 | WTTuWTWTuWuWuWu  | L-661 | WTTWTuWuWuTTu  |
| L-612 | WTTuWTWTuWuWuWu  | L-662 | WTTWTuWuWuTTu  |
| L-613 | WTTuWTWTuWuWuWu  | L-663 | WTTWTuWuWuTTu  |
| L-614 | WTTuWTWTuWuWuWu  | L-664 | WTTWTuWuWuTTu  |
| L-615 | WTTuWTWTuWuWuWu  | L-665 | WTTWTuWuWuTTu  |
| L-616 | WTTuWTWTuWuWuWu  | L-666 | WTTWTuWuWuTTu  |
| L-617 | WTTuWTWTuWuWuWu  | L-667 | WTTWTuWuWuTTu  |
| L-618 | WTTuWTWTuWuWuWu  | L-668 | WTTWTuWuWuTTu  |
| L-619 | WTTuWTWTuWuWuWu  | L-669 | WTTWTuWuWuTTu  |
| L-620 | WTTuWTWTuWuWuWu  | L-670 | WTTWTuWuWuTTu  |
| Index | Tie string |
|-------|------------|
| L-671 | WWWuWWuTWWTTTU |
| L-672 | WWWuWWuTWTWTTU |
| L-673 | WWWuWWuWWuTTuTTU |
| L-674 | TWWWuWWuWWuWWuWWU |
| L-675 | WWTWWWuWWuWWuWWU |
| L-676 | WTWuWWuWWuWWuWWU |
| L-677 | WWWuWTWWuWWuWWU |
| L-678 | WWWuTWWWuWWuWWU |
| L-679 | WWWuWWuWTTWWuWWU |
| L-680 | WWWuWuTTWWuWWU |
| L-681 | WWWuWWuWWuWTTWWU |
| L-682 | WWWuWWuWWuWTTWWU |
Next, we list the knots that tuck from the right. Again, with all optional single depth front tuck sites marked with a lowercase u.

| Index | Tie string | Index | Tie string | Index | Tie string |
|-------|------------|-------|------------|-------|------------|
| R-1   | TTU        | R-41  | WWuWTWWu   | R-81  | WWuWTWWu   |
| R-2   | TWuWu      | R-42  | WWWuWTWWu  | R-82  | WWuWTWWu   |
| R-3   | WTTuTU     | R-43  | TTuTTuTTuTU| R-83  | WWuWTWWu   |
| R-4   | WTTuTU     | R-44  | TTuTTuWTWWu| R-84  | WWuWTWWu   |
| R-5   | WWTuWu     | R-45  | TTuTTuTWu  | R-85  | WWuWTWWu   |
| R-6   | TTTuTTu    | R-46  | TTuWTuTTu  | R-86  | TTuTTuTTu  |
| R-7   | TWTuWWu    | R-47  | TTuWTuWTu  | R-87  | TTuTTuWTu  |
| R-8   | TWuWu      | R-48  | TTuWTuWTu  | R-88  | TTuTTuWTu  |
| R-9   | WTTuWu     | R-49  | TTuTWuTWu  | R-89  | TTuTWuWTu  |
| R-10  | WWuTuTTu   | R-50  | TTuWuWTu   | R-90  | TTuWTuTTu  |
| R-11  | TTuTuWuTuWu| R-51  | TTuWuWTuTu | R-91  | TTuWTuWTu  |
| R-12  | TTuWuTuTu  | R-52  | WTTuTuWTu  | R-92  | TTuWTuWTu  |
| R-13  | WTTuTTu    | R-53  | WTTuWuWTu  | R-93  | TWTuWTuTTu |
| R-14  | WTTuTTu    | R-54  | WTTuWTuTu  | R-94  | TWTuWTuTTu |
| R-15  | WTTuTTu    | R-55  | WTTuWTuTu  | R-95  | TWTuWTuTTu |
| R-16  | WTTuTTu    | R-56  | WTTuWTuTu  | R-96  | TWTuWTuTTu |
| R-17  | WWuTuTTu   | R-57  | WTTuWTuTu  | R-97  | TWTuWTuTTu |
| R-18  | WTTuWTuWu  | R-58  | WTTuWTuTu  | R-98  | TWTuWTuTTu |
| R-19  | WTTuWTuWu  | R-59  | WTTuWTuTu  | R-99  | TWTuWTuTTu |
| R-20  | WWuWTuWu   | R-60  | WTTuWTuTu  | R-100 | TWTuWTuTTu |
| R-21  | WWuWWTWu   | R-61  | TWTuWTuTu  | R-101 | WWTuTTuTTu |
| R-22  | WTTuWTWu   | R-62  | TWTuWTuTu  | R-102 | WWTuTTuTTu |
| R-23  | WTTuWTuTTu | R-63  | TWTuWTuTu  | R-103 | WWTuTTuTTu |
| R-24  | WTTuWTuTTu | R-64  | TWTuWTuTu  | R-104 | WWTuTTuTTu |
| R-25  | WTTuWTuTTu | R-65  | TWTuWTuTu  | R-105 | WWTuTTuTTu |
| R-26  | WTTuWTuTTu | R-66  | WWTuWTuTu  | R-106 | WWTuTTuTTu |
| R-27  | WTTuWTuWu  | R-67  | WWTuWTuTu  | R-107 | WWTuTTuTTu |
| R-28  | WTTuWTuWu  | R-68  | WWTuWTuTu  | R-108 | WWTuTTuTTu |
| R-29  | WTTuWTuWu  | R-69  | WWTuWTuTu  | R-109 | WWTuTTuTTu |
| R-30  | WTTuWTuWu  | R-70  | WWTuWTuTu  | R-110 | WWTuTTuTTu |
| R-31  | WTTuWTuWu  | R-71  | WWTuWTuTu  | R-111 | WWTuTTuTTu |
| R-32  | WTTuWTuWu  | R-72  | WWTuWTuTu  | R-112 | WWTuTTuTTu |
| R-33  | WTTuWTuWu  | R-73  | WWTuWTuTu  | R-113 | WWTuTTuTTu |
| R-34  | WTTuWTuWu  | R-74  | WWTuWTuTu  | R-114 | WWTuTTuTTu |
| R-35  | WTTuWTuWu  | R-75  | WWTuWTuTu  | R-115 | WWTuTTuTTu |
| R-36  | WTTuWTuWu  | R-76  | WWTuWTuTu  | R-116 | WWTuTTuTTu |
| R-37  | WTTuWTuWu  | R-77  | WWTuWTuTu  | R-117 | WWTuTTuTTu |
| R-38  | WTTuWTuWu  | R-78  | WWTuWTuTu  | R-118 | WWTuTTuTTu |
| R-39  | WTTuWTuWu  | R-79  | WWTuWTuTu  | R-119 | WWTuTTuTTu |
| R-40  | WTTuWTuWu  | R-80  | WWTuWTuTu  | R-120 | WWTuTTuTTu |
| Index | Tie string                  | Index | Tie string                  |
|-------|----------------------------|-------|----------------------------|
| R-271 | WWuWTWTTuWTuWWU            | R-321 | WWuWTWTTWWuAAA            |
| R-272 | WWuWTWTTuWTuWWU            | R-322 | WWuWTWTTWWuAAA            |
| R-273 | WWuWTWTTuWTuWWU            | R-323 | WWuWTWTTWWuAAA            |
| R-274 | WWuWTWTTuWTuWWU            | R-324 | WWuWTWTTWWuAAA            |
| R-275 | WWuWTWTTuWTuWWU            | R-325 | WWuWTWTTWWuAAA            |
| R-276 | WWuWTWTTuWTuWWU            | R-326 | WWuWTWTTWWuAAA            |
| R-277 | WWuWTWTTuWTuWWU            | R-327 | WWuWTWTTWWuAAA            |
| R-278 | WWuWTWTTuWTuWWU            | R-328 | WWuWTWTTWWuAAA            |
| R-279 | WWuWTWTTuWTuWWU            | R-329 | WWuWTWTTWWuAAA            |
| R-280 | WWuWTWTTuWTuWWU            | R-330 | WWuWTWTTWWuAAA            |
| R-281 | WWuWTWTTuWTuWWU            | R-331 | WWuWTWTTWWuAAA            |
| R-282 | WWuWTWTTuWTuWWU            | R-332 | WWuWTWTTWWuAAA            |
| R-283 | WWuWTWTTuWTuWWU            | R-333 | WWuWTWTTWWuAAA            |
| R-284 | WWuWTWTTuWTuWWU            | R-334 | WWuWTWTTWWuAAA            |
| R-285 | WWuWTWTTuWTuWWU            | R-335 | WWuWTWTTWWuAAA            |
| R-286 | WWuWTWTTuWTuWWU            | R-336 | WWuWTWTTWWuAAA            |
| R-287 | WWuWTWTTuWTuWWU            | R-337 | WWuWTWTTWWuAAA            |
| R-288 | WWuWTWTTuWTuWWU            | R-338 | WWuWTWTTWWuAAA            |
| R-289 | WWuWTWTTuWTuWWU            | R-339 | WWuWTWTTWWuAAA            |
| R-290 | WWuWTWTTuWTuWWU            | R-340 | WWuWTWTTWWuAAA            |
| R-291 | WWuWTWTTuWTuWWU            | R-341 | WWuWTWTTWWuAAA            |
| R-292 | WWuWTWTTuWTuWWU            | R-342 | WWuWTWTTWWuAAA            |
| R-293 | WWuWTWTTuWTuWWU            | R-343 | WWuWTWTTWWuAAA            |
| R-294 | WWuWTWTTuWTuWWU            | R-344 | WWuWTWTTWWuAAA            |
| R-295 | WWuWTWTTuWTuWWU            | R-345 | WWuWTWTTWWuAAA            |
| R-296 | WWuWTWTTuWTuWWU            | R-346 | WWuWTWTTWWuAAA            |
| R-297 | WWuWTWTTuWTuWWU            | R-347 | WWuWTWTTWWuAAA            |
| R-298 | WWuWTWTTuWTuWWU            | R-348 | WWuWTWTTWWuAAA            |
| R-299 | WWuWTWTTuWTuWWU            | R-349 | WWuWTWTTWWuAAA            |
| R-300 | WWuWTWTTuWTuWWU            | R-350 | WWuWTWTTWWuAAA            |
| R-301 | WWuWTWTTuWTuWWU            | R-351 | WWuWTWTTWWuAAA            |
| R-302 | WWuWTWTTuWTuWWU            | R-352 | WWuWTWTTWWuAAA            |
| R-303 | WWuWTWTTuWTuWWU            | R-353 | WWuWTWTTWWuAAA            |
| R-304 | WWuWTWTTuWTuWWU            | R-354 | WWuWTWTTWWuAAA            |
| R-305 | WWuWTWTTuWTuWWU            | R-355 | WWuWTWTTWWuAAA            |
| R-306 | WWuWTWTTuWTuWWU            | R-356 | WWuWTWTTWWuAAA            |
| R-307 | WWuWTWTTuWTuWWU            | R-357 | WWuWTWTTWWuAAA            |
| R-308 | WWuWTWTTuWTuWWU            | R-358 | WWuWTWTTWWuAAA            |
| R-309 | WWuWTWTTuWTuWWU            | R-359 | WWuWTWTTWWuAAA            |
| R-310 | WWuWTWTTuWTuWWU            | R-360 | WWuWTWTTWWuAAA            |
| R-311 | WWuWTWTTuWTuWWU            | R-361 | WWuWTWTTWWuAAA            |
| R-312 | WWuWTWTTuWTuWWU            | R-362 | WWuWTWTTWWuAAA            |
| R-313 | WWuWTWTTuWTuWWU            | R-363 | WWuWTWTTWWuAAA            |
| R-314 | WWuWTWTTuWTuWWU            | R-364 | WWuWTWTTWWuAAA            |
| R-315 | WWuWTWTTuWTuWWU            | R-365 | WWuWTWTTWWuAAA            |
| R-316 | WWuWTWTTuWTuWWU            | R-366 | WWuWTWTTWWuAAA            |
| R-317 | WWuWTWTTuWTuWWU            | R-367 | WWuWTWTTWWuAAA            |
| R-318 | WWuWTWTTuWTuWWU            | R-368 | WWuWTWTTWWuAAA            |
| R-319 | WWuWTWTTuWTuWWU            | R-369 | WWuWTWTTWWuAAA            |
| R-320 | WWuWTWTTuWTuWWU            | R-370 | WWuWTWTTWWuAAA            |
| Index | Tie string   | Index   | Tie string       |
|-------|--------------|---------|------------------|
| R-371 | TWTuTWuWTuTTu | R-414  | TWTuWTWTuWTuTTu |
| R-372 | TWTuTWuWTuTTu | R-415  | TWTuWTWTuWTuTTu |
| R-373 | TWTuTWuTWuTTu | R-416  | TWTuWTWTuWTuTTu |
| R-374 | TWTuTWuTWuTTu | R-417  | TWTuWTWTuWTuTTu |
| R-375 | TWTuTWuWuTTuTTu | R-418  | TWTuWTWTuWTuTTu |
| R-376 | TWTuTWuWuTTuTTu | R-419  | TWTuWTWTuWTuTTu |
| R-377 | TWTuTWuWTWuTTu | R-420  | TWTuWTWTuWTuTTu |
| R-378 | TWTuTWuWTWuTTu | R-421  | TWTuWTWTuWTuTTu |
| R-379 | TWTuTWuWTWuTTu | R-422  | WTTuTWuTTuTTuTTu |
| R-380 | TWTuTWuWTWuTTu | R-423  | TWTuTWuTTuTTuTTu |
| R-381 | TWTuTWuWTWuTTu | R-424  | TWTuTWuTTuTTuTTu |
| R-382 | TWTuTWuWTWuTTu | R-425  | TWTuTWuTTuTTuTTu |
| R-383 | TWTuTWuWTWuTTu | R-426  | TWTuTWuTTuTTuTTu |
| R-384 | TWTuTWuWTWuTTu | R-427  | TWTuTWuTTuTTuTTu |
| R-385 | TWTuTWuWTWuTTu | R-428  | TWTuTWuTTuTTuTTu |
| R-386 | TWTuTWuWTWuTTu | R-429  | TWTuTWuTTuTTuTTu |
| R-387 | TWTuTWuWTWuTTu | R-430  | TWTuTWuTTuTTuTTu |
| R-388 | TWTuTWuWTWuTTu | R-431  | TWTuTWuTTuTTuTTu |
| R-389 | TWTuTWuWTWuTTu | R-432  | TWTuTWuTTuTTuTTu |
| R-390 | TWTuTWuWTWuTTu | R-433  | TWTuTWuTTuTTuTTu |
| R-391 | TWTuTWuWTWuTTu | R-434  | TWTuTWuTTuTTuTTu |
| R-392 | TWTuTWuWTWuTTu | R-435  | TWTuTWuTTuTTuTTu |
| R-393 | TWTuTWuWTWuTTu | R-436  | TWTuTWuTTuTTuTTu |
| R-394 | TWTuTWuWTWuTTu | R-437  | TWTuTWuTTuTTuTTu |
| R-395 | TWTuTWuWTWuTTu | R-438  | TWTuTWuTTuTTuTTu |
| R-396 | TWTuTWuWTWuTTu | R-439  | TWTuTWuTTuTTuTTu |
| R-397 | TWTuTWuWTWuTTu | R-440  | TWTuTWuTTuTTuTTu |
| R-398 | TWTuTWuWTWuTTu | R-441  | TWTuTWuTTuTTuTTu |
| R-399 | TWTuTWuWTWuTTu | R-442  | TWTuTWuTTuTTuTTu |
| R-400 | TWTuTWuWTWuTTu | R-443  | TWTuTWuTTuTTuTTu |
| R-401 | TWTuTWuWTWuTTu | R-444  | TWTuTWuTTuTTuTTu |
| R-402 | TWTuTWuWTWuTTu | R-445  | TWTuTWuTTuTTuTTu |
| R-403 | TWTuTWuWTWuTTu | R-446  | TWTuTWuTTuTTuTTu |
| R-404 | TWTuTWuWTWuTTu | R-447  | TWTuTWuTTuTTuTTu |
| R-405 | TWTuTWuWTWuTTu | R-448  | TWTuTWuTTuTTuTTu |
| R-406 | TWTuTWuWTWuTTu | R-449  | TWTuTWuTTuTTuTTu |
| R-407 | TWTuTWuWTWuTTu | R-450  | TWTuTWuTTuTTuTTu |
| R-408 | TWTuTWuWTWuTTu | R-451  | TWTuTWuTTuTTuTTu |
| R-409 | TWTuTWuWTWuTTu | R-452  | TWTuTWuTTuTTuTTu |
| R-410 | TWTuTWuWTWuTTu | R-453  | TWTuTWuTTuTTuTTu |
| R-411 | TWTuTWuWTWuTTu | R-454  | TWTuTWuTTuTTuTTu |
| R-412 | TWTuTWuWTWuTTu | R-455  | TWTuTWuTTuTTuTTu |
| R-413 | TWTuTWuWTWuTTu | R-456  | TWTuTWuTTuTTuTTu |
| R-414 | TWTuTWuWTWuTTu | R-457  | TWTuTWuTTuTTuTTu |
| R-415 | TWTuTWuWTWuTTu | R-458  | TWTuTWuTTuTTuTTu |
| R-416 | TWTuTWuWTWuTTu | R-459  | TWTuTWuTTuTTuTTu |
| R-417 | TWTuTWuWTWuTTu | R-460  | TWTuTWuTTuTTuTTu |
| R-418 | TWTuTWuWTWuTTu | R-461  | TWTuTWuTTuTTuTTu |
| R-419 | TWTuTWuWTWuTTu | R-462  | TWTuTWuTTuTTuTTu |
| R-420 | TWTuTWuWTWuTTu | R-463  | TWTuTWuTTuTTuTTu |
|              |              | R-470  | TWTuTWuTTuTTuTTu |
| Index | Tie string                  | Index | Tie string                  |
|-------|----------------------------|-------|----------------------------|
| R-471 | TWTTWuTTuTWuWuWu           | R-521 | TWuTWuTWuTTuWWu            |
| R-472 | TWTTWuWTTuWuWuWu           | R-522 | TWuTWuTWuTTuWu             |
| R-473 | TWTTWuWTuWuWuWuWuWu        | R-523 | TWuTWuWTuTTuWuWuWu         |
| R-474 | TWTTWuWTuTTuWuWuWuWu       | R-524 | TWuTWuWTuTTuWuWuWu         |
| R-475 | TWTTWuTWuWuWuWuWuWu        | R-525 | TWuTWuTWuWuTTuWuWu         |
| R-476 | TWTTWuWuWuWuWuWuWuWu       | R-526 | TWuTWuWuWuWuWuWuWu         |
| R-477 | TWTTWuWuWuWuWuWuWuWu       | R-527 | TWuWuWuWuWuWuWuWuWuWu     |
| R-478 | TTWTuWTuWuWuWuWuWuWu       | R-528 | TWuWuWuWuWuWuWuWuWuWu     |
| R-479 | TTWTuWuWuWuWuWuWuWuWu     | R-529 | TWuWuWuWuWuWuWuWuWuWu     |
| R-480 | TTWTuTuWuWuWuWuWuWu       | R-530 | TWuWuWuWuWuWuWuWuWuWu     |
| R-481 | TTWTuWuWuWuWuWuWuWuWu     | R-531 | TWuWuWuWuWuWuWuWuWuWu     |
| R-482 | TTWTuWuWuWuWuWuWuWuWu     | R-532 | TWuWuWuWuWuWuWuWuWuWu     |
| R-483 | TTWTWTTuTTuWWuWu           | R-533 | TWuWuWuWuWuWuWuWuWuWu     |
| R-484 | TTWTWTTuWWuWWuWu           | R-534 | TTuTuTuTWuWuWuWuWu        |
| R-485 | TTWTWTTuWWuWWuWu           | R-535 | TTuTuTuTWuWuWuWuWu        |
| R-486 | TTWTWTTuTTuWWuWu           | R-536 | TTuTuTuTwuWuWuWuWuWu     |
| R-487 | TTWTWuTuTWuWuWuWuWu        | R-537 | TTuTuTuWuWuWuWuWuWuWu     |
| R-488 | TTWTWuTuWuWuWuWuWuWu       | R-538 | TTuTuTuWuWuWuWuWuWuWu     |
| R-489 | TTWTWuTuWuWuWuWuWuWu       | R-539 | TTuTuTuTWuWuWuWuWuWu     |
| R-490 | TTWTuWTuTTuWWuWu           | R-540 | TTuTuTuTWuWuWuWuWuWu     |
| R-491 | TTWTuWTuTTuWWuWu           | R-541 | TTuTuTuTWuWuWuWuWuWu     |
| R-492 | TTWTuWTuTTuWWuWu           | R-542 | TTuTuTuTWuWuWuWuWuWu     |
| R-493 | TTWTuWTuTTuWWuWu           | R-543 | TTuTuTuTWuWuWuWuWuWu     |
| R-494 | TTWTuWTuTTuWWuWu           | R-544 | TTuTuTuTWuWuWuWuWuWu     |
| R-495 | TTWTuWTuTTuWWuWuWuWuWu     | R-545 | TTuTuTuTWuWuWuWuWuWuWu   |
| R-496 | TTWTuWTuTTuWWuWuWuWuWu     | R-546 | TTuTuTuTWuWuWuWuWuWuWu   |
| R-497 | TTWTuWTuTTuWWuWuWuWuWu     | R-547 | TTuTuTuTWuWuWuWuWuWuWu   |
| R-498 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-548 | TTuTuTuTWuWuWuWuWuWuWu   |
| R-499 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-549 | TTuTuTuTWuWuWuWuWuWuWu   |
| R-500 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-550 | TTuTuTuTWuWuWuWuWuWuWuWu |
| R-501 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-551 | TTuTuTuTWuWuWuWuWuWuWuWu |
| R-502 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-552 | TTuTuTuTWuWuWuWuWuWuWuWu |
| R-503 | TTWTuWTuTTuWWuWuWuWuWuWu   | R-553 | TTuTuTuTWuWuWuWuWuWuWuWu |
| R-504 | TTWTuTTuTuWuWuWuWuWuWuWu   | R-554 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-505 | TTuTuTuTuTWuWuWuWuWuWuWu   | R-555 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-506 | TTuTuTuTuTWuWuWuWuWuWuWu   | R-556 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-507 | TTuTuTuTuTWuWuWuWuWuWuWu   | R-557 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-508 | TTuTuTuTuTWuWuWuWuWuWuWu   | R-558 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-509 | TTuTuTuTuWuWuWuWuWuWuWuWu   | R-559 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-510 | TTuTuTuTuWuWuWuWuWuWuWuWu   | R-560 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-511 | TTuTuTuTWuWuWuWuWuWuWuWu   | R-561 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-512 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-562 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-513 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-563 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-514 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-564 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-515 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-565 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-516 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-566 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-517 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-567 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-518 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-568 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-519 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-569 | TTuTuTuWTuWuWuWuWuWuWuWu |
| R-520 | TTuTuTuWTuWuWuWuWuWuWuWu   | R-570 | TTuTuTuWTuWuWuWuWuWuWuWu |
| Index | Tie string                   | Index | Tie string                   |
|-------|-----------------------------|-------|-----------------------------|
| R-571 | WWTWTTWTuWWuWWu            | R-621 | WWuTuTuTuTWWuWWu            |
| R-572 | WWTWTTWuTuWWuWWu            | R-622 | WWuTuTuTuTWWuWWu            |
| R-573 | WWTWWTuWTuWTTuW            | R-623 | WWTuTuWWTWuWWu              |
| R-574 | WWTWTWTuWTuWTTuW            | R-624 | WWTuTuWTWuWTWuWWu            |
| R-575 | WWTWTuWTuWTWuWWu            | R-625 | WWTuTuWTWuWTWuWWu            |
| R-576 | WWTWTuWTuWTWuWWu            | R-626 | WWTuTuWTWuWTWuWWu            |
| R-577 | WWTWTuWTuWTWuWWu            | R-627 | WWTuTuWTWuWTWuWWu            |
| R-578 | WWTWTuWTuWTWuWWu            | R-628 | WWTuTuWTWuWTWuWWu            |
| R-579 | WWTWTuWTuWTWuWWu            | R-629 | WWTuTuWTWuWTWuWWu            |
| R-580 | WWTWTuWTuWTWuWWu            | R-630 | WWTuTuWTWuWTWuWWu            |
| R-581 | WWTWTuWTuWTWuWWu            | R-631 | WWTuTuWTWuWTWuWWu            |
| R-582 | WWTWTuWTuWTWuWWu            | R-632 | WWTuTuWTWuWTWuWWu            |
| R-583 | WWTWTuWTuWTWuWWu            | R-633 | WWTuTuWTWuWTWuWWu            |
| R-584 | WWTWTuWTuWTWuWWu            | R-634 | WWTuTuWTWuWTWuWWu            |
| R-585 | WWTWTuWTuWTWuWWu            | R-635 | WWTuTuWTWuWTWuWWu            |
| R-586 | WWTWTuWTuWTWuWWu            | R-636 | WWTuTuWTWuWTWuWWu            |
| R-587 | WWTWTuWTuWTWuWWu            | R-637 | WWTuTuWTWuWTWuWWu            |
| R-588 | WWTWTuWTuWTWuWWu            | R-638 | WWTuTuWTWuWTWuWWu            |
| R-589 | WWTWTuWTuWTWuWWu            | R-639 | WWTuTuWTWuWTWuWWu            |
| R-590 | WWTWTuWTuWTWuWWu            | R-640 | WWTuTuWTWuWTWuWWu            |
| R-591 | WWTWTuWTuWTWuWWu            | R-641 | WWTuTuWTWuWTWuWWu            |
| R-592 | WWTWTuWTuWTWuWWu            | R-642 | WWTuTuWTWuWTWuWWu            |
| R-593 | WWTWTuWTuWTWuWWu            | R-643 | WWTuTuWTWuWTWuWWu            |
| R-594 | WWTWTuWTuWTWuWWu            | R-644 | WWTuTuWTWuWTWuWWu            |
| R-595 | WWTWTuWTuWTWuWWu            | R-645 | WWTuTuWTWuWTWuWWu            |
| R-596 | WWTWTuWTuWTWuWWu            | R-646 | WWTuTuWTWuWTWuWWu            |
| R-597 | WWTWTuWTuWTWuWWu            | R-647 | WWTuTuWTWuWTWuWWu            |
| R-598 | WWTWTuWTuWTWuWWu            | R-648 | WWTuTuWTWuWTWuWWu            |
| R-599 | WWTWTuWTuWTWuWWu            | R-649 | WWTuTuWTWuWTWuWWu            |
| R-600 | WWTWTuWTuWTWuWWu            | R-650 | WWTuTuWTWuWTWuWWu            |
| R-601 | WWTWTuWTuWTWuWWu            | R-651 | WWTuTuWTWuWTWuWWu            |
| R-602 | WWTWTuWTuWTWuWWu            | R-652 | WWTuTuWTWuWTWuWWu            |
| R-603 | WWTWTuWTuWTWuWWu            | R-653 | WWTuTuWTWuWTWuWWu            |
| R-604 | WWTWTuWTuWTWuWWu            | R-654 | WWTuTuWTWuWTWuWWu            |
| R-605 | WWTWTuWTuWTWuWWu            | R-655 | WWTuTuWTWuWTWuWWu            |
| R-606 | WWTWTuWTuWTWuWWu            | R-656 | WWTuTuWTWuWTWuWWu            |
| R-607 | WWTWTuWTuWTWuWWu            | R-657 | WWTuTuWTWuWTWuWWu            |
| R-608 | WWTWTuWTuWTWuWWu            | R-658 | WWTuTuWTWuWTWuWWu            |
| R-609 | WWTWTuWTuWTWuWWu            | R-659 | WWTuTuWTWuWTWuWWu            |
| R-610 | WWTWTuWTuWTWuWWu            | R-660 | WWTuTuWTWuWTWuWWu            |
| R-611 | WWTWTuWTuWTWuWWu            | R-661 | WWTuTuWTWuWTWuWWu            |
| R-612 | WWTWTuWTuWTWuWWu            | R-662 | WWTuTuWTWuWTWuWWu            |
| R-613 | WWTWTuWTuWTWuWWu            | R-663 | WWTuTuWTWuWTWuWWu            |
| R-614 | WWTWTuWTuWTWuWWu            | R-664 | WWTuTuWTWuWTWuWWu            |
| R-615 | WWTWTuWTuWTWuWWu            | R-665 | WWTuTuWTWuWTWuWWu            |
| R-616 | WWTWTuWTuWTWuWWu            | R-666 | WWTuTuWTWuWTWuWWu            |
| R-617 | WWTWTuWTuWTWuWWu            | R-667 | WWTuTuWTWuWTWuWWu            |
| R-618 | WWTWTuWTuWTWuWWu            | R-668 | WWTuTuWTWuWTWuWWu            |
| R-619 | WWTWTuWTuWTWuWWu            | R-669 | WWTuTuWTWuWTWuWWu            |
| R-620 | WWTWTuWTuWTWuWWu            | R-670 | WWTuWTWuWTWuWWu              |
| Index | Tie string            |
|-------|-----------------------|
| R-671 | WWWuTWuTWuTWuWu       |
| R-672 | WWWuTWuTWuTWuWu       |
| R-673 | WWWuTWuTWuTWuWu       |
| R-674 | WWWuTWuTWuTWuWu       |
| R-675 | WWWuTWuTWuTWuWu       |
| R-676 | WWWuTWuTTuWuWuWu      |
| R-677 | WWWuTWuTTuWuWuWu      |
| R-678 | WWWuTWuTTuWuWuWu      |
| R-679 | WWWuTWuTTuWuWuWu      |
| R-680 | WWWuTWuTTuWuWuWu      |
| R-681 | WWWuTWuTTuWuWuWu      |
| R-682 | WWWuTWuTTuWuWuWu      |
Next, we list the knots that tuck from the center. Again, with all optional single depth front tuck sites marked with a lowercase u. All these tie knots will have the thin blade sitting on top of the broad blade, which for most choices will be an unusual look, even for a modern tie knot. By comparing entries we may see that the 85 first knots in this enumeration are – with medial tucks ignored and with some permutation of assigned indices – the same as the 85 tie knots enumerated by Fink and Mao [2].

| Index | Tie string | Index | Tie string | Index | Tie string |
|-------|------------|-------|------------|-------|------------|
| C-1   | WWU        | C-41  | WWuWTWWU   | C-81  | WWuWTWWuWTW |
| C-2   | WTTU       | C-42  | WWuWTWWu   | C-82  | WWuWTWWuWTW |
| C-3   | TTuTTU     | C-43  | TTuTTuTTuWTW | C-83  | WWuWTWWuWTW|
| C-4   | WTWuU      | C-44  | TTuTTuWTWWu | C-84  | WWuWTWWuWTW |
| C-5   | TWWuU      | C-45  | TTuWTWTuuT | C-85  | WWuWTWWuWTW |
| C-6   | TTTuWWu    | C-46  | TTuWTWTuuT | C-86  | WWuWTWWuWTW |
| C-7   | TWWuTTu    | C-47  | TTuWTWTuuT | C-87  | WWuWTWWuWTW |
| C-8   | WWTU       | C-48  | TTuWTWTuuT | C-88  | WWuWTWWuWTW |
| C-9   | WWTUU      | C-49  | TTuWTWTuuT | C-89  | WWuWTWWuWTW |
| C-10  | WWuWTWWu   | C-50  | WTTuWTWTuuT | C-90  | WWuWTWWuWTW |
| C-11  | TTuWTWTuuT | C-51  | WTTuWTWTuuT | C-91  | WWuWTWWuWTW |
| C-12  | TTuWTWTuuT | C-52  | WTTuWTWTuuT | C-92  | WWuWTWWuWTW |
| C-13  | WTTuWTWTuuT | C-53 | WTTuWTWTuuT | C-93  | WWuWTWWuWTW |
| C-14  | TWTTuWTWTuuT | C-54 | WTTuWTWTuuT | C-94  | WWuWTWWuWTW |
| C-15  | TTTuWTWTuuT | C-55 | WTTuWTWTuuT | C-95  | WWuWTWWuWTW |
| C-16  | WWTWTuuT   | C-56  | WTTuWTWTuuT | C-96  | WWuWTWWuWTW |
| C-17  | WWTWTuuT   | C-57  | WTTuWTWTuuT | C-97  | WWuWTWWuWTW |
| C-18  | WWTWTuuT   | C-58  | WTTwWTWTuuT | C-98  | WWuWTWWuWTW |
| C-19  | WWTWTuuT   | C-59  | WTTwWTWTuuT | C-99  | WWuWTWWuWTW |
| C-20  | WWTWTuuT   | C-60  | WTTwWTWTuuT | C-100 | WWuWTWWuWTW |
| C-21  | WWTWTuuT   | C-61  | WTTwWTWTuuT | C-101 | WWuWTWWuWTW |
| C-22  | WWTWTuuT   | C-62  | WTTwWTWTuuT | C-102 | WWuWTWWuWTW |
| C-23  | WWTWTuuT   | C-63  | WTTwWTWTuuT | C-103 | WWuWTWWuWTW |
| C-24  | WWTWTuuT   | C-64  | WTTwWTWTuuT | C-104 | WWuWTWWuWTW |
| C-25  | WWTWTuuT   | C-65  | WTTwWTWTuuT | C-105 | WWuWTWWuWTW |
| C-26  | WWTWTuuT   | C-66  | WTTwWTWTuuT | C-106 | WWuWTWWuWTW |
| C-27  | WWTWTuuT   | C-67  | WTTwWTWTuuT | C-107 | WWuWTWWuWTW |
| C-28  | WWTWTuuT   | C-68  | WTTwWTWTuuT | C-108 | WWuWTWWuWTW |
| C-29  | WWTWTuuT   | C-69  | WTTwWTWTuuT | C-109 | WWuWTWWuWTW |
| C-30  | WWTWTuuT   | C-70  | WTTwWTWTuuT | C-110 | WWuWTWWuWTW |
| C-31  | WWTWTuuT   | C-71  | WTTwWTWTuuT | C-111 | WWuWTWWuWTW |
| C-32  | WWTWTuuT   | C-72  | WTTwWTWTuuT | C-112 | WWuWTWWuWTW |
| C-33  | WWTWTuuT   | C-73  | WTTwWTWTuuT | C-113 | WWuWTWWuWTW |
| C-34  | WWTWTuuT   | C-74  | WTTwWTWTuuT | C-114 | WWuWTWWuWTW |
| C-35  | WWTWTuuT   | C-75  | WTTwWTWTuuT | C-115 | WWuWTWWuWTW |
| C-36  | WWTWTuuT   | C-76  | WTTwWTWTuuT | C-116 | WWuWTWWuWTW |
| C-37  | WWTWTuuT   | C-77  | WTTwWTWTuuT | C-117 | WWuWTWWuWTW |
| C-38  | WWTWTuuT   | C-78  | WTTwWTWTuuT | C-118 | WWuWTWWuWTW |
| C-39  | WWTWTuuT   | C-79  | WTTwWTWTuuT | C-119 | WWuWTWWuWTW |
| C-40  | WWTWTuuT   | C-80  | WTTwWTWTuuT | C-120 | WWuWTWWuWTW |
| Index  | Tie string         | Index  | Tie string         | Index  | Tie string         |
|--------|--------------------|--------|--------------------|--------|--------------------|
| C-121  | TWuTuTWuTuTuTu    | C-171  | TTuTTuTTuTTuTTu   | C-221  | TWuTuTWuTuTuTu    |
| C-122  | TWuTuTuTuTuTuTu   | C-172  | TTuTTuTuTuTWuTuTu | C-222  | TWuTuTWuTuTuTu    |
| C-123  | WWuTuTuTuTuTuTu   | C-173  | TTuTTuTuTuTuTuTu  | C-223  | TWuTuTWuTuTuTuTu  |
| C-124  | WWuTuTuTuTuTuTu   | C-174  | TTuTTuTuTuTuTuTu  | C-224  | TWuTuTWuTuTuTuTu  |
| C-125  | WWuTuTuTuTuTuTu   | C-175  | TTuTTuTuTuTuTuTu  | C-225  | TWuTuTWuTuTuTuTu  |
| C-126  | WWuTuTuTuTuTuTu   | C-176  | TTuTTuTuTuTuTuTu  | C-226  | TWuTuTWuTuTuTuTu  |
| C-127  | WWuTuTuTuTuTuTu   | C-177  | TTuTTuTuTuTuTuTu  | C-227  | TWuTuTWuTuTuTuTu  |
| C-128  | WWuTuTuTuTuTuTu   | C-178  | TTuTTuTuTuTuTuTu  | C-228  | TWuTuTWuTuTuTuTu  |
| C-129  | WWuTuTuTuTuTuTu   | C-179  | TTuTTuTuTuTuTuTu  | C-229  | WWuTuTuTuTuTuTuTu |
| C-130  | WWuTuTuTuTuTuTu   | C-180  | TTuTTuTuTuTuTuTu  | C-230  | WWuTuTuTuTuTuTuTu |
| C-131  | WWuTuTuTuTuTuTu   | C-181  | TTuTTuTuTuTuTuTu  | C-231  | WWuTuTuTuTuTuTuTu |
| C-132  | WWuTuTuTuTuTuTu   | C-182  | TTuTTuTuTuTuTuTu  | C-232  | WWuTuTuTuTuTuTuTu |
| C-133  | WWuTuTuTuTuTuTu   | C-183  | TTuTTuTuTuTuTuTu  | C-233  | WWuTuTuTuTuTuTuTu |
| C-134  | WWuTuTuTuTuTuTu   | C-184  | TTuTTuTuTuTuTuTu  | C-234  | WWuTuTuTuTuTuTuTu |
| C-135  | WWuTuTuTuTuTuTu   | C-185  | TTuTTuTuTuTuTuTu  | C-235  | WWuTuTuTuTuTuTuTu |
| C-136  | WWuTuTuTuTuTuTu   | C-186  | TTuTTuTuTuTuTuTu  | C-236  | TTuTTuTuTuTuTuTuT |
| C-137  | WWuTuTuTuTuTuTu   | C-187  | TTuTTuTuTuTuTuTu  | C-237  | TTuTTuTuTuTuTuTuT |
| C-138  | WWuTuTuTuTuTuTu   | C-188  | TTuTTuTuTuTuTuTu  | C-238  | TTuTTuTuTuTuTuTuT |
| C-139  | WWuTuTuTuTuTuTu   | C-189  | TTuTTuTuTuTuTuTu  | C-239  | TTuTTuTuTuTuTuTuT |
| C-140  | WWuTuTuTuTuTuTu   | C-190  | TTuTTuTuTuTuTuTu  | C-240  | TTuTTuTuTuTuTuTuT |
| C-141  | WWuTuTuTuTuTuTu   | C-191  | TTuTTuTuTuTuTuTu  | C-241  | TTuTTuTuTuTuTuTuT |
| C-142  | WWuTuTuTuTuTuTu   | C-192  | TTuTTuTuTuTuTuTu  | C-242  | TTuTTuTuTuTuTuTuT |
| C-143  | WWuTuTuTuTuTuTu   | C-193  | TTuTTuTuTuTuTuTu  | C-243  | TTuTTuTuTuTuTuTuT |
| C-144  | WWuTuTuTuTuTuTu   | C-194  | TTuTTuTuTuTuTuTu  | C-244  | TTuTTuTuTuTuTuTuT |
| C-145  | WWuTuTuTuTuTuTu   | C-195  | TTuTTuTuTuTuTuTu  | C-245  | TTuTTuTuTuTuTuTuT |
| C-146  | WWuTuTuTuTuTuTu   | C-196  | TTuTTuTuTuTuTuTu  | C-246  | TTuTTuTuTuTuTuTuT |
| C-147  | WWuTuTuTuTuTuTu   | C-197  | TTuTTuTuTuTuTuTu  | C-247  | TTuTTuTuTuTuTuTuT |
| C-148  | WWuTuTuTuTuTuTu   | C-198  | TTuTTuTuTuTuTuTu  | C-248  | TTuTTuTuTuTuTuTuT |
| C-149  | WWuTuTuTuTuTuTu   | C-199  | TTuTTuTuTuTuTuTu  | C-249  | TTuTTuTuTuTuTuTuT |
| C-150  | WWuTuTuTuTuTuTu   | C-200  | TTuTTuTuTuTuTuTu  | C-250  | TTuTTuTuTuTuTuTuT |
| C-151  | WWuTuTuTuTuTuTu   | C-201  | TTuTTuTuTuTuTuTu  | C-251  | TTuTTuTuTuTuTuTuT |
| C-152  | WWuTuTuTuTuTuTu   | C-202  | TTuTTuTuTuTuTuTu  | C-252  | TTuTTuTuTuTuTuTuT |
| C-153  | WWuTuTuTuTuTuTu   | C-203  | TTuTTuTuTuTuTuTu  | C-253  | TTuTTuTuTuTuTuTuT |
| C-154  | WWuTuTuTuTuTuTu   | C-204  | TTuTTuTuTuTuTuTu  | C-254  | TTuTTuTuTuTuTuTuT |
| C-155  | WWuTuTuTuTuTuTu   | C-205  | TTuTTuTuTuTuTuTu  | C-255  | TTuTTuTuTuTuTuTuT |
| C-156  | WWuTuTuTuTuTuTu   | C-206  | TTuTTuTuTuTuTuTu  | C-256  | TTuTTuTuTuTuTuTuT |
| C-157  | WWuTuTuTuTuTuTu   | C-207  | TTuTTuTuTuTuTuTu  | C-257  | TTuTTuTuTuTuTuTuT |
| C-158  | WWuTuTuTuTuTuTu   | C-208  | TTuTTuTuTuTuTuTu  | C-258  | TTuTTuTuTuTuTuTuT |
| C-159  | WWuTuTuTuTuTuTu   | C-209  | TTuTTuTuTuTuTuTu  | C-259  | TTuTTuTuTuTuTuTuT |
| C-160  | WWuTuTuTuTuTuTu   | C-210  | TTuTTuTuTuTuTuTu  | C-260  | TTuTTuTuTuTuTuTuT |
| C-161  | WWuTuTuTuTuTuTu   | C-211  | TTuTTuTuTuTuTuTu  | C-261  | TTuTTuTuTuTuTuTuT |
| C-162  | WWuTuTuTuTuTuTu   | C-212  | TTuTTuTuTuTuTuTu  | C-262  | TTuTTuTuTuTuTuTuT |
| C-163  | WWuTuTuTuTuTuTu   | C-213  | TTuTTuTuTuTuTuTu  | C-263  | TTuTTuTuTuTuTuTuT |
| C-164  | WWuTuTuTuTuTuTu   | C-214  | TTuTTuTuTuTuTuTu  | C-264  | TTuTTuTuTuTuTuTuT |
| C-165  | WWuTuTuTuTuTuTu   | C-215  | TTuTTuTuTuTuTuTu  | C-265  | TTuTTuTuTuTuTuTuT |
| C-166  | WWuTuTuTuTuTuTu   | C-216  | TTuTTuTuTuTuTuTu  | C-266  | TTuTTuTuTuTuTuTuT |
| C-167  | WWuTuTuTuTuTuTu   | C-217  | TTuTTuTuTuTuTuTu  | C-267  | TTuTTuTuTuTuTuTuT |
| C-168  | WWuTuTuTuTuTuTu   | C-218  | TTuTTuTuTuTuTuTu  | C-268  | TTuTTuTuTuTuTuTuT |
| C-169  | WWuTuTuTuTuTuTu   | C-219  | TTuTTuTuTuTuTuTu  | C-269  | TTuTTuTuTuTuTuTuT |
| C-170  | WWuTuTuTuTuTuTu   | C-220  | TTuTTuTuTuTuTuTu  | C-270  | TTuTTuTuTuTuTuTuT |
| Index | Tie string            | Index | Tie string            |
|-------|----------------------|-------|----------------------|
| C-271 | WTWWuTTuTWuWWu       | C-321 | WWuTWTTuTTuWWu       |
| C-272 | WTWWuWTTuWWuWu       | C-322 | WWuTWWTuTuWWuTTu     |
| C-273 | WTWWuWTWWuTTu        | C-323 | WWuTWWTuTTuWWu       |
| C-274 | WTWWuWTWTuWWuWu      | C-324 | WWuTWWTuWWuTTu       |
| C-275 | WTWWuTWWWuTTu        | C-325 | WWuTWWTuWWuTTu       |
| C-276 | WTWWuWWuWTWTu        | C-326 | WWuWTWWuTTuTWTTu     |
| C-277 | WTWWuWuWWTuTTu       | C-327 | WWuWWuTWTuTuWWuTu    |
| C-278 | TWTTuWTWWuWuWu       | C-328 | WWuWWuWTuTTuWWuTTu   |
| C-279 | TWTTuTWuWWuWu       | C-329 | WWuWWuWTuTTuWWuTTu   |
| C-280 | TWTTuWuWTWTu       | C-330 | WWuWWuWTuTTuWWuTTu   |
| C-281 | TWTTuWuTWWWuWu       | C-331 | WWuWWuTWuWWuTTuTTu   |
| C-282 | TWWTWTuWuWWuWu       | C-332 | WWuWWuTWuTTuWWuTTu   |
| C-283 | TWWTWTWTuWWu       | C-333 | WWuWTuWWuWTuTTuTTu   |
| C-284 | TWWTWTWTuWu       | C-334 | WTWWuWTuWWuWTuWWu    |
| C-285 | TWWTWTWTuWWu       | C-335 | TWWTuWTuWWuWTuWWu    |
| C-286 | TWWTWTWTuWWu       | C-336 | WTWTuWTuWWuWTuWWu    |
| C-287 | TWWTWTuWTuWWu       | C-337 | WTWTuWTuWWuWTuWWu    |
| C-288 | TWWTWTuWTuWWu       | C-338 | WTWTuWTuWWuWTuWWu    |
| C-289 | TWWTWTuWTuWWu       | C-339 | WTWTuWTuWWuWTuWWu    |
| C-290 | TWWTWTWTuWWu       | C-340 | WTWTuWTuWWuWTuWWu    |
| C-291 | TWWTWTWTuWWu       | C-341 | WTWTuWTuWWuWTuWWu    |
| C-292 | TWWTWTWTuWWu       | C-342 | WTWTuWTuWWuWTuWWu    |
| C-293 | TWWTWTWTuWWu       | C-343 | WTWTuWTuWWuWTuWWu    |
| C-294 | TWWTWTuWTuWWu       | C-344 | WTWTuWTuWWuWTuWWu    |
| C-295 | TWWTWTuWTuWWu       | C-345 | WTWTuWTuWWuWTuWWu    |
| C-296 | TWWTWTuWTuWWu       | C-346 | WTWTuWTuWWuWTuWWu    |
| C-297 | TWWTWTuWTuWWu       | C-347 | WTWTuWTuWWuWTuWWu    |
| C-298 | TWWTWTuWTuWWu       | C-348 | WTWTuWTuWWuWTuWWu    |
| C-299 | TWWTWTWTuWWu       | C-349 | WTWTuWTuWWuWTuWWu    |
| C-300 | TWWTWTWTuWWu       | C-350 | WTWTuWTuWWuWTuWWu    |
| C-301 | TWWTWTWTuWWu       | C-351 | WTWTuWTuWWuWTuWWu    |
| C-302 | TWWTWTWTuWWu       | C-352 | WTWTuWTuWWuWTuWWu    |
| C-303 | TWWTWTWTuWWu       | C-353 | WTWTuWTuWWuWTuWWu    |
| C-304 | WWuWTuTTuWWuWWu     | C-354 | WWuWTuWWuWTuTTuTTu   |
| C-305 | WWuWTuWTWTuWWu     | C-355 | WWuWTuWWuWTuTTuTTu   |
| C-306 | WWuWTuWTWTuWWu     | C-356 | WWuWTuWWuWTuTTuTTu   |
| C-307 | WWuWTuWTWTuWWu     | C-357 | WWuWTuWWuWTuTTuTTu   |
| C-308 | WWuWTuWTWTuWWu     | C-358 | WWuWTuWWuWTuTTuTTu   |
| C-309 | WWuWTuWTWTuWWu     | C-359 | WWuWTuWWuWTuTTuTTu   |
| C-310 | WWuWTuWTWTuWWu     | C-360 | WWuWTuWWuWTuTTuTTu   |
| C-311 | WWuWTWTWTuWWu     | C-361 | WWuWTuWWuWTuTTuTTu   |
| C-312 | WWuWTWTWTuWWu     | C-362 | WWuWTWTuWTuTTuTTu   |
| C-313 | WWuWTWTWTuWWu     | C-363 | WWuWTWTuWTuTTuTTu   |
| C-314 | WWuWTWTWTuWWu     | C-364 | WWuWTWTuWTuTTuTTu   |
| C-315 | WWuWTWTWTuWWu     | C-365 | WWuWTWTuWTuTTuTTu   |
| C-316 | WWuWTWTWTuWWu     | C-366 | WWuWTWTuWTuTTuTTu   |
| C-317 | WWuWTWTWTuWWu     | C-367 | WWuWTWTuWTuTTuTTu   |
| C-318 | WWuWTWTWTuWWu     | C-368 | WWuWTWTuWTuTTuTTu   |
| C-319 | WWuWTWTWTuWWu     | C-369 | WWuWTWTuWTuTTuTTu   |
| C-320 | WWuWTWTWTuWWu     | C-370 | WWuWTWTuWTuTTuTTu   |
| Index  | Tie string                      | Index  | Tie string                      |
|--------|---------------------------------|--------|---------------------------------|
| C-371  | WTTuTuTTuWTWuW       | C-421  | TWTWTTuTuTWWuW       |
| C-372  | WTTuTuTTuTTuTW       | C-422  | TWTWTTuTuTWWuW       |
| C-373  | WTTuTuTTuWTTuTTu       | C-423  | TWTWTTuTuTWWuTTu       |
| C-374  | WTTuTuTTuWTTuTTu       | C-424  | TWTWTTuTuTWWuTTu       |
| C-375  | WTTuWTTuTTuTTu       | C-425  | TWTWTTuTuTWWuTTu       |
| C-376  | WTTuWTTuTTuTTu       | C-426  | TWTWTTuTuTWWuTTu       |
| C-377  | WTTuTTuTuTTuTTu       | C-427  | TWTWTTuTuTWWuTTu       |
| C-378  | WTTuTTuTuTTuTTu       | C-428  | TWTWuTuTTuTTuW       |
| C-379  | WTTuTuWTWuWuW       | C-429  | TWTWuTuTTuWuTTu       |
| C-380  | WTTuTuTuWwwuWuW       | C-430  | TWTWuTuWWTWu       |
| C-381  | WTTuTuTuWwwuWuW       | C-431  | TWTWuTuWWTWu       |
| C-382  | WTTuTuWwwuWuWuW       | C-432  | TWTWuTuWWTWu       |
| C-383  | WTTuWTTuWwwuWuW       | C-433  | TWTWuTuWWTWu       |
| C-384  | WTTuWTTuWwwuWuW       | C-434  | TWTWuTuWWTWu       |
| C-385  | WTTuWTTTWwwuWuW       | C-435  | WTTuTuTuWwwuWuW       |
| C-386  | WTTuWTTTWwwuWuW       | C-436  | WTTuTuTuWwwuWuW       |
| C-387  | WTTuWTTTWwwuWuW       | C-437  | WTTuTuTuWwwuWuW       |
| C-388  | WTTuWTWwwuWuWuW       | C-438  | WTTuTuTuWwwuWuW       |
| C-389  | WTTuWTWwwuWuWuW       | C-439  | WTTuTuTuWwwuWuW       |
| C-390  | WTTuWTWwwuWuWuW       | C-440  | WTTuTuTuWwwuWuW       |
| C-391  | WTTuWTWwwuWuWuW       | C-441  | WTTuTuTuWwwuWuW       |
| C-392  | WTTuWTWwwuWuWuW       | C-442  | WTTuTuTuWwwuWuW       |
| C-393  | WTTuWTWwwuWuWuW       | C-443  | WTTuTuTuWwwuWuW       |
| C-394  | WTTuWTWwwuWuWuW       | C-444  | WTTuTuTuWwwuWuW       |
| C-395  | WTTuWTWwwuWuWuW       | C-445  | WTTuTuTuWwwuWuW       |
| C-396  | WTTuWTWwwuWuWuW       | C-446  | WTTuTuTuWwwuWuW       |
| C-397  | WTTuWTWwwuWuWuW       | C-447  | WTTuTuTuWwwuWuW       |
| C-398  | WTTuWTWwwuWuWuW       | C-448  | WTTuTuTuWwwuWuW       |
| C-399  | WTTuWTWwwuWuWuW       | C-449  | WTTuTuTuWwwuWuW       |
| C-400  | WTTuWTWwwuWuWuW       | C-450  | WTTuTuTuWwwuWuW       |
| C-401  | WTTuWTWwwuWuWuW       | C-451  | WTTuTuTuWwwuWuW       |
| C-402  | WTTuWTWwwuWuWuW       | C-452  | WTTuTuTuWwwuWuW       |
| C-403  | WTTuWTWwwuWuWuW       | C-453  | WTTuTuTuWwwuWuW       |
| C-404  | WTTuWTWwwuWuWuW       | C-454  | WTTuTuTuWwwuWuW       |
| C-405  | WTTuWTWwwuWuWuW       | C-455  | WTTuTuTuWwwuWuW       |
| C-406  | WTTuWTWwwuWuWuW       | C-456  | WTTuTuTuWwwuWuW       |
| C-407  | WTTuWTWwwuWuWuW       | C-457  | WTTuTuTuWwwuWuW       |
| C-408  | WTTuWTWwwuWuWuW       | C-458  | WTTuTuTuWwwuWuW       |
| C-409  | WTTuWTWwwuWuWuW       | C-459  | WTTuTuTuWwwuWuW       |
| C-410  | WTTuWTWwwuWuWuW       | C-460  | WTTuTuTuWwwuWuW       |
| C-411  | WTTuWTWwwuWuWuW       | C-461  | WTTuTuTuWwwuWuW       |
| C-412  | WTTuWTWwwuWuWuW       | C-462  | WTTuTuTuWwwuWuW       |
| C-413  | WTTuWTWwwuWuWuW       | C-463  | WTTuTuTuWwwuWuW       |
| C-414  | WTTuWTWwwuWuWuW       | C-464  | WTTuTuTuWwwuWuW       |
| C-415  | WTTuWTWwwuWuWuW       | C-465  | WTTuTuTuWwwuWuW       |
| C-416  | WTTuWTWwwuWuWuW       | C-466  | WTTuTuTuWwwuWuW       |
| C-417  | WTTuWTWwwuWuWuW       | C-467  | WTTuTuTuWwwuWuW       |
| C-418  | WTTuWTWwwuWuWuW       | C-468  | WTTuTuTuWwwuWuW       |
| C-419  | WTTuWTWwwuWuWuW       | C-469  | WTTuTuTuWwwuWuW       |
| C-420  | WTTuWTWwwuWuWuW       | C-470  | WTTuTuTuWwwuWuW       |
| Index | Tie string          | Index | Tie string          |
|-------|---------------------|-------|---------------------|
| C-471 | TWuTTuWuWuWTuu      | C-521 | WTTuTTuTWuWTuu      |
| C-472 | TWuTTuWuWuTWuu      | C-522 | WTTuTTuTWuTWuu      |
| C-473 | TWuWTTuTTuTuWu      | C-523 | WTTuTTuWTTuTuWu     |
| C-474 | TWuWTTuTTuTuWu      | C-524 | WTTuTTuWTuuWuTTu    |
| C-475 | TWuWTuWTuWTuWu      | C-525 | WTTuTTuTWuTuWu      |
| C-476 | TWuWTuWTuWTuWu      | C-526 | WTTuTTuTWuTuWu      |
| C-477 | TWuWTTuWTuWTuWu     | C-527 | WTTuTTuWuWuWTuu     |
| C-478 | TWuWTTuWTuWTuWu     | C-528 | WTTuTTuWuWuWTuu     |
| C-479 | TWuWTuWTuWTuWu      | C-529 | WTTuTTuTuTuWu      |
| C-480 | TWuWTuWTuWTuWu      | C-530 | WTTuTTuWuWTuu      |
| C-481 | TWuWTuWTuWTuWu      | C-531 | WTTuTTuWTuu      |
| C-482 | TWuWTuWTuWTuWu      | C-532 | WTTuTTuWTuu      |
| C-483 | TWuWTuWTuWTuWu      | C-533 | WTTuTTWuWTuu      |
| C-484 | TWuWTuWTuWTuWu      | C-534 | WTTuTTWTuu      |
| C-485 | TWuWTuWTuWTuWu      | C-535 | WTTuTTWuWTuu      |
| C-486 | TWuWTuWTuWTuWu      | C-536 | WTTuTTuWTuu      |
| C-487 | TWuWTuWTuWTuWu      | C-537 | WTTuTTuWTuu      |
| C-488 | TWuWTuWTuWTuWu      | C-538 | WTTuTTuWTuu      |
| C-489 | TWuWTuWTuWTuWu      | C-539 | WTTuTTuWTuu      |
| C-490 | TWuWTuWTuWTuWu      | C-540 | WTTuTTuWTuu      |
| C-491 | WTTuTTuTuWuWuTu      | C-541 | WTTuTTuWTuu      |
| C-492 | WTTuTTuTuWTuWu      | C-542 | WTTuTTuWTuu      |
| C-493 | WTTuTTuTuWTuWu      | C-543 | WTTuTTuWTuu      |
| C-494 | WTTuTTuTuWTuWu      | C-544 | WTTuTTuWTuu      |
| C-495 | WTTuTTuTuWTuWu      | C-545 | WTTuTTuWTuu      |
| C-496 | WTTuTTuTuWTuWu      | C-546 | WTTuTTuWTuu      |
| C-497 | WTTuTTuTuWTuWu      | C-547 | WTTuTTuWTuu      |
| C-498 | WTTuTTuTuWTuWu      | C-548 | WTTuTTuWTuu      |
| C-499 | WTTuTTuTuWTuWu      | C-549 | WTTuTTuWTuu      |
| C-500 | WTTuTTuTuWTuWu      | C-550 | WTTuTTuWTuu      |
| C-501 | WTTuTTuTuWTuWu      | C-551 | WTTuTTuWTuu      |
| C-502 | WTTuTTuTuWTuWu      | C-552 | WTTuTTuWTuu      |
| C-503 | WTTuTTuTuWTuWu      | C-553 | WTTuTTuWTuu      |
| C-504 | WTTuTTuTuWTuWu      | C-554 | WTTuTTuWTuu      |
| C-505 | WTTuTTuTuWTuWu      | C-555 | WTTuTTuWTuu      |
| C-506 | WTTuTTuTuWTuWu      | C-556 | WTTuTTuWTuu      |
| C-507 | WTTuTTuTuWTuWu      | C-557 | WTTuTTuWTuu      |
| C-508 | WTTuTTuTuWTuWu      | C-558 | WTTuTTuWTuu      |
| C-509 | WTTuTTuTuWTuWu      | C-559 | WTTuTTuWTuu      |
| C-510 | WTTuTTuTuWTuWu      | C-560 | WTTuTTuWTuu      |
| C-511 | WTTuTTuTuWTuWu      | C-561 | WTTuTTuWTuu      |
| C-512 | WTTuTTuTuWTuWu      | C-562 | WTTuTTuWTuu      |
| C-513 | WTTuTTuTuWTuWu      | C-563 | WTTuTTuWTuu      |
| C-514 | WTTuTTuTuWTuWu      | C-564 | WTTuTTuWTuu      |
| C-515 | WTTuTTuTuWTuWu      | C-565 | WTTuTTuWTuu      |
| C-516 | WTTuTTuTuWTuWu      | C-566 | WTTuTTuWTuu      |
| C-517 | WTTuTTuTuWTuWu      | C-567 | WTTuTTuWTuu      |
| C-518 | WTTuTTuTuWTuWu      | C-568 | WTTuTTuWTuu      |
| C-519 | WTTuTTuTuWTuWu      | C-569 | WTTuTTuWTuu      |
| C-520 | WTTuTTuTuWTuWu      | C-570 | WTTuTTuWTuu      |
| Index | Tie string          | Index | Tie string          |
|-------|---------------------|-------|---------------------|
| C-571 | WWWWuTTTuTTuTU     | C-572 | WWWWuTTTuTTuTU     |
| C-573 | WWWuTTuTTuTTuWWU   | C-574 | WWWuTTuTTuWWuTTU   |
| C-575 | WWWuTTuWTWTTU      | C-576 | WWWuTTuWTWTTU      |
| C-577 | WWWuTTuTWTTT       | C-578 | WWWuTTuTWTTTU      |
| C-579 | WWWuTTuWuTTuTTu    | C-580 | WWWuWTTTuWTTU      |
| C-581 | WWWuWTTTuTTuTU     | C-582 | WWWuWTTTuTTuTU     |
| C-583 | WWWuWTWTuTTuTU     | C-584 | WWWuWTWTuTTuTU     |
| C-585 | WWWuWTWTuTTuTTU    | C-586 | WWWuWTWTuTTuTTU    |
| C-587 | WWWuWTTWTTuTTU     | C-588 | WWWuWuWWuTTuTTu    |
| C-589 | TTTuWuWuWuWuWU      | C-590 | TWWWTuWuWuWU       |
| C-591 | TWTWuWTWuWuWU       | C-592 | TWTWuWTWuWuWU       |
| C-593 | TWTWuWTWuWuWU       | C-594 | TWTWuWTWuWuWU       |
| C-595 | TWTWuWTWuWuWU       | C-596 | TWTWuWTWuWuWU       |
| C-597 | TWTWuWTWuWuWU       | C-598 | TWTWuWTWuWuWU       |
| C-599 | TWTWuWTWuWuWU       | C-600 | TWTWuWTWuWuWU       |
| C-601 | TWTWuWTWuWuWU       | C-602 | TWTWuWTWuWuWU       |
| C-603 | TWTWuWTWuWuWU       | C-604 | TWTWuWTWuWuWU       |
| C-605 | TWTWuWTWuWuWU       | C-606 | TWTWuWTWuWuWU       |
| C-607 | TWTWuWTWuWuWU       | C-608 | TWTWuWTWuWuWU       |
| C-609 | TWTWuWTWuWuWU       | C-610 | TWTWuWTWuWuWU       |
| C-611 | TWTWuWTWuWuWU       | C-612 | TWTWuWTWuWuWU       |
| C-613 | TWTWuWTWuWuWU       | C-614 | TWTWuWTWuWuWU       |
| C-615 | TWTWuWTWuWuWU       | C-616 | TWTWuWTWuWuWuWU     |
| C-617 | TWTWuWTWuWuWuWU     | C-618 | TWTWuWTWuWuWuWU     |
| C-619 | TWTWuWTWuWuWuWU     | C-620 | TWTWuWTWuWuWuWU     |
| Index | Tie string               |
|-------|--------------------------|
| C-671 | WWWuTWTTWWU             |
| C-672 | WWWuTWWTuWWU            |
| C-673 | WWWuTWWTuWWuTTU         |
| C-674 | WWWuWTWTuWWTWU          |
| C-675 | WWWuWTWTuTWWU           |
| C-676 | WWWuWTWTuTTuWWU         |
| C-677 | WWWuWTWTuTTuWWTWU       |
| C-678 | WWWuWTWTuTTuTWTTU       |
| C-679 | WWWuWTWTuTWWuTTU        |
| C-680 | WWWuWTWTuTTuWWTWU       |
| C-681 | WWWuWTWTuTTuWWTWTU      |
| C-682 | WWWuWTWTuTTuWWTWTWU     |
Next, we include the Python code listing that we used to generate these enumerations. With some utility functions, the core of this code is in the combinatorics generated from the `itertools` package and in the mod 3 tests on final segments of strings.

```python
# ties.py
# (c) 2013 Mikael Vejdemo-Johansson
# Released as CC-BY

import itertools
from scipy.special import binom

""
Reproducing Fink & Mao is done by the following:

finkmao = classical(prototies(wtpairs(9)))
""

def wtpairs(maxN, mod3 = 1):
    return [(i, n-i) for n in range(maxN)
            for i in range(n+1) if (n-2*i) % 3 == mod3]

def prototies(wtps):
    ret = []
    for (n, k) in wtps:
        if k==0:
            ret.append([ 'W' ]*(n+k))
            continue
        poss = itertools.combinations(range(n+k), k)
        for pos in poss:
            tie = [ 'W' ]*(n+k)
            for p in pos:
                tie[p] = 'T'
            ret.append(tie)
    return ret

def classical(protos):
    return [p for p in protos if len(p)>=2 and p[-2] == p[-1]]

def ktuckfinals(protos, k):
    def wort(t):
        if t == 'W':
            return 1
        elif t == 'T':
            return -1
        else:
            return 0
    def wmint(tt):
        return sum(map(wort, tt)) % 3
    return [p for p in protos
```
if len(p) >= 2*k and wmint(p[-2*k:]) == ((-wort(p[-2*k])) % 3)]

"""
States & cyclic transitions:
W T
L (left) ^ v
C (center) ^ v
R (right) ^ v
"""

statevec = {'T': {'L': 'C', 'C': 'R', 'R': 'L'},
            'W': {'L': 'R', 'R': 'C', 'C': 'L'}}

def WTtoCLR(tie, startstate='L'):
    state = startstate
    ret = state
    for wt in tie:
        state = statevec[wt][state]
        ret += state
    return ret

def balance(tie):
    return len([i for i in range(1, len(tie)) if tie[i] != tie[i-1]])

def symmetry(tie, startstate='L'):
    clrtie = WTtoCLR(tie, startstate=startstate)
    return len([i for i in range(len(clrtie)) if clrtie[i]=='L']) - len([i for i in range(len(clrtie)) if clrtie[i]=='R'])

def CLRtoWT(tie):
    ret = []
    fr = tie[0]
    for st in tie[1:]:
        to = st
        if to == 'T':
            ret.append('U')
        elif statevec['W'][fr] == to:
            ret.append('W')
        elif statevec['T'][fr] == to:
            ret.append('T')
        fr = to
    return ret

def countWT(w, t):
    if w+t < 2:
        return 0
    return binom(w+t, t) - 2*binom(w+t-2, t-1)
def tucks(t):
    return ktucks(t)

def ktuck(t,k):
    """
    Tests whether the end of the winding string t is a valid site for
    a k-fold tuck
    """
    if len(t) < 2*k:
        return False
    ws = [c for c in t[-2*k:] if c == 'W']
    ts = [c for c in t[-2*k:] if c == 'T']
    diff = (len(ws)-len(ts)) % 3
    if t[-2*k] == 'W':
        return diff == 2
    elif t[-2*k] == 'T':
        return diff == 1
    return False

def ktucks(t,k=1):
    """
    Annotates all valid k-fold tuck sites in a winding string
    """
    revt = list(t)
    ret = []
    while len(revt) > 0:
        if ktuck(revt,k):
            ret.insert(0,'u')
            ret.insert(0,revt.pop())
    return ret

def printall(maxN,k=1,p=lambda t: True):
    """
    Prints out a TeX-able table of all tie knots with k-fold tuck
    sites marked. Defaults to k=1. Separated by final tuck direction.
    """
    for j,c in enumerate(['L','C','R']):
        ties = [ktucks(t,k) for t in prototypes(wtpairs(maxN,mod3=j)) if len(t) > 1 and p(t)]
        for i,t in enumerate(ties):
            print ('%s-%d' % (c,i+1)), '&', ''.join(t), '\\\

finkmao = ['LRCT',

```