

A Relatively Small Turing Machine Whose Behavior Is Independent of Set Theory

Adam Yedidia
MIT
adamy@mit.edu

Scott Aaronson
MIT
aaronson@csail.mit.edu

April 22, 2016

Abstract

Since the definition of the Busy Beaver function by Radó in 1962, an interesting open question has been what the smallest value of n for which $BB(n)$ is independent of ZFC set theory. Is this n approximately 10, or closer to 1,000,000, or is it even larger? In this paper, we show that it is at most 7,918 by presenting an explicit description of a 7,918-state Turing machine Z with 1 tape and a 2-symbol alphabet that cannot be proved to run forever in ZFC (even though it presumably does), assuming ZFC is consistent. The machine is based on work of Harvey Friedman on independent statements involving order-invariant graphs. In doing so, we give the first known upper bound on the highest provable Busy Beaver number in ZFC. We also present a 4,888-state Turing machine G that halts if and only if there is a counterexample to Goldbach's conjecture, and a 5,372-state Turing machine R that halts if and only if the Riemann hypothesis is false. To create G , R , and Z , we develop and use a higher-level language, Laconic, which is much more convenient than direct state manipulation.

1 Introduction

1.1 Background and Motivation

Zermelo-Fraenkel set theory with the axiom of choice, more commonly known as ZFC, is an axiomatic system invented in the twentieth which has since been used as the foundation of most of modern mathematics. It encodes arithmetic by describing natural numbers as increasing sets of sets.

Like any axiomatic system capable of encoding arithmetic, ZFC is constrained by Gödel's two incompleteness theorems. The first incompleteness theorem states that if ZFC is *consistent* (it never proves both a statement and its opposite), then ZFC cannot also be *complete* (able to prove every true statement). The second incompleteness theorem states that if ZFC is consistent, then ZFC cannot prove its own consistency. Because we have built modern mathematics on top of ZFC, we can reasonably be said to have assumed ZFC's consistency. This means that we must also believe that ZFC cannot prove its own consistency. This fact carries with it certain surprising conclusions.

In particular, consider a Turing machine Z that enumerates, one after the other, each of the provable statements in ZFC. To describe how such a machine might be constructed, Z could iterate over the axioms and inference rules of ZFC, applying each in every possible way to each conclusion

or pair of conclusions that had been reached so far. We might ask Z to halt if it ever reaches a contradiction; in other words, Z will halt if and only if it finds a proof of $0 = 1$. Because this machine will enumerate *every* provable statement in ZFC, it will run forever if and only if ZFC is consistent.

It follows that Z is a Turing machine for which the question of its behavior (whether or not it halts when run indefinitely) is equivalent to the consistency of ZFC.¹ Therefore, just as ZFC cannot prove its own consistency (assuming ZFC is consistent), ZFC also cannot prove that Z will run forever.

This is interesting because, while the undecidability of the halting problem tells us that there cannot exist an algorithmic method for determining whether an *arbitrary* Turing machine loops or halts, Z is an example of a *specific* Turing machine whose behavior cannot be proven one way or the other using the foundation of modern mathematics. Mathematicians and computer scientists think of themselves as being able to determine how a given algorithm will behave if given enough time to stare at it; despite this intuition, Z is a machine whose behavior we can never prove without assuming axioms more powerful than those generally assumed in modern mathematics.

1.2 Turing Machines

There are many slightly-different definitions of Turing machines. For example, some definitions allow the machine to have multiple tapes; others only allow it to have one; some allow an arbitrarily large alphabet, while others allow only two symbols, and so on. In most research regarding Turing machines, mathematicians don't concern themselves with which of these models to use, because any one can simulate the others. However, because this work is concerned with upper-bounding the exact number of states required to perform certain tasks, it's important to define the model precisely.

Formally, a k -state Turing machine is a 7-tuple $M = (Q, \Gamma, b, \Sigma, \delta, q_0, F)$, where:

Q is the set of k states $\{q_0, q_1, \dots, q_{k-2}, q_{k-1}\}$

$\Gamma = \{a, b\}$ is the set of *tape alphabet symbols*

a is the *blank symbol*

Σ is the set of *input symbols*

$\delta = Q \times \Gamma \rightarrow (Q \cup F) \times \Gamma \times \{L, R\}$ is the *transition function*

q_0 is the *start state*

$F = \{\text{ACCEPT}, \text{REJECT}, \text{ERROR}\}$ is the set of *halting transitions*.

A Turing machine's *states* make up the Turing machine's easily-accessible, finite memory. The Turing machine's state is initialized to q_0 .

The *tape alphabet symbols* correspond to the symbols that can be written on the Turing machine's infinite tape.

In this work, all Turing machines are run on the all- a input.

The *transition function* encodes the Turing machine's behavior. It takes two inputs: the current state of the Turing machine (an element of Q) and the symbol read off the tape (an element of Γ).

¹While we will talk about ZFC throughout this paper, rather than simple ZF set theory, this is simply a convention. For our purposes, the Axiom of Choice is irrelevant: the consistency of ZFC is equivalent to the consistency of simple ZF set theory, [11] and ZFC and ZF prove exactly the same arithmetical statements (which include, among other things, statements about whether Turing machines halt). [20]

It outputs three instructions: what state to enter (an element of Q), what symbol to write onto the tape (an element of Γ) and what direction to move the head in (an element of $\{L, R\}$). A transition function specifies the entire behavior of the Turing machine in all cases.

The *start state* is the state that the Turing machine is in at initialization.

A *halting transition* is a transition that causes the Turing machine to halt. While having three possible halting transitions is not necessary for our purposes, being able to differentiate between three different types of halting (ACCEPT, REJECT, and ERROR) is useful for testing.

1.3 The Busy Beaver Function

Consider the set of all Turing machines with k states, for some positive integer k . We call a Turing machine B a *k-state Busy Beaver* if when run on the empty tape as input, B halts, and also runs for at least as many steps before halting as all other halting k -state Turing machines. [19]

In other words, a Busy Beaver is a Turing machine that runs for at least as long as all other halting Turing machines with the same number of states. Another common definition for a Busy Beaver is a Turing machine that writes as many 1's on the tape as possible; because the number of 1's written is a somewhat arbitrary measure, it is not used in this work.

The *Busy Beaver function*, written $BB(k)$, equals the number of steps it takes for a k -state Busy Beaver to halt. The Busy Beaver function has many striking properties. To begin with, it is not *computable*; in other words, there does not exist an algorithm that takes k as input and returns $BB(k)$, for arbitrary values of k . This follows directly from the undecidability of the halting problem. Suppose an algorithm existed to compute the Busy Beaver function; then given a k -state Turing machine M as input, we could compute $BB(k)$ and run M for $BB(k)$ steps. If, after $BB(k)$ steps, M had not yet halted, we could safely conclude that M would never halt. Thus, we could solve the halting problem, which we know is impossible.

By the same argument, $BB(k)$ must grow faster than any computable function. (To check this, assume that some computable function $f(k)$ grows faster than $BB(k)$, and substitute $f(k)$ for $BB(k)$ in the rest of the proof.) In particular, the Busy Beaver grows even faster than (for instance) the Ackermann function, a well-known fast-growing function.

Because finding the value of $BB(k)$ for a given k requires so much work (one must fully explore the behavior of all k -state Turing machines), few explicit values of the Busy Beaver function are known. The known values are [3, 13]:

$$BB(1) = 1$$

$$BB(2) = 6$$

$$BB(3) = 21$$

$$BB(4) = 107$$

For $BB(5)$ and $BB(6)$, only lower bounds are known: $BB(5) \geq 47,176,870$, and $BB(6) \geq 7.4 \times 10^{36,534}$. Researchers have worked on pinning down the value of $BB(5)$ exactly, and some consider it to be possibly within reach. A summary of the current state of human knowledge about Busy Beaver values can be found at [16].

Another way to discuss the Busy Beaver sequence is to say that modern mathematics has established a *lower bound* of 4 on the highest provable Busy Beaver value. In this paper, we prove

the first known *upper bound* on the highest provable Busy Beaver value in ZFC; that is, we give a value of k , namely 7,918, such that the value of $BB(k)$ cannot be proven in ZFC.

Intuitively, one might expect that while no algorithm may exist to compute $BB(k)$ for *all* values of k , we could find the value of $BB(k)$ for any *specific* k using a procedure similar to the one we used to find the value of $BB(k)$ for $k \leq 4$. The reason this is not so is closely tied to the existence of a machine like the Gödelian machine Z , as described in Section 1.1. Suppose that Z has k states. Because Z 's behavior (whether it halts or loops) cannot be proven in ZFC, it follows that the value of $BB(k)$ also can't be proven in ZFC; if it could, then a proof would exist of Z 's behavior in ZFC. Such a proof would consist of a *computation history* for Z , which is an explicit step-by-step description of Z 's behavior for a certain number of steps. If Z halts, then a computation history leading up to Z 's halting would be the entire proof; if Z loops, then a computation history that takes $BB(k)$ steps, combined with a proof of the value of $BB(k)$, would constitute a proof that Z will run forever.

In this paper we construct a machine like Z , for which a proof that Z runs forever would imply that ZFC was consistent. In doing so, we give an explicit upper bound on the highest Busy Beaver value provable in ZFC assuming the consistency of a slightly stronger set theory. Our machine, which we shall refer to as Z hereafter, contains 7,918 states. Therefore, we will never be able to prove the value of $BB(7,918)$ without assuming more powerful axioms than those of ZFC. This upper bound is presumably very far from tight, but it is a first step.

Even to achieve a state count of 7,918, we will need three nontrivial ideas: Harvey Friedman's order-theoretic statements, *on-tape processing*, and *introspective encoding*. Without all three ideas, we found that the state count would be in the tens of thousands, hundreds of thousands, or even millions. We briefly introduce these ideas in the following subsection, and explore them in much greater detail in Section 8. The implementation of these ideas constitutes this paper's main technical contribution.

1.4 Parsimony

In most algorithmic study, efficiency is the primary concern. In designing Z , however, parsimony is the only thing that matters. One historical analogue is the practice of "code-golfing": a recreational pursuit adopted by some programmers in which the goal is to produce a piece of code in a given programming language, using as few characters as possible. Many examples of code-golfing can be found at [22]. The goal of designing a Turing machine with as few states as possible to accomplish a certain task, without concern for the machine's efficiency or space usage, can be thought of as code-golfing with a particularly low-level programming language.

Part of the charm of Turing machines is that they give us a "standard reference point" for measuring complexity, unencumbered by the details of more sophisticated programming languages. Also, with Turing machines, there can be no suspicion that we engineered a programming formalism just for the purpose of code-golfing, or for making the concepts we want artificially simple to describe. This is why we prefer Turing machines as a tool for measuring complexity; not because they are particularly special, but simply because they are so primitive that their specifics will interfere minimally with what we mean by an algorithm being "complicated."

In this paper, we use three ideas for generating parsimonious Turing machines: Harvey Friedman's mathematical statements, *on-tape processing*, and *introspective Turing machines*. The last of these ideas was proposed, under a different name and with some variations, by Ben-Amram and Petersen in 2002 [2]. These three ideas are explained in more detail in Subsections 3.1, 8.1, and 8.3,

respectively, but we summarize them very briefly here.

The first idea is simply to use the research done by Friedman into finding simple-to-express statements that are equivalent to the consistency of various axiomatic systems. In particular, we use a statement discovered by Friedman to be equivalent to the consistency of a set theory stronger than ZFC (and whose consistency, therefore, would imply the consistency of ZFC). [7]

The second idea, on-tape processing, is a way to encode high-level commands into a Turing machine parsimoniously. Instead of converting commands to groups of states directly, which incurs a multiplicative overhead based on how large these groups need to be, on-tape processing begins by writing the commands onto the tape, using as efficient an encoding as possible. Then, once the commands are on the tape, the commands are processed by a single group of states that understands how to interpret them.

The third idea, introspective Turing machines, is a way to write long strings onto the tape using as few states as possible. The idea is to encode information one of each state's transitions, instead of encoding information in each state's write field. This is advantageous because there are many choices for which state to point a transition to, but only two choices for which bit to write. Therefore, more information can be encoded in each state using this method.

1.5 Implementation Overview

To generate descriptions of Turing machines with nice mathematical properties entirely by hand is a daunting task. Rather than approach the problem directly, we created tools for generating parsimonious Turing machines while presenting an interface that is comfortably familiar to most programmers (and to us!).

We created two tools. At the top level is the Laconic programming language, whose syntax and capabilities are similar to those of most programming languages, such as Java or Python. Beneath it we created a lower-level language called Turing Machine Descriptor (TMD). TMD is quite unlike most programming languages, and is better thought of as a convenient way to describe a multi-tape, 3-symbol Turing machine plus a function stack. The style of multi-tape Turing machine used in TMD is the commonly used "one-tape-at-a-time" abstraction: only one tape at a time can be interacted with, for reading, writing, and moving the head. Laconic compiles down to a TMD program, and TMD compiles down to a description of a single-tape, 2-symbol Turing machine. This process is illustrated in Figure 1.

We recommend that programmers hoping to use our tools to generate their own encodings of mathematical statements or algorithms as Turing machines use Laconic. Laconic's interface is perfect for somebody hoping to write in a "traditional" language. On the other hand, if the programmer wishes to improve upon Laconic's compilation process, writing code directly in TMD is likely to be the better option.

2 Related Work

Gregory Chaitin raised the problem of proving a version of our result in his book *The Limits of Mathematics*. [6] He wrote:

I would like to have somebody program out Zermelo-Fraenkel set theory in my version of LISP, which is pretty close to normal LISP as far as this task is concerned, just to see how many bits of complexity mathematicians normally assume ... If you programmed

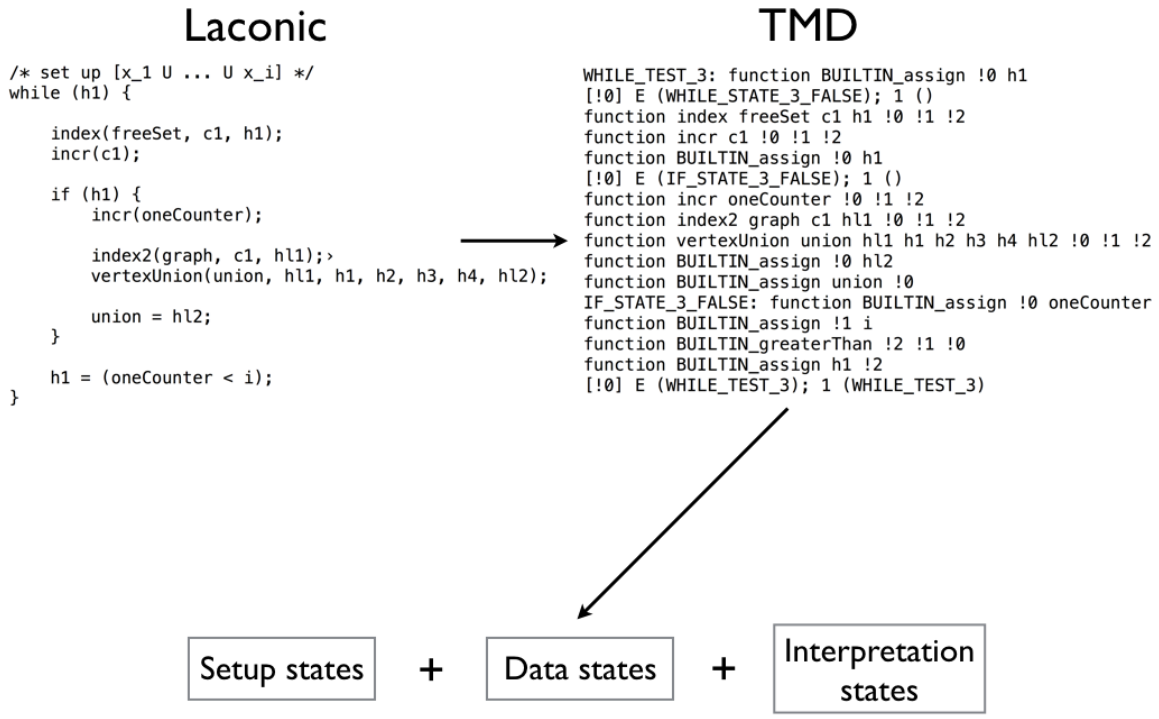


Figure 1: A visual overview of the compilation process.

ZF, you'd get a really sharp incompleteness result. It wouldn't say that you can get at most $H(ZF) + 15328$ bits of [Chaitin's halting probability] Ω , it would say, perhaps, at most 96000 bits! We'd have a much more definite incompleteness theorem.

We did not program ZF set theory in LISP, but we programmed it in an even simpler language—thereby answering Chaitin's call for an explicit number of bits to attach to the complexity of ZF set theory.

This paper is not the first to attempt to quantify the complexity of arithmetical statements. Calude and Calude [5] define a register machine of their own design, and provide quantifications of the complexity of Legendre's conjecture, Fermat's last theorem, Goldbach's conjecture, Dyson's conjecture, the Riemann hypothesis, and the four color theorem.² In addition, Koza [12] and Pargellis [18] each invent instruction sets that are particularly well-suited to representing self-reproducing programs simply, and show that starting from a "primordial soup" of such instructions distributed about a large memory, along with an increasing number of program threads, a rich ecosystem of increasingly efficient self-reproducing programs start to dominate the "landscape."

This paper differs from the previous work in two ways: firstly, it is the first to give explicit, relatively small machines whose behavior is provably independent of the standard axioms of modern mathematics. Secondly, to our knowledge, this paper is the first concrete study of parsimony to use Turing machines as the model of computation—rather than (for example) a new programming language proposed by the authors! We consider it important to use the weakest and most common model of computation for complexity comparisons across different mathematical statements. This is because the more powerful and complex the model of computation used, the more of the complexity of the algorithm can be "shunted" onto the model of computation, and the greater the potential distortion created by the choice of model. As a *reductio ad absurdum*, we could imagine a programming language that included "test the Riemann hypothesis" and "test the consistency of ZFC" as primitive operations. By using the "weakest" model of computation that is commonly known, and one which is generally accepted as the mathematical basis of algorithms, we hope to avoid this pitfall and make it easier to interpret our results in a model-independent way.

3 A Turing Machine that Cannot Be Shown to Run Forever Using ZFC

We present a 7,918-state Turing machine whose behavior is *independent of ZFC*; it is not possible to prove that this machine halts or doesn't halt using the axioms of ZFC, assuming that a stronger set theory is consistent. It's therefore impossible to prove the value of $BB(7,918)$ to be any given value without assuming axioms more powerful than ZFC, assuming that ZFC is consistent.

For an explicit listing of this machine, see Appendix C.

We call this machine Z . One way to build this machine would be to start with the axioms of ZFC and apply the inference rules of first-order logic repeatedly in each possible way so as to enumerate every statement ZFC could prove, and to halt if ever a contradiction was found. While this method is conceptually simple, to actually construct such a machine would lead to a huge number of states, because it would require writing a program to manipulate the axioms of ZFC

²Because Fermat's last theorem and the four color theorem have been proved, their "complexity" is now known to be 1—the minimum number of states in a Turing machine that runs forever.

and the inference rules of first-order logic, and then compiling that program all the way down to Turing machine states.

3.1 Friedman’s Mathematical Statement

Thankfully, a simpler method exists for creating Z . Friedman [7] was able to derive a graph-theoretic statement whose truth implies the consistency of ZFC, and which will be false if ZFC is inconsistent.³ Here is Friedman’s statement (the notation will be explained in the rest of this section):

Statement 1. *For all $k, n, r > 0$, every order invariant graph on $[\mathbb{Q}]^{\leq k}$ has a free $\{x_1, \dots, x_r, \text{ush}(x_1), \dots, \text{ush}(x_r)\}$ of complexity $\leq (8knr)!$, each $\{x_1, \dots, x_{(8kni)!}\}$ reducing $[x_1 \cup \dots \cup x_i \cup \{0, \dots, n\}]^{\leq k}$. [7]*

If s is a set, the operation $(.)^{\leq k}$ refers to the set of all subsets of s with size at most k .

A graph on $[\mathbb{Q}]^{\leq k}$ is an irreflexive symmetric relation on $[\mathbb{Q}]^{\leq k}$. In other words, it can be thought of as a graph whose vertices are elements of $[\mathbb{Q}]^{\leq k}$, and whose edges are undirected, connect pairs of vertices, and never connect vertices to themselves.

A *free* set is a set such that no pair of elements in that set are connected by an edge.

A number of *complexity* at most c refers to a number that can be written as a fraction a/b , where a and b are both integers with absolute value less than or equal to c . A set has complexity at most c if all the numbers it contains have complexity at most c .

An *order invariant graph* is a graph containing a countably infinite number of nodes. In particular, it has one node for each finite set of rational numbers. The only numbers relevant to the statement are numbers of complexity $(8knr)!$ or smaller. In every description of nodes that follows, the term *node* refers both to the object in the order invariant graph and to the set of numbers that it represents.

In an order invariant graph, two nodes (a, b) have an edge between them if and only if each other pair of nodes (c, d) that is *order equivalent* with (a, b) has an edge between them. Two pairs of nodes (a, b) and (c, d) are *order equivalent* if a and c are the same size and b and d are the same size and if for all $1 \leq i \leq |a|$ and $1 \leq j \leq |b|$, the i -th element of a is less than the j -th element of b if and only if the i -th element of c is less than the j -th element of d .

To give some trivial examples of order invariant graphs: the graph with no edges is order invariant, as is the complete graph. A less trivial example is a graph on $[\mathbb{Q}]^{\leq 2}$, in which each node corresponds to a set of two rational numbers of a given complexity, and there is an edge between two nodes if and only if their corresponding sets a and b satisfy $|a| = |b| = 2$ and $a_1 < b_1 < a_2 < b_2$. (Because edges are undirected in order invariant graphs, such an edge will exist if *either* assignment of the vertices to a and b satisfies the inequality above.)

The *ush()* function takes as input a set and returns a copy of that set with all non-negative numbers in that set incremented by 1.

For vertices x and y , $x \leq_{\text{rlex}} y$ if and only if $x = y$ or $x_{|x|-i} < y_{|y|-i}$ where i is the least integer

³In fact, Friedman’s statement is equivalent to the consistency of SRP (“stationary Ramsey property”), which is a system of axioms more powerful than ZFC. Because SRP is strictly more powerful than ZFC (it in fact consists of ZFC plus some additional axioms), the consistency of SRP implies the consistency of ZFC, and the inconsistency of ZFC implies the inconsistency of SRP.

such that $x_{|x|-i} \neq y_{|y|-i}$.⁴ (The \leq_{rlex} operation creates a lexicographic ordering over the vertices, weighting the last and largest elements of those vertices most heavily.)

Finally, a set of vertices X *reduces* a set of vertices Y if and only if for all $y \in Y$, there exists $x \in X$ such that either $x = y$ or $x \leq_{rlex} y$ and an edge exists between x and y .

3.2 Implementation Methods

To create Z , we needed to design a Turing machine that halts if Statement 1 is false, and loops if Statement 1 is true. Such a Turing Machine's behavior is necessarily independent of ZFC, because the truth or falsehood of Statement 1 is independent of ZFC, assuming the consistency of SRP. [7] SRP is an extension of ZFC by certain relatively mild large cardinal hypotheses, and is widely regarded by set theorists as consistent. For more information about SRP, see [10].

To design such a Turing machine, we wrote a Laconic program which encodes Friedman's statement, then compiled the program down to a description of a single-tape, 2-symbol Turing machine. What follows is an extremely brief description of the design of the Laconic program; for the documented Laconic code itself, along with a detailed explanation of the full compilation process, see [21].

Our Laconic program begins by looping over all non-negative values for k , n , and r . For each trio (k, n, r) , our program generates a list N of all numbers of complexity at most $(8knr)!$. These numbers represent the vertices in our putative order invariant graph. Because Laconic does not support floating-point numbers, the list is entirely composed of integers; it is a list of all numbers that can be written in the form $((8knr)!)((8kni)!)/((8knj)!)$, where i and j are integers satisfying $-(8knr)! \leq i \leq (8knr)!$ and $1 \leq j \leq (8knr)!$. (Note that any number that can be expressed in this form is necessarily an integer, because of the large scaling factor in front.)

After we generate N , we generate the nodes in a potential order invariant graph by adding to N all possible lists of k or fewer numbers from N . We call this list of lists V .

We iterate over all binary lists of length $|V|^2$. Any such list E represents a possible set of edges in the graph. To be more precise, we say that an edge exists between node i and node j (represented by V_i and V_j respectively) if and only if $E_{i|V|+j}$ is 1.

For any graph (V, E) , we say that it is "valid" if the following three conditions hold:

1. No node has an edge to itself.
2. If an edge exists between node i and node j , an edge also exists between node j and node i .
3. The graph has a free $\{x_1, \dots, x_r, \text{ush}(x_1), \dots, \text{ush}(x_r)\}$, each $\{x_1, \dots, x_{(8kni)!}\}$ reducing $[x_1 \cup \dots \cup x_i \cup \{0, \dots, n\}]^{\leq k}$.

For each list of nodes V , we loop over every possible binary list E , and if no pair (V, E) yields a valid graph, we halt.

When verifying the validity of a graph, checking the first two conditions is trivial, but the third merits further explanation. In order to verify that a given graph (V, E) has a free $\{x_1, \dots, x_r, \text{ush}(x_1), \dots, \text{ush}(x_r)\}$, each $\{x_1, \dots, x_{(8kni)!}\}$ reducing $[x_1 \cup \dots \cup x_i \cup \{0, \dots, n\}]^{\leq k}$, we look at every possible subset of the nodes in V . For each subset, we verify that it has length r ,

⁴Friedman recommended in private communication that we use the \leq_{rlex} comparator to compare vertices, instead of comparing their maximum elements as described in his manuscript.

that $\text{ush}(x_1), \dots, \text{ush}(x_r)$ all exist in V , and for each i such that $(8kni)! \leq r$, that $\{x_1, \dots, x_{(8kni)!}\}$ reduces $[x_1 \cup \dots \cup x_i \cup \{0, \dots, n\}]^{\leq k}$. Once we have found such a subset, we know that the third condition is satisfied.

4 A Turing Machine that Encodes Goldbach's Conjecture

We present a 4,888-state Turing machine that *encodes Goldbach's conjecture*; in other words, to know whether this machine halts is to know whether Goldbach's conjecture is true. It is therefore impossible to prove the value of $BB(4,888)$ without simultaneously proving or disproving Goldbach's conjecture.

Recall that Goldbach's conjecture is as follows:

Statement 2. Every even integer greater than 2 can be expressed as the sum of two primes.

Because Goldbach's conjecture is so simple to state, the Laconic program encoding the statement is also quite simple. It can be found in Appendix A. A detailed explanation of the compilation process, documentation for the Laconic language, and an explicit description of this Turing machine are available at [21].

5 A Turing Machine that Encodes Riemann's Hypothesis

We present a 5,372-state Turing machine that *encodes Riemann's hypothesis*; in other words, to know whether this machine halts is to know whether Riemann's hypothesis is true. An explicit description of this machine can be found at [21]

Riemann's hypothesis is traditionally stated as follows:

Statement 3. The Riemann zeta function has its zeros only at the negative even integers and the complex numbers with real part $1/2$.

5.1 Equivalent Statement

Instead of encoding the Riemann zeta function into a Laconic program, it is simpler to use the following statement, which was shown by Lagarias [4] to be equivalent to the Riemann hypothesis:

Statement 4. For all integers $n \geq 1$,

$$\left(\left(\sum_{k \leq \delta(n)} \frac{1}{k} \right) - \frac{n^2}{2} \right)^2 < 36n^3$$

The function $\delta(n)$ used in Statement 4 is defined as follows:

$$\begin{aligned} \eta(j) &= p \text{ if } j = p^k, p \text{ is prime, } k \text{ is a positive integer} \\ \eta(j) &= 1 \text{ otherwise} \\ \delta(x) &= \prod_{n < x} \prod_{j \leq n} \eta(j) \end{aligned}$$

5.2 Implementation Methods

Statement 4 is equivalent to the following statement, which contains only positive integers⁵:

$$l(n) < r(n)$$

for all positive integers n , where

$$\begin{aligned} l(n) &= a(n)^2 + b(n)^2 \\ r(n) &= 36n^3(\delta(n)!)^2 + 2a(n)b(n) \end{aligned}$$

$$\begin{aligned} a(n) &= \sum_{k \leq \delta(n)} \frac{\delta(n)!}{k} \\ b(n) &= \frac{n^2 \delta(n)!}{2} \end{aligned}$$

To check the Riemann hypothesis, our program computes $a(n)$, $b(n)$, $l(n)$, and $r(n)$, in that order, for each possible value of n . If $l(n) \geq r(n)$, our program halts.

6 Laconic

Laconic is a programming language designed to be both user-friendly and easy to compile down to parsimonious Turing machine descriptions.

Laconic is a strongly-typed language that supports recursive functions. Laconic compiles to an intermediate language called TMD. TMD programs are spread across multiple files and grouped into directories. TMD directories are meant to represent sequences of commands that could be given to a multi-tape, 3-symbol Turing machine, using the Turing machine abstraction that allows the machine to read and write from one head at a time.

For an example of a Laconic program, see Appendix A. For an illustration of the compilation process, see Figure 1.

7 TMD

TMD is a programming language designed to help the user describe the behavior of a multi-tape, 3-symbol Turing machine with a function stack. Each tape is infinite in one direction and supports three symbols: $_$, 1, and E. The blank symbol is $_$: that is, $_$ is the only symbol that can appear on the tape an infinite number of times. The tape must always have the form $_?(1|E)^+_$; in other words, each tape must always contain a string of 1's and E's of size at least 1, possibly preceded by a $_$ symbol, and necessarily followed by an infinite number of copies of the $_$ symbol.

What is the purpose of having a language like TMD as an intermediary between Laconic and a description of a single-tape machine? The concept of tapes in a multi-tape Turing machine and the concept of variables in standard imperative programming languages map to one another very nicely. The idea of the Laconic-to-TMD compiler is to encode the value of each variable on one tape. Then,

⁵Although it is not immediately obvious, $\frac{\delta(n)!}{k}$ is necessarily an integer for all $k \leq \delta(n)$, and $\frac{\delta(n)!}{2}$ is an integer for all $n > 1$.

each Laconic command that manipulates the value of one or more variables compiles down to a TMD function call that manipulates the tapes that correspond to those variables appropriately.

As an example, consider the following Laconic command:

```
a=b*c;
```

This Laconic command assigns the value of `a` to the value of `b*c`. It compiles down to the following TMD function call:

```
function BUILTIN_multiply a b c
```

This function call will result in `BUILTIN_multiply` being run on the three tapes `a`, `b`, and `c`. This will cause the symbols on tape `a` to take on a representation of an integer whose value is equal to `bc`.

In turn, the TMD code compiles directly to a string of bits that are written onto the tape at the start of the Turing machine's execution.

A TMD directory consists of three types of files:

1. The `functions` file. This file contains a list of the names of all the functions used by the TMD program. The top function in the file is pushed onto the stack at initialization. When this top function returns, the Turing machine halts.
2. The `initvar` file. This file contains the non-`_` symbols that start in each register at initialization.
3. Any files used to describe TMD functions. These files all end in a `.tfn` extension and only have any relevance to the compiled program if they show up in the `functions` file.

8 Compilation and Processing

There are two ways to think about the layout of the tape symbols: with a 4-symbol alphabet (`{_, 1, H, E}`, blank symbol `_`), and with a 2-symbol alphabet (`{a, b}`, blank symbol `a`). The 2-symbol alphabet version is the one that's ultimately used for the results in this paper, since we advertised a Turing machine that used only two symbols. However, in nearly all parts of the Turing machine, the 2-symbol version of the machine is a direct translation of the 4-symbol version, according to the following mapping:

- `_` \leftrightarrow `aa`
- `1` \leftrightarrow `ab`
- `H` \leftrightarrow `ba`
- `E` \leftrightarrow `bb`

The sections that follow sometimes refer to the `ERROR` state. Transitions to the `ERROR` state should never be taken under any circumstances, and are useful for debugging purposes.

8.1 Concept

A directory of TMD functions is converted at compilation time to a string of bits to be written onto the tape, along with other states designed to interpret these bits. The resulting Turing machine has three main components, or *submachines*:

1. The *initializer* sets up the basic structure of the variable registers and the function stack.
2. The *printer* writes down the binary string that corresponds to the compiled TMD code.
3. The *processor* interprets the compiled binary, modifying the variable registers and the function stack as necessary.

The Turing machine’s control flow proceeds from the initializer to the the printer to the interpreter. In other words, initializer states point only to initializer states or to printer states, printer states point only to printer states or to interpreter states, and interpreter states point only to interpreter states or the HALT state.

This division of labor, while seemingly straightforward, actually constitutes an important idea. The problem of the compiler is to convert a higher-level representation—a machine with many tapes, a larger alphabet, and a function stack—to the lower-level representation of a machine with a single tape, a 2-symbol alphabet and no function stack. The immediately obvious solution, and the one taught in every computability theory class as a proof of the equivalence of different kinds of Turing machines, is to have every “state” in the higher-level machine compile down to many states in the lower-level machine.

While simple, this approach is suboptimal in terms of the number of states. As is nearly always true when designing systems to be parsimonious, the clue that improvement is possible lies in the presence of repetition. Each state transition in the higher-level machine is converted to a group of lower-level states with the same basic structure. Why not instead explain how to perform this conversion exactly once, and then apply the conversion many times?

This idea is at the core of the division of labor described previously. We begin by writing a description of the higher-level machine onto the tape, and then “run” the higher-level machine by reading what is on the tape with a set of states that understands how to interpret the encoded higher-level machine. We refer to this idea as *on-tape processing*.

In this paper, we use TMD as the representation of the higher-level machine.⁶ The printer writes the TMD program onto the tape, and the processor executes it. As a result of using this scheme, we incur a constant *additive* overhead—we have to include the processor in our final Turing machine—but we avoid the constant *multiplicative* overhead required for the naïve scheme.

Is this additive overhead small enough to be worth it? We found that it is. Our implementation of the processor requires 3,860 states. (See Section 8.5 for a detailed breakdown of the state cost by submachine.) In contrast to this additive overhead of 3,860, the naïve approach incurs a large multiplicative overhead that depends in part on how many states must be used to represent each higher-level state transition, and in part on how efficient an encoding scheme can be devised for

⁶Note that instead of TMD, the on-tape processing scheme could be used for any language, assuming the designer provides both a processor and an encoding for that language. We chose TMD because it made the interpreter easy to write, but other minimalist languages, like Unlambda [14], Brainf*ck [17], or Iota and Jot [1], might be good candidates for parsimonious designs, with the additional advantage of being already known to some programmers! Thanks to Luke Schaeffer for this point.

the on-tape approach. The following table compares the performance of on-tape processing to the performance of an implementation that used the naïve approach. The comparison is shown for three kinds of machines: a machine that halts if and only if Goldbach’s conjecture is false, a machine that halts if and only if the Riemann hypothesis is false, and a machine whose behavior is independent of ZFC.

Program	States (Naïve)	States (On-Tape Processing)
Goldbach	7,902	4,888
Riemann	36,146	5,372
ZFC	340,943	7,918

As can be seen from this table, on-tape interpretation results in huge gains, particularly in large and complex programs.

The subsections that follow describe each of the three submachines—the initializer, the printer, and the processor—in greater detail.

8.2 The Initializer

The initializer starts by writing a counter onto the tape which encodes how many registers there will be in the program. Using the value in that counter, it creates each register, with demarcation patterns between registers, and unique identifiers for each register. Each register’s value begins with the pattern of non-`_` symbols laid out in the `initvar` file. The initializer also creates the program counter, which starts at 0, and the function stack, which starts out with only a single function call to the top function in the `functions` file.

Figure 2 is a detailed diagram describing the tape’s state when the initializer passes control to the printer.

8.3 The Printer

8.3.1 Specification

The printer writes down a long binary string which encodes the entirety of the TMD program onto the tape.

Figure 3 shows the tape’s state when the printer passes control to the processor.

8.3.2 Introspection

Writing down a long binary string onto a Turing machine tape in a parsimonious fashion is not as straightforward as it might initially appear. The first idea that comes to mind is simply to use one state per symbol, with each state pointing to the next, as shown in Figure 4.

On closer examination, however, this approach is quite wasteful for all but the smallest binary files. Every `a` transition points to the next state in the sequence, and none of the `b` transitions are used at all! Indeed, the only information-bearing part of the state is the single bit contained in the choice of which symbol to write. But in theory, far more information than that could be encoded in each state. In a machine with n states, each state could contain $2(\log_2(n) + 1)$ bits of information, because each of its two transitions could point to any of the n states, and write either

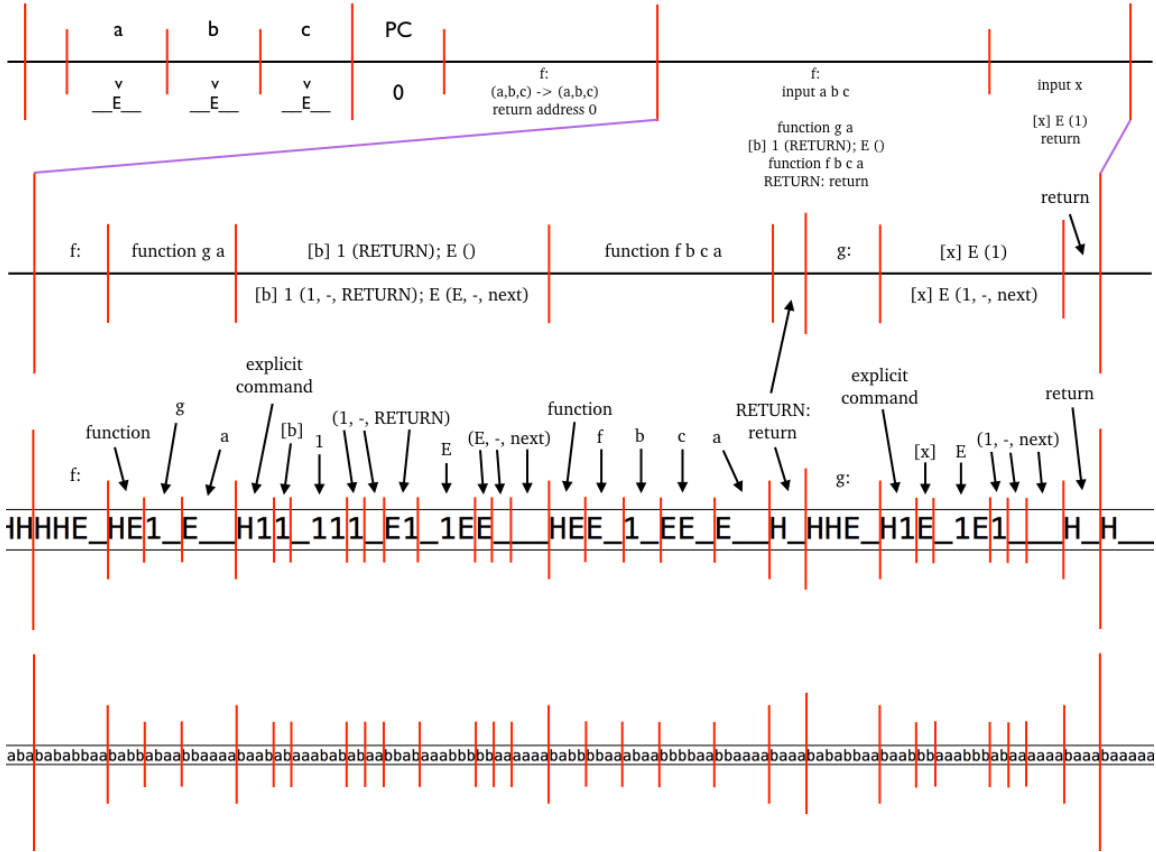


Figure 3: The state of the Turing machine tape after the printer completes. The TMD program being expressed in Turing machine form is described in full in Appendix B. The top bar is a high-level description of the entire tape; unfortunately, at this point there are so many symbols on the tape that it is impossible to see everything at once. For a detailed view of the first two-thirds of the tape (registers, program counter, and stack), see Figure 2. The bottom three bars show a zoomed-in view of the program binary. From the top, the second bar gives a high-level description of what each part of the program binary means; the third bar gives the direct correspondence between 4-symbol alphabet symbols on the tape and their meaning in TMD; the fourth and final bar gives the translation of the third bar into the 2-symbol alphabet. For a more detailed explanation of the encoding of TMD into tape symbols, see [21].

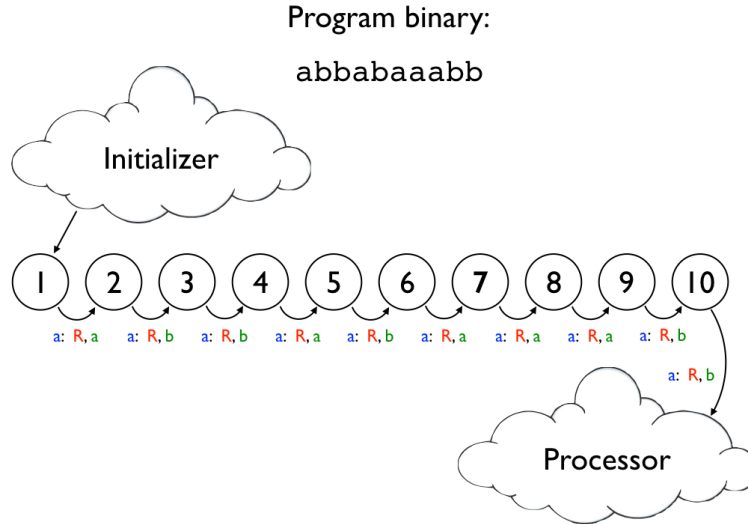


Figure 4: A naïve implementation of the printer. In this example, the hypothetical program is ten bits long, and the printer uses ten states, one for each bit. In the diagram, the blue symbol is the symbol that is read on a transition, the red letter indicates the direction the head moves, and the green symbol indicates the symbol that it written. Note the lack of transitions on reading a **b**; this is because in this implementation, the printer will only ever read the blank symbol, which is **a**, since the head is always proceeding to untouched parts of the tape. It therefore makes no difference what behavior the Turing machine adopts upon reading a **b** in states 1-10 (and therefore **b** transitions are presumed to lead to the ERROR state)

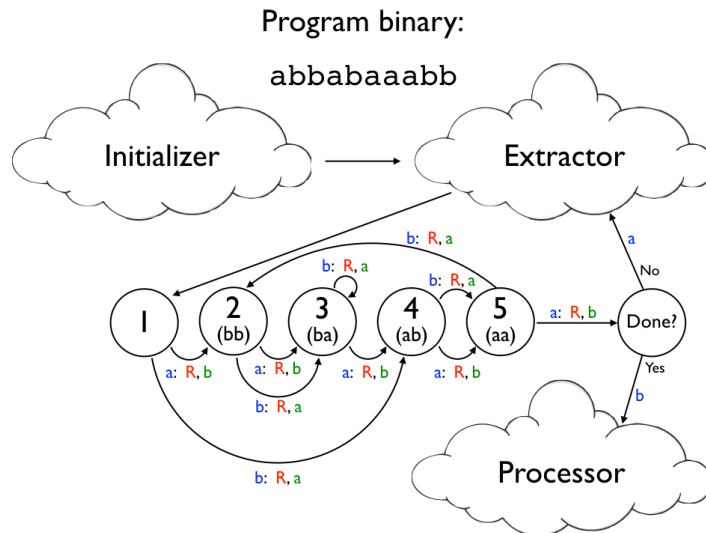


Figure 5: An introspective implementation of the printer. In this example, the hypothetical program is $k = 10$ bits long, and so the word size must be 2 (since $w = 2$ is the largest w such that $w2^w \leq 10$). There are therefore $n_w = \lceil \frac{k}{w} \rceil = 5$ data states, each encoding two bits. The **b** transitions carry the information about the encoding; note that each one only points to one of the last four data states. The last four data states have in parentheses what word we mean to encode if we point to them.

an **a** or a **b** onto the tape. Of course, this is only in theory; in practice, to extract the information contained in the Turing machine’s states and translate it into bits on the tape is nontrivial.

We will use a scheme originally conceived by Ben-Amram and Petersen [2] and refined further and suggested to us by Luke Schaeffer. It does not achieve the optimal theoretical encoding described above, but is relatively simple to implement and understand, and is within a factor of 2 of optimal for large binary strings. Schaeffer named Turing machines that use this idea *introspective*.

Introspection works as follows. If the binary string contains k bits, then let w be the *word size*. w takes the largest value it can such that $w2^w \leq k$. We can split the binary string into $n_w = \lceil \frac{k}{w} \rceil$ words of w bits each (we can pad the last word with copies of the blank symbol). In our scheme, each word in the bit-string is represented by a *data state*. Each data state points to the state representing the next word in the sequence for its **a** transition, but which state the **b** transition points to encodes the next word. Every **b** transition points to one of the last 2^w data states, thereby encoding w bits of information.

Of course, the encoding is useless until we specify how to extract the encoded bit-string from the data states. The extraction scheme works as follows. To query the i^{th} data state for the bits it encodes, we run the data states on the string $\mathbf{a}^{i-1}\mathbf{b}\mathbf{a}^\infty$ (a string of $i - 1$ **a**’s followed by a **b** in the i^{th} position). After running the data states on that string, what remains on the tape is the string $\mathbf{b}^{i-1}\mathbf{a}\mathbf{b}^r\mathbf{a}^\infty$, assuming that the i^{th} data state pointed to the r^{th} -to-last data state. Thus, what we’re left with is essentially a unary encoding of the “value” of the word in binary. Thus, the job of the extractor is to set up a binary counter which removes one **b** at a time and increments the counter appropriately. Then, afterward, the extractor reverts the tape back to the form $\mathbf{a}^i\mathbf{b}\mathbf{a}^\infty$, shifts all symbols on the tape over by w bits, and repeats the process. Finally, when the state beyond the last data state sees a **b** on the tape, we know that the process has completed, and we can pass control to the processor. Figure 5 shows the whole procedure.

How much have we gained by using introspection for encoding the program binary, instead of the naïve approach? It depends on how large the program binary is. Using introspection incurs an $O(\log k)$ *additive* overhead, because we have to include the extractor in our machine. (Our implementation of the extractor takes $10w + 17$ states.) In return, we save a *multiplicative* factor of w (which scales with $\log k$) on the number of data states needed.

This is plainly not worth it for the 10-bit example binary shown in Figs. 4 and 5. For that binary, we require 69 additional states for the extractor in order to save 5 data states. For real programs, however, it is worth it, as can be seen from the following table.

Program	Binary Size	w	n_w	Extractor Size	States (Naïve)	States (Introspective)
Example TMD	116	4	29	57	116	86
Goldbach	4,964	9	552	107	4,964	659
Riemann	9,532	10	1,024	117	9,532	1,141
ZFC	38,956	11	3,542	127	38,956	3,621

One minor detail concerns the numbers presented for the Riemann program. Ordinarily, with a binary of size 9,532, we would opt to split the program into 1,060 words of 9 bits each plus a 107-state extractor, since 9 is the greatest w such that $w2^w < 9,532$. But because 9,532 is so close to the “magic number” 10,240, it’s actually more parsimonious to pad the program with copies of

the blank symbol until it's 10,240 bits long, and split it into 1,024 words of 10 bits each plus a 117-state extractor.

8.4 The Processor

The processor's job is to interpret the code written onto the tape and modify the variable registers and function stack accordingly. The processor does this by the following sequence of steps:

START:

1. Find the function call at the top of the stack. Mark the function f in the code whose ID matches that of the top function call.
2. Read the current program counter. Mark the line of code l in f whose line number matches the program counter.
3. Read l . Depending on what type of command l is, carry out one of the following three lists of tasks.

IF l IS AN EXPLICIT TAPE COMMAND:

1. Read the variable name off l . Index the variable name into the list of variables in the top function on the stack. This list of variables corresponds to the mapping between the function's local variables and the register names.
2. Match the indexed variable to its corresponding register r . Mark r . Read the symbol s_r to the right of the head marker in that register.
3. Travel back to l , remembering the value of s_r using states. Find and mark the reaction x corresponding to the symbol. See what symbol s_w should be written in response to reading s_r .
4. Travel back to r , remembering the value of s_w using states. Replace s_r with s_w .
5. Travel back to x . See which direction d the head should move in response to reading s_r .
6. Travel back to r , remembering the value of d using states. Move the head marker accordingly.
7. Travel back to x . See if a jump is specified. If a jump is specified, copy the jump address onto the program counter. Otherwise, increment the program counter by 1.
8. Go back to START.

IF l IS A FUNCTION CALL:

1. Write the function's name to the top of the stack.
2. For each variable in the function call, index the variable name into the list of variables in the top function on the stack. This list of variables corresponds to the mapping between the function's local variables and the register names. Push the corresponding register names in the order that they correspond to the variables in the function call.

3. Copy the current program counter to the return address of the newborn function call at the top of the stack.
4. Replace the current program counter with 0 (meaning “read the first line of code”).
5. Go back to START.

IF l IS A RETURN STATEMENT:

1. Replace the current program counter with f 's return address.
2. Increment the program counter by 1.
3. Erase the call to f from the top of the stack.
4. Check if the stack is now empty. If so, halt.
5. Go back to START.

8.5 Cost Analysis

It's worthwhile to analyze the relative contributions of the initializer, the printer, and the processor to the machine's final state count. The following table lists the number of states in each submachine for each of the four different TMD programs under discussion.

Program	Initializer	Printer	Processor	Total
Example TMD	349	86	3,860	4,295
Goldbach	369	659	3,860	4,888
Riemann	371	1,141	3,860	5,372
ZFC	389	3,621	3,860	7,918

As can be seen from this table, the processor makes the largest contribution to all four programs. Improving the processor, therefore, is probably the best approach for improving upon the bounds we present. Equally clear, however, is that for programs more complicated than the ones presented here, the cost of the printer will grow almost linearly but the cost of the processor will stay the same. The cost of the initializer grows very slightly with the complexity of programs because of the need to initialize additional registers.

Improving the printer, and with it the TMD and Laconic languages, is probably the best approach for reducing state count for very large and complex programs.

9 Future Work

This paper still leaves a three orders-of-magnitude gap between the smallest n , namely 7,918, for which $BB(n)$ is known to be independent of ZF set theory, and the largest n , namely 4, for which $BB(n)$ is known to be determinable. We regard it as a fascinating problem to pin down the truth here: for example, is it conceivable that $BB(10)$ or even $BB(6)$ might be independent of ZF? If so, that would arguably force a qualitative change in our understanding of the Gödel incompleteness phenomenon—showing that incompleteness from strong set theories rears its head for much simpler arithmetical questions than had previously been known.

A more immediate question is how much further Z 's state count can be reduced. Without a significant new insight, further order-of-magnitude reductions seem unlikely via tweaks to our existing system: note that such tweaks would need to reduce the sizes of both the printer and the processor by an order of magnitude. However, improvement is possible by (for example) a redesigned processor, combined with a simpler independent mathematical statement than Friedman's.

Other future work might involve further use of our Laconic language to upper-bound the 'complexities' of mathematical statements and algorithms, in as standardized and model-independent a way as possible. Perhaps Laconic could be used to measure the complexity of other well-known conjectures, or even to compare different algorithms for solving the same problem to each other (e.g., to try to quantify the notion that an insertion sort is simpler than a merge sort)!

10 Acknowledgements

We thank Prof. Harvey Friedman for having done the crucial theoretical work that made this project feasible. Prof. Friedman was endlessly available over email, and provided us with detailed clarifications when we needed them.

We thank Luke Schaeffer for his early help, as well as his help designing introspective Turing machines.

We thank Alex Arkhipov for introducing us to the term "code golfing."

Supported by an Alan T. Waterman Award from the National Science Foundation, under grant no. 1249349.

References

- [1] Barker, C. "Iota and Jot: the Simplest Languages?" <http://semarch.linguistics.fas.nyu.edu/barler/Iota/> [A website describing the Iota and Jot programming languages]
- [2] Ben-Amram, A., Petersen, H. "Improved Bounds for Functions Related to Busy Beavers" *Theory of Computing Systems* 35, 1-11 (2002)
- [3] Brady, A.H. "Solution of the Non-computable 'Busy Beaver' game for $k = 4$." Abstracts for: ACM Computer Science Conference (Washington, DC, February 18-20, 1975), p. 27, ACM, 1975.
- [4] Browder, F. "Mathematical Developments Arising from Hilbert Problems." *American Mathematical Society*. Volume 28, Part 1.
- [5] Calude, C., Calude, E. "Evaluating the Complexity of Mathematical Problems: Part 1," "Evaluating the Complexity of Mathematical Problems: Part 2." *Complex Systems* 18, pp. 387-401. 2010.
- [6] Chaitin, G. "The Limits of Mathematics." pp. 79. 1994.
- [7] Friedman, H. "Order Invariant Graphs and Finite Incompleteness." <https://u.osu.edu/friedman.8/files/2014/01/FIiniteSeqInc062214a-v9w7q4.pdf>

- [8] Personal communications with H. Friedman.
- [9] Friedman, H. "Order Theoretic Equations, Maximality, and Incompleteness." June 7, 2014. <http://u.osu.edu/friedman.8/foundational-adventures/downloadable-manuscripts/#78>.
- [10] Friedman H. "The Upper Shift Kernel Theorems." October 9, 2010. <https://u.osu.edu/friedman.8/files/2014/01/KernStruThm100910-1lu0b8v.pdf>
- [11] Gödel, K. "The Consistency of the Axiom of Choice and of the Generalized Continuum-Hypothesis with the Axioms of Set Theory." Published in 1940 by the Princeton University Press. *Annals of Mathematics Studies*.
- [12] Koza, J. "Spontaneous Emergence of Self-Replicating and Evolutionarily Self-Improving Computer Programs." in *Artificial Life III (SFI Studies in the Sciences of Complexity, vol. XVII)*, C. G. Langton, Ed. Reading, MA: Addison-Wesley. pp. 225-262. 1994.
- [13] Lin, S., Rado, T. "Computer Studies of Turing Machine Problems." Published in *Journal of the ACM*, Volume 12, Issue 2, April 1965. Pages 196-212.
- [14] Madore, D. "The Unlambda Programming Language." <http://www.madore.org/~david/programs/unlambda/>
[A website describing the Unlambda programming language]
- [15] Marxen, H., Buntrock, J. "Attacking the Busy Beaver 5." *Bull EATCS*, Vol. 40, pp. 247-251. 1990.
- [16] Marxen, H. <http://www.drb.insel.de/~heiner/BB/>
[A list of the known busy beaver values]
- [17] Müller, U. "Brainfuck." <http://www.muppetlabs.com/~breadbox/bf/>
[A website describing the Brainf*ck programming language]
- [18] Pargellis, A. "The Spontaneous Generation of Digital 'Life.'" *Physica D*, 91, 86-96. 1996.
- [19] Rado, T. "On Non-Computable Functions." *Bell System Technical Journal*, 41: 3. May 1962 pp 877-884.
- [20] Schoenfield, J. "The Problem of Predicativity." *Essays on the foundations of mathematics*, Y. Bar-Hillel et al., eds., pp. 132-142. 1961.
- [21] Yedidia, A. <https://github.com/adamyedidia/parsimony>
[A link to a GitHub repository containing all programs and Turing machines related to this paper, with accompanying documentation.]
- [22] <http://codegolf.stackexchange.com/>
[A place where programmers go for recreational code golfing]

Appendices

A Example Laconic Program: Goldbach's Conjecture

The following is an example Laconic program, which compiles down to the Turing machine G mentioned in Section 4 (which halts if and only Goldbach's Conjecture is false).

```
func zero(x) {
    x = 0;
    return;
}

func one(x) {
    x = 1;
    return;
}

func incr(x) {
    x = x + 1;
    return;
}

/* Computes x modulo y */
func modulus(x, y, out) {
    out = x;

    while (out >= y) {
        out = out - y;
    }

    return;
}

func assignXtoYminusX(x, y) {
    x = y - x;
    return;
}

/* Figures out if x is prime, and puts the output in y */
/* Does not modify x, modifies y */
func isPrime(x, h, y) {
    if (x == 1) {
        zero(y);
        return;
    }

    y = 2;

    while (x > y) {
        modulus(x, y, h);

        if (h == 0) {
            zero(y);
            return;
        }
        incr(y);
    }

    return;
}

int evenNumber;
```

```

int primeCounter;
int isThisOnePrime;
int foundSum;
int h;

evenNumber = 2;
one(foundSum);

while (foundSum) {
    zero(foundSum);
    evenNumber = evenNumber + 2;
    one(primeCounter);

    while (primeCounter < evenNumber) {
        isPrime(primeCounter, h, isThisOnePrime);

        if (isThisOnePrime) {
            assignXtoYminusX(primeCounter, evenNumber);
            isPrime(primeCounter, h, isThisOnePrime);
            assignXtoYminusX(primeCounter, evenNumber);

            if (isThisOnePrime) {
                print evenNumber;
                print primeCounter;

                one(foundSum);
            }
        }

        incr(primeCounter);
    }
}

halt;

```

For detailed documentation of the Laconic programming language, see [21]. To find this file specifically, navigate to `parsimony/src/laconic/laconic_files/goldbach.lac` at [21].

B Example TMD Program

The following is an example TMD directory, which compiles down to a binary string to be written on a Turing machine tape. It's the example used in illustrations throughout this paper, most notably in the example compilation shown in Figs. 2 and 3. The program calls itself recursively three times until the starting symbol on each tape, E, is replaced with a 1, at which point the program halts.

This TMD directory is called `example_tmd_dir`, and contains four files: `f.tmd`, `g.tmd`, `initvar`, and `functions`.

```

f.tmd:
input a b c

// Recursively writes a 1 on every tape.

function g a
[b] 1 (RETURN); E ()
function f b c a
RETURN: return

```



```

g.tmd:
input x
// Writes a 1 on the input tape.

[x] E (1)
return
  functions:
f
g
  initvar:
E

```

For detailed documentation of the TMD programming language, see [21]. To find this directory specifically, navigate to `parsimony/src/tmd/tmd_dirs/example_tmd_dir/` at [21].

C Explicit Listing of Z

A description of a single state in Z

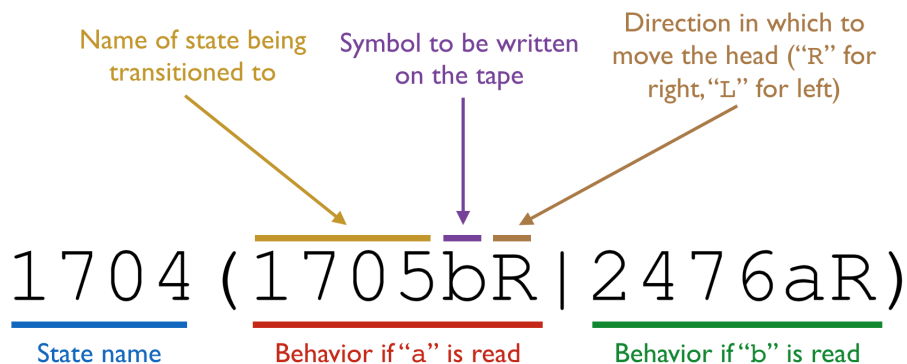


Figure 6: This figure explains how to read a description of a single state. Note that “ERROR-” or “HALT--” denote transitions to the ERROR or HALT states, respectively (no further information is provided because what symbol is written and which direction the head moves are at that point irrelevant).

We present below an explicit listing of Z . For a more easily readable version of Z , complete with descriptive state names, see [21].

We ran this Turing machine for 10,000,000,000 steps (more than half a day on our simulators) and within that time it did not halt. We note, however, that Z was designed for parsimony rather than efficiency, and that this “experiment” is of little consequence! We similarly ran a Turing machine designed to test the conjecture that all perfect squares are less than 5, and it ran for

2,895,083,899 steps (a couple hours on our simulator) before it found the counterexample 9 and halted.

Figure 6 provides help to interpret the description shown below. In addition, note the following:

1. The tape has a 2-symbol alphabet, with tape symbols {a, b} and blank symbol a (in other words, a is the only symbol that can appear an infinite times on the tape).
2. The start state of Z is 0000.
3. Z will never transition to the ERROR state. Any transition to the ERROR state could be replaced by a transition to any other state (including HALT) and the Turing machine's behavior would remain identical.
4. Z contains only one transition to the HALT state, out of state 7870.

0000(0001aR) ERROR--	0001(0004aR) ERROR--	0002(0003aR) ERROR--	0003(0012aR) 0012bR	0004(0005aR) ERROR--	0005(0006aR) ERROR--	0006(0007bR) ERROR--	0007(0008bR) ERROR--	0008(0009aR) ERROR--	0009(0010bR) ERROR--
0010(0011bR) ERROR--	0011(0002aR) ERROR--	0012(0031aR) ERROR--	0013(0001aR) 0014bR	0014(0005aR) ERROR--	0015(0007aR) 0057bR	0016(0017bR) ERROR--	0017(0018bR) ERROR--	0018(0019aR) ERROR--	0019(0020aR) 0020bR
0020(0021aR) ERROR--	0021(0022aR) 0022bR	0022(0023aR) ERROR--	0023(0024aR) 0024bR	0024(0025aR) ERROR--	0025(0027aR) 0027bR	0026(0027aR) 0027bR	0027(0028aR) 0028bR	0028(0029aR) 0029bR	0029(0030aR) 0030bR
0030(0031aR) 0031bR	0031(0026aR) 0026bR	0032(0035aR) 0035bR	0033(0034aR) 0034bR	0034(0037aR) 0037bL	0043(0037aL) 0037bL	0044(0040aR) 0040bR	0045(0046aR) 0046bR	0046(0049aL) 0049bL	0047(0050aR) 0050bR
0050(0049aL) 0049bR	0051(0052aR) 0049bR	0052(0053aR) 0054bR	0053(0052aR) 0052bR	0054(0055aL) 0055bL	0055(0056aR) 0056aR	0056(0012aR) 0012bR	0057(0058aR) ERROR--	0058(0059aR) 0059bR	0059(0060aR) 0060bR
0060(0058aR) 0058bR	0061(0062aR) 0062bR	0062(0063aR) 0063bR	0063(0064aR) 0064bR	0064(0065aR) 0065bR	0065(0066aR) 0066bR	0066(0016aR) 0016bR	0067(0068aR) ERROR--	0068(0069aR) 0069bR	0069(0070aR) 0070bR
0070(0062aL) 0062bL	0071(0072aR) 0072bR	0072(0071aR) 0073bL	0073(0074aR) 0074bR	0074(0075aL) 0075bL	0075(0066aR) 0076bL	0076(0077aR) 0077bR	0077(0078aL) 0078bL	0078(0079aR) 0079bR	0079(0080aR) 0078bR
0080(0081aR) 0081bR	0081(0083aR) 0083bR	0082(0078aR) 0078bR	0083(0084aR) 0088bR	0084(0082aR) 0083bR	0085(0086aL) 0083bR	0086(0087aL) 0087bL	0087(0088aL) 0088bL	0088(0089aR) 0094bR	0089(0090aL) 0092bL
0090(0091aR) 0091bR	0091(0091aL) 0091bL	0092(0091aL) 0093bL	0093(0088aL) 0088bL	0094(0095aL) 0097bL	0095(0095aL) 0096bL	0096(0088aL) 0088bL	0097(0098aL) 0098bL	0098(0099aL) 0098bL	0099(0100aR) 0099bR
0100(0101aL) 0101bL	0101(0102aL) 0102bL	0102(0099aL) 0099bL	0103(0104aR) 0104bR	0104(0101aL) 0110bL	0105(0106aL) 0108bL	0106(0107aR) 0107aR	0107(0139aL) 0139bL	0108(0109aR) 0109bR	0109(0110aL) 0110bL
0110(0111aR) 0111bR	0111(0112bL) 0111bL	0112(0113bL) 0113bL	0113(0130aL) 0130bL	0114(0115bL) 0115bL	0115(0110aL) 0110bL	0116(0117aR) 0117aR	0117(0118aR) 0118aR	0118(0119aL) 0119aL	0119(0120aR) 0120bR
0120(0121aL) 0121bL	0131(0132aL) 0132bL	0132(0131aL) 0131bL	0133(0130aL) 0130bL	0134(0135aR) 0135bR	0135(0088aR) 0088bL	0136(0139aR) 0139bR	0137(0138aL) 0138aL	0138(0139aL) 0139aL	0139(0140aR) 0140bR
0140(0139aR) 0141bL	0141(0142aR) 0142bR	0142(0146aL) 0146bL	0143(0144aR) 0144bR	0144(0145aR) 0145bR	0145(0146aL) 0146bL	0146(0147aR) 0147aR	0147(0148aL) 0148aL	0148(0149aR) 0149aR	0149(0149aL) 0149bL
0150(0151aL) 0151bL	0152(0152aR) 0152bR	0152(0152aR) 0152bR	0153(0153aR) 0153bR	0154(0154aR) 0154bR	0155(0155aR) 0155bR	0156(0156aL) 0156bL	0157(0157aR) 0157aR	0158(0158aR) 0158bR	0159(0159aL) 0159bL
0160(0157aL) 0157bL	0161(0162aR) 0162bR	0162(0166aL) 0166bL	0163(0197aR) 0197bR	0164(0165aR) 0165bR	0165(0166aL) 0166bL	0166(0167aR) 0167aR	0167(0168aL) 0168aL	0168(0169aR) 0169aR	0169(0177aL) 0177bL
0170(0171aL) 0171bL	0171(0166aL) 0166bL	0172(0173aL) 0173bL	0173(0174aL) 0174bL	0174(0166aL) 0166bL	0175(0176aL) 0176bL	0176(0166aL) 0166bL	0177(0178aR) 0178aR	0178(0179aL) 0181bL	0179(0180aL) 0180bL
0180(0177aL) 0177bL	0181(0166aL) 0166bL	0192(0191aL) 0191bL	0193(0194aL) 0194bL	0194(0186aL) 0186bL	0195(0196aL) 0196bL	0196(0186aL) 0186bL	0197(0198aR) 0198bR	0198(0189aR) 0189bR	0189(0187aL) 0187bL
0190(0191aL) 0191bL	0191(0186aL) 0186bL	0192(0191aL) 0191bL	0193(0194aL) 0194bL	0194(0186aL) 0186bL	0195(0196aL) 0196bL	0196(0186aL) 0186bL	0197(0198aR) 0198bR	0198(0189aR) 0189bR	0199(0200aR) 0199bR
0200(0201aR) 0201bR	0201(0202aL) 0197bR	0202(0203bR) 0203bR	0203(0205aR) 0205bR	0204(0200aR) 0197bR	0205(0206aR) 0206bR	0206(0207aR) 0207bR	0207(0208aR) ERROR--	0208(0209aR) 0209bR	0209(0210aR) 0210bR
0210(0211aR) 0211bR	0211(0212bR) ERROR--	0212(0213bR) ERROR--	0213(0214aR) ERROR--	0214(0233aR) 0233bR	0215(0216aR) ERROR--	0216(0217bR) ERROR--	0217(0218bR) ERROR--	0218(0263aL) 0263bL	0219(0220aR) 0220bR
0220(0221aR) 0221bR	0221(0222aR) ERROR--	0222(0223aR) 0223bR	0223(0224aR) ERROR--	0224(0233aR) 0233bR	0225(0236aR) ERROR--	0226(0237aR) ERROR--	0227(0238aL) ERROR--	0228(0239aR) 0239bR	0229(0240aR) 0240bR
0230(0231aR) 0231bR	0231(0232aR) ERROR--	0232(0211aR) 0211bR	0233(0234aR) ERROR--	0234(0237aR) ERROR--	0235(0236aR) ERROR--	0236(0245aR) 0245bR	0237(0238aR) ERROR--	0238(0239aR) ERROR--	0239(0240aR) 0240bR
0240(0241bR) ERROR--	0241(0242bR) ERROR--	0242(0243bR) ERROR--	0243(0244aR) ERROR--	0244(0235aR) ERROR--	0245(0246aR) ERROR--	0246(0247aR) ERROR--	0247(0248aR) ERROR--	0248(0249aR) 0249bR	0249(0250aR) 0250bR
0250(0251aR) 0251bR	0251(0252aR) ERROR--	0252(0253aR) 0253bR	0253(0254aR) ERROR--	0254(0255aR) ERROR--	0255(0256aR) ERROR--	0256(0257aR) 0257bR	0257(0258aR) ERROR--	0258(0259aR) 0259bR	0259(0260aR) 0260bR
0260(0261aR) 0261bR	0261(0262aR) ERROR--	0262(0215aR) 0215bR	0263(0264aR) 0269bR	0264(0265aL) 0267bL	0265(0266aL) 0266bL	0266(0263aL) 0263bL	0267(0268aL) 0268bL	0268(0263aL) 0263bL	0269(0270aL) 0270bL
0270(0271aL) 0271bL	0271(0274aL) 0274bL	0272(0273aL) 0273bL	0273(0265aL) 0265bL	0274(0275aR) 0275bR	0275(0276aR) 0276bR	0276(0277bR) 0277bR	0277(0288aL) ERROR--	0278(0289aR) 0289bR	0279(0290aL) 0290bL
0280(0291aR) 0281bR	0291(0292aR) 0292bR	0292(0291aL) 0291bL	0293(0294aL) 0291bL	0294(0295aR) 0295aR	0295(0296aR) 0296bR	0296(0297bR) 0297bR	0297(0298bR) ERROR--	0298(0299bL) ERROR--	0299(0263aL) 0263bL
0300(0301aR) 0301bR	0301(0302aR) 0302bR	0302(0303aR) 0303bR	0303(0304aR) 0304bR	0304(0305aR) 0305bR	0305(0306aR) 0306bR	0306(0307aR) 0307aR	0307(0308aR) 0308aR	0308(0309aR) 0309aR	0309(0310aR) 0310bR
0310(0311aL) 0309bR	0311(0321aL) 0312bL	0312(0311aL) 0311bL	0313(0314aR) 0317bR	0314(0325aR) 0325bR	0315(0316aL) 0316bL	0316(0317aR) 0317aR	0317(0338aR) 0318bL	0318(0319aR) 0319aR	0319(0360aL) 0360bL
0320(0321aR) 0321bR	0321(0320aR) 0322aL	0322(0323aR) 0323aR	0323(0329aR) 0329bR	0324(0325aL) 0327aL	0325(0326aR) 0326aR	0326(0347aR) 0347bR	0327(0328aR) 0328aR	0328(0338aR) 0338bR	0329(0330aR) 0333bR
0330(0331aL) 0329bR	0331(0320aR) 0322aL	0332(0323aR) 0323aR	0333(0334aR) 0334bR	0334(0335aR) 0335bR	0335(0347aR) 0347bR	0336(0337aR) 0337aR	0337(0338aR) 0338bR	0338(0339aR) 0339bR	0339(0340aL) 0342bL
0340(0341aR) 0341bR	0341(0320aR) 0320bR	0342(0343bR) 0343bR	0343(0329aR) 0329bR	0344(0345aR) 0345bR	0345(0346aR) 0346bR	0346(0347aR) 0347bR	0347(0348aR) 0348aR	0348(0349aL) 0349bL	0349(0350aR) 0355bR
0350(0351aL) 0353bL	0351(0352aL) 0352bL	0352(0349aL) 0349bL	0353(0354aR) 0354bR	0354(0313aL) 0313bL	0355(0356aL) 0356bL	0356(0357aR) 0357aR	0357(0371aR) 0371bR	0358(0359aL) 0359bL	0359(0349aL) 0349bL
0360(0361aR) 0360bR	0361(0362aR) 0362bR	0362(0363aR) 0363bR	0363(0313aL) 0313bL	0364(0304aR) 0304bR	0365(0366aL) 0366bL	0366(0367aR) 0367aR	0367(0371aR) 0371bR	0368(0369aR) 0369bR	0369(0370aR) 0370bR
0370(0360aL) 0360bL	0371(0372aR) 0372bR	0372(0373aR) 0373bR	0373(0374bR) 0374bR	0374(0303aR) 0303bR	0375(0357aL) 0357bL	0376(0377aR) 0377aR	0377(0378aR) 0378aR	0378(0379aR) 0379aR	0379(0380bR) 0380bR
0380(0381aR) ERROR--	0381(0382aR) 0382bR	0382(0383aR) ERROR--	0383(0384aR) 0384bR	0384(0385aR) ERROR--	0385(0389aR) 0389bR	0386(0387aR) 0387bR	0387(0306aR) 0306bR	0388(0306aR) 0306bR	0389(0390aR) 0390bR
0390(0391bR) ERROR--	0391(0392aR) 0392bR	0392(0393aR) 0393bR	0393(0394aR) 0394bR	0394(0395aR) 0395bR	0395(0396aR) 0396bR	0396(0397aR) 0397aR	0397(0306aR) 0306bR	0398(0306aR) 0306bR	0399(0399aR) 0399bR
0400(0401bR) 2959aR	0401(0402bR) 3618aR	0402(0403bR) 2073aR	0403(0404bR) 2273aR	0404(0405bR) 1947aR	0405(0406bR) 3419aR	0406(0407bR) 2587aR	0407(0408bR) 2024aR	0408(0409bR) 2078aR	0409(0410bR) 2242aR
0410(0411bR) 2075aR	0411(0412bR) 2267aR	0412(0413bR) 2959aR	0413(0414bR) 3010aR	0414(0415bR) 3745aR	0415(0416bR) 3198aR	0416(0417bR) 2877aR	0417(0418bR) 3193aR	0418(0419bR) 2428aR	0419(0420bR) 2662aR
0420(0421bR) 2625aR	0421(0422bR) 3765aR	0422(0423bR) 2829aR	0423(0424bR) 3846aR	0424(0425bR) 2657aR	0425(0426bR) 3301aR	0426(0427bR) 2648aR	0427(0428bR) 3301aR	0428(0429bR) 2549aR	0429(0430bR) 2267aR
0430(0431bR) 2959aR	0431(0432bR) 3010aR	0432(0433bR) 3745aR	0433(0434bR) 2625aR	0434(0435bR) 1947aR	0435(0436bR) 3193aR	0436(0437bR) 2448aR	0437(0438bR) 2625aR	0438(0439bR) 2428aR	0439(0440bR) 3762aR
0440(0441bR) 2829aR	0441(0442bR) 3846aR	0442(0443bR) 2657aR	0443(0444bR) 3301aR	0444(0445bR) 2648aR	0445(0446bR) 3301aR	0446(0447bR) 2648aR	0447(0448bR) 3291aR	0448(0449bR) 2549aR	0449(0450bR) 3010aR
0450(0451bR) 2745aR	0451(0452bR) 3202aR	0452(0453bR) 3193aR	0453(0454bR) 1910aR	0454(0455bR) 3603aR	0455(0456bR) 3706aR	0456(0457bR) 2091aR	0457(0458bR) 3368aR	0458(0459bR) 2829aR	0459(0460bR) 2242aR
0460(0461bR) 2020aR	0461(0462bR) 3191aR	0462(0463bR) 1900aR	0463(0464bR) 1910aR	0464(0465bR) 2648aR	0465(0466bR) 3706aR	0466(0467bR) 2091aR	0467(0468bR) 3686aR	0468(0469bR) 2549aR	0469(0470bR) 2232aR
0470(0471bR) 2449aR	0471(0472bR) 2258aR	0472(0473bR) 2659aR	0473(0474bR) 2277aR	0474(0475bR) 2595aR	0475(0476bR) 3010aR	0476(0477bR) 2590aR	0477(0478bR) 2146aR	0478(0479bR) 2765aR	0479(0480bR) 3296aR
0480(0481bR) 2625aR	0481(0482bR) 3007aR	0482(0483bR) 2924aR	0483(0484bR) 1898aR	0484(0485bR) 2703aR	0485(0486bR) 3229aR	0486(0487bR) 3085aR	0487(0488bR) 1929aR	0488(0489bR) 2616aR	0489(0490bR) 3000aR
0490(0491bR) 2959aR	0491(0492bR) 3442aR	0492(0493bR) 2873aR	0493(0494bR) 2216aR	0494(0495bR) 1947aR	0495(0496bR) 3191aR	0496(0497bR) 2654aR	0497(0498bR) 1947aR	0498(0499bR) 2765aR	0499(0500bR) 3770aR
0500(0501bR) 2797aR	0501(0502bR) 3017aR	0502(0503bR) 2665aR	0503(0504bR) 3302aR	0504(0505bR) 2841aR	0505(0506bR) 3291aR	0506(0507bR) 2595aR	0507(0508bR) 2246aR	0508(0509bR) 2867aR	0509(0510bR) 1926aR
0510(0511bR) 2449aR	0511(0512bR) 3007aR	0512(0513bR) 2924aR	0513(0514bR) 1898aR	0514(0515bR) 2703aR	0515(0516bR) 3229aR	0516(0517bR) 3085aR	0517(0518bR) 1929aR	0518(0519bR) 2616aR	0519(0520bR) 3000aR
0520(0521bR) 2703aR	0521(0522bR) 1937aR	0522(0523bR) 1937aR	0523(0524bR) 3002aR	0524(0525bR) 2051aR	0525(0526bR) 2232aR	0526(0527bR) 3213aR	0527(0528bR) 2168aR	0528(0529bR) 2878aR	0529(0530bR) 2278aR
0530(0531bR) 2457aR	0531(0532bR) 3402aR	0532(0533bR) 2577aR	0533(0534bR) 3257aR	0534(0535bR) 2445aR	0535(0536bR) 3216aR	0536(0537bR) 32			

1090 (1091br 2061aR) 1091 (1092br 3867aR) 1092 (1093br 3121aR) 1093 (1094br 1978aR) 1094 (1095br 3669aR) 1095 (1096br 1910aR) 1096 (1097br 2974aR) 1097 (1098br 3591aR) 1098 (1099br 3081aR) 1099 (1100br 2367aR)
1100 (1101br 3092aR) 1101 (1102br 3407aR) 1102 (1103br 3221aR) 1103 (1104br 1914aR) 1104 (1105br 3768aR) 1105 (1106br 3264aR) 1106 (1107br 1945aR) 1107 (1108br 3857aR) 1108 (1109br 3828aR) 1109 (1110br 3698aR)
1110 (1111br 2337aR) 1111 (1112br 3179aR) 1112 (1113br 3179aR) 1113 (1114br 3179aR) 1114 (1115br 3179aR) 1115 (1116br 3179aR) 1116 (1117br 3179aR) 1117 (1118br 3179aR) 1118 (1119br 3179aR) 1119 (1120br 3179aR)
1120 (1121br 2061aR) 1121 (1122br 2061aR) 1122 (1123br 2061aR) 1123 (1124br 2061aR) 1124 (1125br 2061aR) 1125 (1126br 2061aR) 1126 (1127br 2061aR) 1127 (1128br 2061aR) 1128 (1129br 2061aR) 1129 (1130br 2061aR)
1130 (1131br 2061aR) 1131 (1132br 2061aR) 1132 (1133br 2061aR) 1133 (1134br 2061aR) 1134 (1135br 2061aR) 1135 (1136br 2061aR) 1136 (1137br 2061aR) 1137 (1138br 2061aR) 1138 (1139br 2061aR) 1139 (1140br 2061aR)
1140 (1141br 2061aR) 1141 (1142br 2061aR) 1142 (1143br 2061aR) 1143 (1144br 2061aR) 1144 (1145br 2061aR) 1145 (1146br 2061aR) 1146 (1147br 2061aR) 1147 (1148br 2061aR) 1148 (1149br 2061aR) 1149 (1150br 2061aR)
1150 (1151br 2133aR) 1151 (1152br 2133aR) 1152 (1153br 2590aR) 1153 (1154br 2590aR) 1154 (1155br 2590aR) 1155 (1156br 2590aR) 1156 (1157br 2590aR) 1157 (1158br 2590aR) 1158 (1159br 2590aR) 1159 (1160br 2590aR)
1160 (1161br 2133aR) 1161 (1162br 2133aR) 1162 (1163br 2590aR) 1163 (1164br 2590aR) 1164 (1165br 2590aR) 1165 (1166br 2590aR) 1166 (1167br 2590aR) 1167 (1168br 2590aR) 1168 (1169br 2590aR) 1169 (1170br 2590aR)
1170 (1171br 2133aR) 1171 (1172br 2133aR) 1172 (1173br 2590aR) 1173 (1174br 2590aR) 1174 (1175br 2590aR) 1175 (1176br 2590aR) 1176 (1177br 2590aR) 1177 (1178br 2590aR) 1178 (1179br 2590aR) 1179 (1180br 2590aR)
1180 (1181br 3747aR) 1181 (1182br 3747aR) 1182 (1183br 2042aR) 1183 (1184br 2042aR) 1184 (1185br 2042aR) 1185 (1186br 2042aR) 1186 (1187br 2042aR) 1187 (1188br 2042aR) 1188 (1189br 2042aR) 1189 (1190br 2042aR)
1190 (1191br 3381aR) 1191 (1192br 3558aR) 1192 (1193br 2068aR) 1193 (1194br 2268aR) 1194 (1195br 3745aR) 1195 (1196br 3326aR) 1196 (1197br 2969aR) 1197 (1198br 3769aR) 1198 (1199br 2136aR) 1199 (1200br 2558aR)
1200 (1201br 2068aR) 1201 (1202br 2068aR) 1202 (1203br 2068aR) 1203 (1204br 2068aR) 1204 (1205br 2068aR) 1205 (1206br 2068aR) 1206 (1207br 2068aR) 1207 (1208br 2068aR) 1208 (1209br 2068aR) 1209 (1210br 2068aR)
1210 (1211br 2829aR) 1211 (1212br 2011aR) 1212 (1213br 2532aR) 1213 (1214br 1990aR) 1214 (1215br 3745aR) 1215 (1216br 3265aR) 1216 (1217br 2878aR) 1217 (1218br 2068aR) 1218 (1219br 2136aR) 1219 (1220br 3846aR)
1220 (1221br 3100aR) 1221 (1222br 3554aR) 1222 (1223br 2643aR) 1223 (1224br 2279aR) 1224 (1225br 2645aR) 1225 (1226br 3326aR) 1226 (1227br 2878aR) 1227 (1228br 2606aR) 1228 (1229br 2606aR) 1229 (1230br 1401aR)
1230 (1231br 2068aR) 1231 (1232br 2068aR) 1232 (1233br 2068aR) 1233 (1234br 2068aR) 1234 (1235br 2068aR) 1235 (1236br 2068aR) 1236 (1237br 2068aR) 1237 (1238br 2068aR) 1238 (1239br 2068aR) 1239 (1240br 2068aR)
1240 (1241br 2705aR) 1241 (1242br 3987aR) 1242 (1243br 3121aR) 1243 (1244br 1992aR) 1244 (1245br 1992aR) 1245 (1246br 3548aR) 1246 (1247br 3246aR) 1247 (1248br 1992aR) 1248 (1249br 2605aR) 1249 (1250br 3801aR)
1250 (1251br 2833aR) 1251 (1252br 3867aR) 1252 (1253br 3129aR) 1253 (1254br 3768aR) 1254 (1255br 2788aR) 1255 (1256br 2267aR) 1256 (1257br 1901aR) 1257 (1258br 3871aR) 1258 (1259br 2124aR) 1259 (1260br 2374aR)
1260 (1261br 2833aR) 1261 (1262br 3867aR) 1262 (1263br 2723aR) 1263 (1264br 1194aR) 1264 (1265br 3061aR) 1265 (1266br 3061aR) 1266 (1267br 2183aR) 1267 (1268br 1976aR) 1268 (1269br 2669aR) 1269 (1270br 3368aR)
1270 (1271br 2593aR) 1271 (1272br 3867aR) 1272 (1273br 3121aR) 1273 (1274br 1992aR) 1274 (1275br 2788aR) 1275 (1276br 1919aR) 1276 (1277br 1901aR) 1277 (1278br 3871aR) 1278 (1279br 2124aR) 1279 (1280br 2374aR)
1280 (1281br 2531aR) 1281 (1282br 3867aR) 1282 (1283br 3788aR) 1283 (1284br 3334aR) 1284 (1285br 2747aR) 1285 (1286br 3406aR) 1286 (1287br 2785aR) 1287 (1288br 3761aR) 1288 (1289br 2590aR) 1289 (1290br 3281aR)
1290 (1291br 2795aR) 1291 (1292br 3867aR) 1292 (1293br 2889aR) 1293 (1294br 3867aR) 1294 (1295br 2889aR) 1295 (1296br 3624aR) 1296 (1297br 1947aR) 1297 (1298br 3867aR) 1298 (1299br 1940aR) 1299 (1300br 3624aR)
1300 (1301br 1947aR) 1301 (1302br 3867aR) 1302 (1303br 3484aR) 1303 (1304br 3553aR) 1304 (1305br 1947aR) 1305 (1306br 3304aR) 1306 (1307br 3381aR) 1307 (1308br 3761aR) 1308 (1309br 2590aR) 1309 (1310br 1909aR)
1310 (1311br 3769aR) 1311 (1312br 2276aR) 1312 (1313br 3215aR) 1313 (1314br 2303aR) 1314 (1315br 3059aR) 1315 (1316br 3059aR) 1316 (1317br 2647aR) 1317 (1318br 2281aR) 1318 (1319br 2796aR) 1319 (1320br 3264aR)
1320 (1321br 3215aR) 1321 (1322br 3304aR) 1322 (1323br 3169aR) 1323 (1324br 3711aR) 1324 (1325br 2223aR) 1325 (1326br 3030aR) 1326 (1327br 3097aR) 1327 (1328br 3864aR) 1328 (1329br 2971aR) 1329 (1330br 3814aR)
1330 (1331br 3769aR) 1331 (1332br 3710aR) 1332 (1333br 2774aR) 1333 (1334br 3362aR) 1334 (1335br 3059aR) 1335 (1336br 3804aR) 1336 (1337br 3246aR) 1337 (1338br 2281aR) 1338 (1339br 2966aR) 1339 (1340br 3248aR)
1340 (1341br 2879aR) 1341 (1342br 2506aR) 1342 (1343br 2605aR) 1343 (1344br 2378aR) 1344 (1345br 2637aR) 1345 (1346br 2383aR) 1346 (1347br 3221aR) 1347 (1348br 3319aR) 1348 (1349br 2667aR) 1349 (1350br 3710aR)
1350 (1351br 3769aR) 1351 (1352br 1919aR) 1352 (1353br 2900aR) 1353 (1354br 2772aR) 1354 (1355br 2124aR) 1355 (1356br 2267aR) 1356 (1357br 1956aR) 1357 (1358br 3036aR) 1358 (1359br 3246aR) 1359 (1360br 3025aR)
1360 (1361br 2647aR) 1361 (1362br 2358aR) 1362 (1363br 2604aR) 1363 (1364br 2378aR) 1364 (1365br 2875aR) 1365 (1366br 2506aR) 1366 (1367br 1932aR) 1367 (1368br 3036aR) 1368 (1369br 2645aR) 1369 (1370br 3120aR)
1370 (1371br 2867aR) 1371 (1372br 3706aR) 1372 (1373br 3753aR) 1373 (1374br 3773aR) 1374 (1375br 2704aR) 1375 (1376br 2744aR) 1376 (1377br 1945aR) 1377 (1378br 3382aR) 1378 (1379br 2899aR) 1379 (1380br 2362aR)
1380 (1381br 2795aR) 1381 (1382br 2068aR) 1382 (1383br 2828aR) 1383 (1384br 3406aR) 1384 (1385br 3221aR) 1385 (1386br 3391aR) 1386 (1387br 2867aR) 1387 (1388br 3036aR) 1388 (1389br 3704aR) 1389 (1400br 2400aR)
1390 (1391br 3779aR) 1391 (1392br 3688aR) 1392 (1393br 2828aR) 1393 (1394br 3773aR) 1394 (1395br 2704aR) 1395 (1396br 3391aR) 1396 (1397br 2867aR) 1397 (1398br 3036aR) 1398 (1399br 2899aR) 1399 (1400br 2400aR)
1400 (1401br 2844aR) 1401 (1402br 2214aR) 1402 (1403br 3603aR) 1403 (1404br 3842aR) 1404 (1405br 2677aR) 1405 (1406br 3710aR) 1406 (1407br 2867aR) 1407 (1408br 3450aR) 1408 (1409br 2878aR) 1409 (1410br 2505aR)
1410 (1411br 1940aR) 1411 (1412br 3179aR) 1412 (1413br 3179aR) 1413 (1414br 3179aR) 1414 (1415br 3179aR) 1415 (1416br 3179aR) 1416 (1417br 3179aR) 1417 (1418br 3179aR) 1418 (1419br 3179aR) 1419 (1420br 3179aR)
1420 (1421br 3311aR) 1421 (1422br 2008aR) 1422 (1423br 3169aR) 1423 (1424br 3324aR) 1424 (1425br 3798aR) 1425 (1426br 2306aR) 1426 (1427br 2413aR) 1427 (1428br 2608aR) 1428 (1429br 2878aR) 1429 (1430br 2826aR)
1430 (1431br 3717aR) 1431 (1432br 3522aR) 1432 (1433br 3473aR) 1433 (1434br 3278aR) 1434 (1435br 3056aR) 1435 (1436br 2418aR) 1436 (1437br 3628aR) 1437 (1438br 3265aR) 1438 (1439br 2669aR) 1439 (1440br 2176aR)
1440 (1441br 3769aR) 1441 (1442br 3769aR) 1442 (1443br 3769aR) 1443 (1444br 3769aR) 1444 (1445br 3769aR) 1445 (1446br 3769aR) 1446 (1447br 3769aR) 1447 (1448br 3769aR) 1448 (1449br 3769aR) 1449 (1450br 3769aR)
1450 (1451br 2133aR) 1451 (1452br 1919aR) 1452 (1453br 1124aR) 1453 (1454br 3226aR) 1454 (1455br 3413aR) 1455 (1456br 1977aR) 1456 (1457br 2707aR) 1457 (1458br 3261aR) 1458 (1459br 2021aR) 1459 (1460br 2378aR)
1460 (1461br 2666aR) 1461 (1462br 2262aR) 1462 (1463br 1903aR) 1463 (1464br 1992aR) 1464 (1465br 2123aR) 1465 (1466br 3914aR) 1466 (1467br 2017aR) 1467 (1468br 3914aR) 1468 (1469br 2021aR) 1469 (1470br 3000aR)
1470 (1471br 2666aR) 1471 (1472br 2262aR) 1472 (1473br 1903aR) 1473 (1474br 1992aR) 1474 (1475br 2123aR) 1475 (1476br 3914aR) 1476 (1477br 2017aR) 1477 (1478br 3914aR) 1478 (1479br 2021aR) 1479 (1480br 3000aR)
1480 (1481br 3406aR) 1481 (1482br 2937aR) 1482 (1483br 1947aR) 1483 (1484br 3422aR) 1484 (1485br 2123aR) 1485 (1486br 3265aR) 1486 (1487br 3116aR) 1487 (1488br 1913aR) 1488 (1489br 2021aR) 1489 (1490br 3857aR)
1490 (1491br 2111aR) 1491 (1492br 3432aR) 1492 (1493br 2577aR) 1493 (1494br 3278aR) 1494 (1495br 2666aR) 1495 (1496br 1919aR) 1496 (1497br 3764aR) 1497 (1498br 2402aR) 1498 (1499br 3169aR) 1499 (1500br 2578aR)
1500 (1501br 1913aR) 1501 (1502br 3366aR) 1502 (1503br 2969aR) 1503 (1504br 3366aR) 1504 (1505br 2969aR) 1505 (1506br 1919aR) 1506 (1507br 1901aR) 1507 (1508br 3764aR) 1508 (1509br 3169aR) 1509 (1510br 3366aR)
1510 (1511br 3785aR) 1511 (1512br 3362aR) 1512 (1513br 2008aR) 1513 (1514br 2276aR) 1514 (1515br 2921aR) 1515 (1516br 3522aR) 1516 (1517br 2959aR) 1517 (1518br 2402aR) 1518 (1519br 2828aR) 1519 (1520br 3557aR)
1520 (1521br 1933aR) 1521 (1522br 2263aR) 1522 (1523br 2628aR) 1523 (1524br 3074aR) 1524 (1525br 2692aR) 1525 (1526br 1917aR) 1526 (1527br 2633aR) 1527 (1528br 3768aR) 1528 (1529br 3177aR) 1529 (1530br 2227aR)
1530 (1531br 2666aR) 1531 (1532br 2666aR) 1532 (1533br 2666aR) 1533 (1534br 2666aR) 1534 (1535br 2666aR) 1535 (1536br 2666aR) 1536 (1537br 2666aR) 1537 (1538br 2666aR) 1538 (1539br 2666aR) 1539 (1540br 2666aR)
1540 (1541br 1937aR) 1541 (1542br 3767aR) 1542 (1543br 2133aR) 1543 (1544br 1970aR) 1544 (1545br 2898aR) 1545 (1546br 2258aR) 1546 (1547br 3603aR) 1547 (1548br 3768aR) 1548 (1549br 3177aR) 1549 (1550br 2493aR)
1550 (1551br 2636aR) 1551 (1552br 3576aR) 1552 (1553br 2460aR) 1553 (1554br 3573aR) 1554 (1555br 1947aR) 1555 (1556br 3378aR) 1556 (1557br 3079aR) 1557 (1558br 3265aR) 1558 (1559br 2796aR) 1559 (1560br 3263aR)
1560 (1561br 3091aR) 1561 (1562br 1937aR) 1562 (1563br 2696aR) 1563 (1564br 3637aR) 1564 (1565br 2875aR) 1565 (1566br 2506aR) 1566 (1567br 1932aR) 1567 (1568br 3036aR) 1568 (1569br 2645aR) 1569 (1570br 3120aR)
1570 (1571br 2829aR) 1571 (1572br 1968aR) 1572 (1573br 2460aR) 1573 (1574br 2696aR) 1574 (1575br 2875aR) 1575 (1576br 3378aR) 1576 (1577br 3079aR) 1577 (1578br 3265aR) 1578 (1579br 2796aR) 1579 (1580br 3120aR)
1580 (1581br 2671aR) 1581 (1582br 2359aR) 1582 (1583br 1937aR) 1583 (1584br 3077aR) 1584 (1585br 3221aR) 1585 (1586br 3548aR) 1586 (1587br 2867aR) 1587 (1588br 3518aR) 1588 (1589br 3753aR) 1589 (1590br 2002aR)
1590 (1591br 2966aR) 1591 (1592br 3494aR) 1592 (1593br 2068aR) 1593 (1594br 3795aR) 1594 (1595br 2704aR) 1595 (1596br 3391aR) 1596 (1597br 2867aR) 1597 (1598br 3036aR) 1598 (1599br 2899aR) 1599 (1600br 2400aR)
1600 (1601br 1949aR) 1601 (1602br 3682aR) 1602 (1603br 3755aR) 1603 (1604br 2303aR) 1604 (1605br 3221aR) 1605 (1606br 3384aR) 1606 (1607br 2113aR) 1607 (1608br 3450aR) 1608 (1609br 2878aR) 1609 (1610br 3544aR)
1610 (1611br 3381aR) 1611 (1612br 3558aR) 1612 (1613br 3567aR) 1613 (1614br 3291aR) 1614 (1615br 2443aR) 1615 (1616br 2522aR) 1616 (1617br 2927aR) 1617 (1618br 3265aR) 1618 (1619br 2013aR) 1619 (1620br 3402aR)
1620 (1621br 2068aR) 1621 (1622br 2068aR) 1622 (1623br 2068aR) 1623 (1624br 2068aR) 1624 (1625br 2068aR) 1625 (1626br 2068aR) 1626 (1627br 2068aR) 1627 (1628br 2068aR) 1628 (1629br 2068aR) 1629 (1630br 2068aR)
1630 (1631br 3757aR) 1631 (1632br 1958aR) 1632 (1633br 2636aR) 1633 (1634br 3291aR) 1634 (1635br 3473aR) 1635 (1636br 2277aR) 1636 (1637br 2628aR) 1637 (1638br 3264aR) 1638 (1639br 2413aR) 1639 (1640br 2162aR)
1640 (1641br 3476aR) 1641 (1642br 1919aR) 1642 (1643br 2593aR) 1643 (1644br 2276aR) 1644 (1645br 2867aR) 1645 (1646br 3176aR) 1646 (1647br 1950aR) 1647 (1648br 3256aR) 1648 (1649br 1947aR) 1649 (1650br 3032aR)
1650 (1651br 2911aR) 1651 (1652br 2911aR) 1652 (1653br 2911aR) 1653 (1654br 2911aR) 1654 (1655br 2911aR) 1655 (1656br 2911aR) 1656 (1657br 2911aR) 1657 (1658br 2911aR) 1658 (1659br 2911aR) 1659 (1660br 2911aR)
1660 (1661br 1950aR) 1661 (1662br 3256aR) 1662 (1663br 1947aR) 1663 (1664br 2066aR) 1664 (1665br 2867aR) 1665 (1666br 3768aR) 1666 (1667br 3116aR) 1667 (1668br 3256aR) 1668 (1669br 1947aR) 1669 (1670br 3032aR)
1670 (1671br 2911aR) 1671 (1672br 1919aR) 1672 (1673br 3209aR) 1673 (1674br 3016aR) 1674 (1675br 3016aR) 1675 (1676br 3704aR) 1676 (1677br 2091aR) 1677 (1678br 3767aR) 1678 (1679br 1903aR) 1679 (1680br 1963aR)
1680 (1681br 2911aR) 1681 (1682br 1919aR) 1682 (1683br 3209aR) 1683 (1684br 3016aR) 1684 (1685br 3016aR) 1685 (1686br 3704aR) 1686 (1687br 2091aR) 1687 (1688br 3767aR) 1688 (1689br 1903aR) 1689 (1690br 1963aR)
1690 (1691br 2601aR) 1691 (1692br 3511aR) 1692 (1693br 1937aR) 1693 (1694br 3016aR) 1694 (1695br 3097aR) 1695 (1696br 1913aR) 1696 (1697br 2065aR) 1697 (1698br 2065aR) 1698 (1699br 2631aR) 1699 (1700br 3698aR)
1700 (1701br 3215aR) 1701 (1702br 1917aR) 1702 (1703br 2645aR) 1703 (1704br 2819aR) 1704 (1705br 1935aR) 1705 (1706br 3265aR) 1706 (1707br 2133aR) 1707 (1708br 1921aR) 1708 (1709br 1945aR) 1709 (1710br 3030aR)
1710 (1711br 2068aR) 1711 (1712br 2068aR) 1712 (1713br 2068aR) 1713 (1714br 2068aR) 1714 (1715br 2068aR) 1715 (1716br 2068aR) 1716 (1717br 2068aR) 1717 (1718br 2068aR) 1718 (1719br 2068aR) 1719 (1720br 2068aR)
1720 (1721br 360aR) 1721 (1722br 2040aR) 1722 (1723br 2797aR) 1723 (1724br 3432aR) 1724 (1725br 3121aR) 1725 (1726br 1985aR) 1726 (1727br 2647aR) 1727 (1728br 3121aR) 1728 (1729br 2011aR) 1729 (1730br 3767aR)
1730 (1731br 2590aR) 1731 (1732br 2234aR) 1732 (1733br 2123aR) 1733 (1734br 3867aR) 1734 (1735br 3121aR) 1735 (1736br 2101aR) 1736 (1737br 3628aR) 1737 (1738br 3265aR) 1738 (1739br 2669aR) 1739 (1740br 3544aR)
1740

2620 (2621br|3753ar) 2621 (2622br|3702ar) 2622 (2623br|3460ar) 2623 (2624br|2936ar) 2624 (2625br|3215ar) 2625 (2626br|2002ar) 2626 (2627br|2647ar) 2627 (2628br|2936ar) 2628 (2629br|1939ar) 2629 (2630br|2238ar)
2630 (2631br|2601ar) 2631 (2632br|3291ar) 2632 (2633br|2051ar) 2633 (2634br|3511ar) 2634 (2635br|2447ar) 2635 (2636br|1881ar) 2636 (2637br|3097ar) 2637 (2638br|2246ar) 2638 (2639br|2867ar) 2639 (2640br|3698ar)
2640 (2641br|3631ar) 2641 (2642br|2924ar) 2642 (2643br|3365ar) 2643 (2644br|3956ar) 2644 (2645br|2755ar) 2645 (2646br|3834ar) 2646 (2647br|2700ar) 2647 (2648br|2266ar) 2648 (2649br|2011ar) 2649 (2650br|3545ar)
2650 (2651br|1933ar) 2651 (2652br|3002ar) 2652 (2653br|2625ar) 2653 (2654br|3365ar) 2654 (2655br|2647ar) 2655 (2656br|1984ar) 2656 (2657br|2700ar) 2657 (2658br|2266ar) 2658 (2659br|2817ar) 2659 (2660br|2239ar)
2660 (2661br|3555ar) 2661 (2662br|3688ar) 2662 (2663br|2532ar) 2663 (2664br|3511ar) 2664 (2665br|2447ar) 2665 (2666br|3406ar) 2666 (2667br|2607ar) 2667 (2668br|2246ar) 2668 (2669br|2131ar) 2669 (2670br|3174ar)
2670 (2671br|2455ar) 2671 (2672br|2025ar) 2672 (2673br|2025ar) 2673 (2674br|1975ar) 2674 (2675br|2891ar) 2675 (2676br|3698ar) 2676 (2677br|2187ar) 2677 (2678br|3524ar) 2678 (2679br|3524ar) 2679 (2680br|1961ar)
2680 (2681br|3087ar) 2681 (2682br|2002ar) 2682 (2683br|2820ar) 2683 (2684br|3257ar) 2684 (2685br|1939ar) 2685 (2686br|2025ar) 2686 (2687br|2627ar) 2687 (2688br|2246ar) 2688 (2689br|2187ar) 2689 (2690br|3174ar)
2690 (2691br|2697ar) 2691 (2692br|3511ar) 2692 (2693br|1939ar) 2693 (2694br|2002ar) 2694 (2695br|2891ar) 2695 (2696br|3257ar) 2696 (2697br|1939ar) 2697 (2698br|2246ar) 2698 (2699br|2187ar) 2699 (2700br|3174ar)
2700 (2701br|2790ar) 2701 (2702br|2790ar) 2702 (2703br|2790ar) 2703 (2704br|1985ar) 2704 (2705br|2787ar) 2705 (2706br|3698ar) 2706 (2707br|2787ar) 2707 (2708br|3698ar) 2708 (2709br|3698ar) 2709 (2710br|3698ar)
2710 (2711br|2115ar) 2711 (2712br|3757ar) 2712 (2713br|3044ar) 2713 (2714br|2002ar) 2714 (2715br|2113ar) 2715 (2716br|3256ar) 2716 (2717br|2790ar) 2717 (2718br|2246ar) 2718 (2719br|3564ar) 2719 (2720br|3871ar)
2720 (2721br|1939ar) 2721 (2722br|1919ar) 2722 (2723br|3044ar) 2723 (2724br|2984ar) 2724 (2725br|1939ar) 2725 (2726br|2787ar) 2726 (2727br|2788ar) 2727 (2728br|1958ar) 2728 (2729br|2828ar) 2729 (2730br|3871ar)
2730 (2731br|3087ar) 2731 (2732br|3591ar) 2732 (2733br|3065ar) 2733 (2734br|1985ar) 2734 (2735br|2787ar) 2735 (2736br|3698ar) 2736 (2737br|2647ar) 2737 (2738br|3374ar) 2738 (2739br|2703ar) 2739 (2740br|2423ar)
2740 (2741br|2596ar) 2741 (2742br|3659ar) 2742 (2743br|2764ar) 2743 (2744br|3205ar) 2744 (2745br|1889ar) 2745 (2746br|2647ar) 2746 (2747br|2647ar) 2747 (2748br|2647ar) 2748 (2749br|2647ar) 2749 (2750br|3801ar)
2750 (2751br|1950ar) 2751 (2752br|3029ar) 2752 (2753br|1949ar) 2753 (2754br|1918ar) 2754 (2755br|2828ar) 2755 (2756br|2868ar) 2756 (2757br|2649ar) 2757 (2758br|2392ar) 2758 (2759br|3662ar) 2759 (2760br|3116ar)
2760 (2761br|2601ar) 2761 (2762br|3659ar) 2762 (2763br|3053ar) 2763 (2764br|3402ar) 2764 (2765br|2778ar) 2765 (2766br|3402ar) 2766 (2767br|2778ar) 2767 (2768br|3374ar) 2768 (2769br|2768ar) 2769 (2770br|2423ar)
2770 (2771br|1949ar) 2771 (2772br|3494ar) 2772 (2773br|1949ar) 2773 (2774br|3402ar) 2774 (2775br|2828ar) 2775 (2776br|2868ar) 2776 (2777br|2649ar) 2777 (2778br|2392ar) 2778 (2779br|3662ar) 2779 (2780br|3116ar)
2780 (2781br|2891ar) 2781 (2782br|3030ar) 2782 (2783br|2901ar) 2783 (2784br|2258ar) 2784 (2785br|3065ar) 2785 (2786br|2788ar) 2786 (2787br|2116ar) 2787 (2788br|3374ar) 2788 (2789br|2528ar) 2789 (2790br|3578ar)
2790 (2791br|3437ar) 2791 (2792br|3306ar) 2792 (2793br|3065ar) 2793 (2794br|2001ar) 2794 (2795br|2788ar) 2795 (2796br|3768ar) 2796 (2797br|3747ar) 2797 (2798br|2167ar) 2798 (2799br|2703ar) 2799 (2800br|2358ar)
2800 (2801br|1939ar) 2801 (2802br|2294ar) 2802 (2803br|3440ar) 2803 (2804br|2490ar) 2804 (2805br|2124ar) 2805 (2806br|3457ar) 2806 (2807br|2570ar) 2807 (2808br|3374ar) 2808 (2809br|1928ar) 2809 (2810br|2938ar)
2810 (2811br|2091ar) 2811 (2812br|3861ar) 2812 (2813br|2113ar) 2813 (2814br|3606ar) 2814 (2815br|3620ar) 2815 (2816br|3858ar) 2816 (2817br|2084ar) 2817 (2818br|3320ar) 2818 (2819br|2057ar) 2819 (2820br|3496ar)
2820 (2821br|2585ar) 2821 (2822br|2342ar) 2822 (2823br|3158ar) 2823 (2824br|2169ar) 2824 (2825br|2625ar) 2825 (2826br|3513ar) 2826 (2827br|1989ar) 2827 (2828br|2239ar) 2828 (2829br|3340ar) 2829 (2830br|1961ar)
2830 (2831br|3056ar) 2831 (2832br|2889ar) 2832 (2833br|2116ar) 2833 (2834br|2170ar) 2834 (2835br|3620ar) 2835 (2836br|3858ar) 2836 (2837br|2113ar) 2837 (2838br|3320ar) 2838 (2839br|2577ar) 2839 (2840br|2377ar)
2840 (2841br|2092ar) 2841 (2842br|2170ar) 2842 (2843br|2625ar) 2843 (2844br|3858ar) 2844 (2845br|2113ar) 2845 (2846br|3798ar) 2846 (2847br|2846ar) 2847 (2848br|2239ar) 2848 (2849br|2601ar) 2849 (2850br|3857ar)
2850 (2851br|3121ar) 2851 (2852br|2215ar) 2852 (2853br|2605ar) 2853 (2854br|2244ar) 2854 (2855br|2593ar) 2855 (2856br|3448ar) 2856 (2857br|3215ar) 2857 (2858br|3304ar) 2858 (2859br|3340ar) 2859 (2860br|2937ar)
2860 (2861br|1947ar) 2861 (2862br|3419ar) 2862 (2863br|2625ar) 2863 (2864br|3513ar) 2864 (2865br|2828ar) 2865 (2866br|1989ar) 2866 (2867br|2867ar) 2867 (2868br|2239ar) 2868 (2869br|2601ar) 2869 (2870br|3857ar)
2870 (2871br|2866ar) 2871 (2872br|1913ar) 2872 (2873br|2636ar) 2873 (2874br|1918ar) 2874 (2875br|2843ar) 2875 (2876br|2942ar) 2876 (2877br|2123ar) 2877 (2878br|3914ar) 2878 (2879br|2829ar) 2879 (2880br|2346ar)
2880 (2881br|2660ar) 2881 (2882br|3402ar) 2882 (2883br|2654ar) 2883 (2884br|3283ar) 2884 (2885br|2436ar) 2885 (2886br|3391ar) 2886 (2887br|3764ar) 2887 (2888br|2928ar) 2888 (2889br|2247ar) 2889 (2890br|2247ar)
2890 (2891br|3669ar) 2891 (2892br|1906ar) 2892 (2893br|3477ar) 2893 (2894br|3162ar) 2894 (2895br|3092ar) 2895 (2896br|3036ar) 2896 (2897br|3266ar) 2897 (2898br|1957ar) 2898 (2899br|2836ar) 2899 (2900br|2241ar)
2900 (2901br|2601ar) 2901 (2902br|1978ar) 2902 (2903br|1978ar) 2903 (2904br|2133ar) 2904 (2905br|2924ar) 2905 (2906br|3009ar) 2906 (2907br|3464ar) 2907 (2908br|2175ar) 2908 (2909br|2899ar) 2909 (2910br|2241ar)
2910 (2911br|3669ar) 2911 (2912br|3669ar) 2912 (2913br|3669ar) 2913 (2914br|3669ar) 2914 (2915br|3669ar) 2915 (2916br|3669ar) 2916 (2917br|3669ar) 2917 (2918br|3669ar) 2918 (2919br|3669ar) 2919 (2920br|3669ar)
2920 (2921br|3669ar) 2921 (2922br|1978ar) 2922 (2923br|1978ar) 2923 (2924br|2133ar) 2924 (2925br|2924ar) 2925 (2926br|3009ar) 2926 (2927br|3464ar) 2927 (2928br|2175ar) 2928 (2929br|2899ar) 2929 (2930br|2241ar)
2930 (2931br|3297ar) 2931 (2932br|3687ar) 2932 (2933br|2643ar) 2933 (2934br|3392ar) 2934 (2935br|2948ar) 2935 (2936br|2270ar) 2936 (2937br|2081ar) 2937 (2938br|3447ar) 2938 (2939br|2462ar) 2939 (2940br|3221ar)
2940 (2941br|2844ar) 2941 (2942br|2844ar) 2942 (2943br|2844ar) 2943 (2944br|2844ar) 2944 (2945br|2844ar) 2945 (2946br|2844ar) 2946 (2947br|2844ar) 2947 (2948br|2844ar) 2948 (2949br|2844ar) 2949 (2950br|2844ar)
2950 (2951br|3891ar) 2951 (2952br|3431ar) 2952 (2953br|2645ar) 2953 (2954br|3329ar) 2954 (2955br|2948ar) 2955 (2956br|1978ar) 2956 (2957br|2081ar) 2957 (2958br|3447ar) 2958 (2959br|2462ar) 2959 (2960br|3221ar)
2960 (2961br|2699ar) 2961 (2962br|3768ar) 2962 (2963br|3755ar) 2963 (2964br|3768ar) 2964 (2965br|2136ar) 2965 (2966br|3554ar) 2966 (2967br|2596ar) 2967 (2968br|2453ar) 2968 (2969br|1957ar) 2969 (2970br|3314ar)
2970 (2971br|3777ar) 2971 (2972br|3777ar) 2972 (2973br|3777ar) 2973 (2974br|3777ar) 2974 (2975br|3777ar) 2975 (2976br|3777ar) 2976 (2977br|3777ar) 2977 (2978br|3777ar) 2978 (2979br|3777ar) 2979 (2980br|3777ar)
2980 (2981br|3439ar) 2981 (2982br|3768ar) 2982 (2983br|2865ar) 2983 (2984br|3768ar) 2984 (2985br|1950ar) 2985 (2986br|3591ar) 2986 (2987br|3147ar) 2987 (2988br|2453ar) 2988 (2989br|3052ar) 2989 (2990br|2361ar)
2990 (2991br|2867ar) 2991 (2992br|3257ar) 2992 (2993br|2878ar) 2993 (2994br|2867ar) 2994 (2995br|2716ar) 2995 (2996br|3553ar) 2996 (2997br|1933ar) 2997 (2998br|3455ar) 2998 (2999br|3764ar) 2999 (3000br|2470ar)
3000 (3001br|2660ar) 3001 (3002br|1919ar) 3002 (3003br|2878ar) 3003 (3004br|2984ar) 3004 (3005br|1937ar) 3005 (3006br|3402ar) 3006 (3007br|2787ar) 3007 (3008br|3314ar) 3008 (3009br|3314ar) 3009 (3010br|2754ar)
3010 (3011br|2073ar) 3011 (3012br|2012ar) 3012 (3013br|2436ar) 3013 (3014br|2937ar) 3014 (3015br|2540ar) 3015 (3016br|3264ar) 3016 (3017br|3180ar) 3017 (3018br|3180ar) 3018 (3019br|3180ar) 3019 (3020br|2257ar)
3020 (3021br|2089ar) 3021 (3022br|3846ar) 3022 (3023br|3436ar) 3023 (3024br|2011ar) 3024 (3025br|2457ar) 3025 (3026br|2625ar) 3026 (3027br|2625ar) 3027 (3028br|2382ar) 3028 (3029br|2011ar) 3029 (3030br|1969ar)
3030 (3031br|2441ar) 3031 (3032br|2441ar) 3032 (3033br|2441ar) 3033 (3034br|2441ar) 3034 (3035br|2441ar) 3035 (3036br|2441ar) 3036 (3037br|2441ar) 3037 (3038br|2441ar) 3038 (3039br|2441ar) 3039 (3040br|2441ar)
3040 (3041br|2115ar) 3041 (3042br|2333ar) 3042 (3043br|3861ar) 3043 (3044br|3861ar) 3044 (3045br|2857ar) 3045 (3046br|2635ar) 3046 (3047br|2635ar) 3047 (3048br|2635ar) 3048 (3049br|2635ar) 3049 (3050br|3682ar)
3050 (3051br|3765ar) 3051 (3052br|2920ar) 3052 (3053br|1950ar) 3053 (3054br|2257ar) 3054 (3055br|2696ar) 3055 (3056br|3570ar) 3056 (3057br|3460ar) 3057 (3058br|2936ar) 3058 (3059br|1944ar) 3059 (3060br|2725ar)
3060 (3061br|2780ar) 3061 (3062br|2780ar) 3062 (3063br|2780ar) 3063 (3064br|2780ar) 3064 (3065br|2780ar) 3065 (3066br|2780ar) 3066 (3067br|2780ar) 3067 (3068br|2780ar) 3068 (3069br|2780ar) 3069 (3070br|2780ar)
3070 (3071br|2113ar) 3071 (3072br|2333ar) 3072 (3073br|2867ar) 3073 (3074br|2867ar) 3074 (3075br|3088ar) 3075 (3076br|2493ar) 3076 (3077br|2073ar) 3077 (3078br|2333ar) 3078 (3079br|1944ar) 3079 (3080br|2237ar)
3080 (3081br|1949ar) 3081 (3082br|2267ar) 3082 (3083br|1897ar) 3083 (3084br|3682ar) 3084 (3085br|3448ar) 3085 (3086br|3000ar) 3086 (3087br|3329ar) 3087 (3088br|2338ar) 3088 (3089br|3651ar) 3089 (3090br|3748ar)
3090 (3091br|3748ar) 3091 (3092br|3748ar) 3092 (3093br|3748ar) 3093 (3094br|3748ar) 3094 (3095br|3748ar) 3095 (3096br|3748ar) 3096 (3097br|3748ar) 3097 (3098br|3748ar) 3098 (3099br|3748ar) 3099 (3100br|3748ar)
3100 (3101br|2625ar) 3101 (3102br|3769ar) 3102 (3103br|1113ar) 3103 (3104br|2936ar) 3104 (3105br|2936ar) 3105 (3106br|3841ar) 3106 (3107br|1924ar) 3107 (3108br|2928ar) 3108 (3109br|3601ar) 3109 (3110br|2362ar)
3110 (3111br|3567ar) 3111 (3112br|2265ar) 3112 (3113br|2081ar) 3113 (3114br|3301ar) 3114 (3115br|2667ar) 3115 (3116br|3470ar) 3116 (3117br|3052ar) 3117 (3118br|2920ar) 3118 (3119br|1958ar) 3119 (3120br|2179ar)
3120 (3121br|2265ar) 3121 (3122br|2265ar) 3122 (3123br|2265ar) 3123 (3124br|2265ar) 3124 (3125br|2265ar) 3125 (3126br|2265ar) 3126 (3127br|2265ar) 3127 (3128br|2265ar) 3128 (3129br|2265ar) 3129 (3130br|2265ar)
3130 (3131br|2579ar) 3131 (3132br|1990ar) 3132 (3133br|1113ar) 3133 (3134br|2936ar) 3134 (3135br|2936ar) 3135 (3136br|3769ar) 3136 (3137br|3769ar) 3137 (3138br|2928ar) 3138 (3139br|1958ar) 3139 (3140br|2493ar)
3140 (3141br|2067ar) 3141 (3142br|3033ar) 3142 (3143br|2404ar) 3143 (3144br|2984ar) 3144 (3145br|1903ar) 3145 (3146br|1983ar) 3146 (3147br|3091ar) 3147 (3148br|2258ar) 3148 (3149br|2133ar) 3149 (3150br|2216ar)
3150 (3151br|2660ar) 3151 (3152br|2660ar) 3152 (3153br|2660ar) 3153 (3154br|2660ar) 3154 (3155br|2660ar) 3155 (3156br|2660ar) 3156 (3157br|2660ar) 3157 (3158br|2660ar) 3158 (3159br|2660ar) 3159 (3160br|2257ar)
3160 (3161br|2121ar) 3161 (3162br|3682ar) 3162 (3163br|3436ar) 3163 (3164br|2984ar) 3164 (3165br|1903ar) 3165 (3166br|1983ar) 3166 (3167br|3091ar) 3167 (3168br|2258ar) 3168 (3169br|2133ar) 3169 (3170br|3688ar)
3170 (3171br|3717ar) 3171 (3172br|3365ar) 3172 (3173br|2785ar) 3173 (3174br|3868ar) 3174 (3175br|3777ar) 3175 (3176br|2916ar) 3176 (3177br|2462ar) 3177 (3178br|2257ar) 3178 (3179br|2603ar) 3179 (3180br|3824ar)
3180 (3181br|2073ar) 3181 (3182br|2073ar) 3182 (3183br|2073ar) 3183 (3184br|2073ar) 3184 (3185br|2073ar) 3185 (3186br|2073ar) 3186 (3187br|2073ar) 3187 (3188br|2073ar) 3188 (3189br|2073ar) 3189 (3190br|2073ar)
3190 (3191br|1939ar) 3191 (3192br|3295ar) 3192 (3193br|3097ar) 3193 (3194br|2002ar) 3194 (3195br|2867ar) 3195 (3196br|2891ar) 3196 (3197br|2891ar) 3197 (3198br|2891ar) 3198 (3199br|2891ar) 3199 (3200br|2891ar)
3200 (3201br|1939ar) 3201 (3202br|2493ar) 3202 (3203br|1939ar) 3203 (3204br|2267ar) 3204 (3205br|2447ar) 3205 (3206br|1981ar) 3206 (3207br|2457ar) 3207 (3208br|2011ar) 3208 (3209br|2511ar) 3209 (3210br|3622ar)
3210 (3211br|3174ar) 3211 (3212br|1919ar) 3212 (3213br|3100ar) 3213 (3214br|3493ar) 3214 (3215br|2441ar) 3215 (3216br|3402ar) 3216 (3217br|3747ar) 3217 (3218br|3747ar) 3218 (3219br|3747ar) 3219 (3220br|3747ar)
3220 (3221br|3439ar) 3221 (3222br|1917ar) 3222 (3223br|2073ar) 3223 (3224br|2267ar) 3224 (3225br|2447ar) 3225 (3226br|1981ar) 3226 (3227br|2459ar) 3227 (3228br|3301ar) 3228 (3229br|2856ar) 3229 (3230br|2294ar)
3230 (3231br|3439ar) 3231 (3232br|2936ar) 3232 (3233br|2032ar) 3233 (3234br|2721ar) 3234 (3235br|2785ar) 3235 (3236br|3006ar) 3236 (3237br|2627ar) 3237 (3238br|3824ar) 3238 (3239br|2464ar) 3239 (3240br|1969ar)
3240 (3241br|2949ar) 3241 (3242br|2949ar) 3242 (3243br|2949ar) 3243 (3244br|2949ar) 3244 (3245br|2949ar) 3245 (3246br|2949ar) 3246 (3247br|2949ar) 3247 (3248br|2949ar) 3248 (3249br|2949ar) 3249 (3250br|2949ar)
3250 (3251br|1903ar) 3251 (3252br|1981ar) 3252 (3253br|2457ar) 3253 (3254br|3768ar) 3254 (3255br|2136ar) 3255 (3256br|2342ar) 3256 (3257br|3567ar) 3257 (3258br|2123ar) 3258 (3259br|2133ar) 3259 (3260br|3361ar)
3260 (3261br|2625ar) 3261 (3262br|3633ar) 3262 (3263br|2113ar) 3263 (3264br|3426ar) 3264 (3265br|2625ar) 3265 (3266br|3928ar) 3266 (3267br|3172ar) 3267 (3268br|2173ar) 3268 (3269br|1945ar) 3269 (3270br|2258ar)
3270 (3271br|3717ar) 3271 (3272br|3717ar) 3272 (3273br|3717ar) 3273 (3274br|3717ar) 3274 (3275br|3717ar) 3275 (3276br|3717ar) 3276 (3277br|3717ar) 3277 (3278br|3717ar) 3278 (3279br|3717ar) 3279 (3280br|3717ar)
3280 (3281br|3719ar) 3281 (3282br|3668ar) 3282 (3283br|2668ar) 3283 (3284br|2136ar) 3284 (3285br|2113ar) 3285 (3286br|2793ar) 3286 (3287br|2793ar) 3287 (3288br|2702ar) 3288 (3289br|1958ar) 3289 (3290br|2920ar)
3290 (3291br|2462ar) 3291 (3292br|3170ar) 3292 (3293br|3437ar) 3293 (3294br|2160ar) 3294 (3295br|3660ar) 3295 (3296br|1937ar) 3296 (3297br|1937ar) 3297 (3298br|3361ar) 3298 (3299br|2797ar) 3299 (3300br|3698ar)
3300 (3301br|3780ar) 3301 (3302br|3780ar) 3302 (3303br|3780ar) 3303 (3304br|3780ar) 3304 (3305br|3780ar) 3305 (3306br|3780ar) 3306 (

4150 (4151ar|4151ar) 4151 (4157ar|4157ar) 4152 (4153ar|4154ar) 4153 (4152ar|4147ar) 4154 (4155al|4147ar) 4155 (4156al|4156bl) 4156 (4160al|4160bl) 4157 (4158ar|4159br) 4158 (4176ar|4157br) 4159 (4157ar|4157br) 4160 (4161al|4161br) 4161 (4162al|4164bl) 4162 (4163al|4163bl) 4163 (4160al|4160bl) 4164 (4165al|4165bl) 4165 (4160al|4161bl) 4166 (4167al|4169bl) 4167 (4168al|4168al) 4168 (4141al|4170bl) 4169 (4161al|4161br) 4170 (4161al|4161br) 4171 (4172ar|4172ar) 4172 (4183al|4183bl) 4173 (4184al|4184bl) 4174 (4185al|4185bl) 4175 (4186al|4186bl) 4176 (4187al|4187br) 4177 (4188al|4188al) 4178 (4189al|4189al) 4179 (4170al|4171bl) 4180 (4062al|4062bl) 4181 (4182al|4187br) 4182 (4183al|4187br) 4183 (4184al|4184bl) 4184 (4185al|4186bl) 4185 (4186al|4186bl) 4186 (4187al|4187br) 4187 (4188al|4190bl) 4188 (4189al|4189al) 4189 (4141al|4171bl) 4190 (4191al|4191br) 4191 (4200al|4160bl) 4192 (4193al|4194br) 4193 (4206ar|4197br) 4194 (4195al|4195br) 4195 (4196al|4196bl) 4196 (4234al|4234bl) 4197 (4198ar|4199br) 4198 (4200ar|4200br) 4199 (4200ar|4200br) 4200 (4201al|4201br) 4201 (4202al|4202bl) 4202 (4223al|4224bl) 4203 (4224al|4224bl) 4204 (4225al|4225bl) 4205 (4226al|4226bl) 4206 (4227al|4227bl) 4207 (4228ar|4228br) 4208 (4229ar|4229br) 4209 (4230al|4230bl) 4210 (4211ar|4211br) 4211 (4209ar|4209br) 4212 (4192ar|4212br) 4213 (4192ar|4212br) 4214 (4192ar|4212br) 4215 (4192al|4212bl) 4216 (4192al|4212bl) 4217 (4192al|4212bl) 4218 (4222al|4222bl) 4219 (4220ar|4220br) 4220 (4192ar|4212br) 4221 (4219ar|4219br) 4222 (4223al|4224bl) 4223 (4223al|4224bl) 4224 (4223al|4224bl) 4225 (4223al|4224bl) 4226 (4226ar|4226br) 4227 (4226ar|4226br) 4228 (4226ar|4226br) 4229 (4226ar|4226br) 4230 (4231al|4231br) 4231 (4232al|4232bl) 4232 (4233al|4233bl) 4233 (4234al|4234bl) 4234 (4235al|4235bl) 4235 (4236al|4236bl) 4236 (4237al|4237br) 4237 (4238al|4238al) 4238 (4239al|4239al) 4239 (4240al|4240al) 4240 (4241al|4241bl) 4241 (4242al|4242bl) 4242 (4243al|4243bl) 4243 (4244al|4244bl) 4244 (4245al|4245bl) 4245 (4246al|4246bl) 4246 (4247ar|4247br) 4247 (4248ar|4248br) 4248 (4248ar|4248br) 4249 (4248ar|4248br) 4250 (4251al|4251bl) 4251 (4252al|4252bl) 4252 (4253al|4253bl) 4253 (4254al|4254bl) 4254 (4255al|4255bl) 4255 (4256al|4256bl) 4256 (4257al|4257br) 4257 (4258ar|4258br) 4258 (4258ar|4258br) 4259 (4258ar|4258br) 4260 (4279ar|4279br) 4261 (4262ar|4262br) 4262 (4263ar|4263br) 4263 (4264ar|4264br) 4264 (4265al|4265bl) 4265 (4266al|4266bl) 4266 (4267ar|4267br) 4267 (4270ar|4270br) 4268 (4270ar|4270br) 4269 (4270ar|4270br) 4270 (4271ar|4271br) 4271 (4272ar|4272ar) 4272 (4273ar|4273ar) 4273 (4274ar|4274ar) 4274 (4275al|4275bl) 4275 (4276al|4276bl) 4276 (4277ar|4277br) 4277 (4278ar|4278br) 4278 (4278ar|4278br) 4279 (4278ar|4278br) 4280 (4281al|4281bl) 4281 (4282al|4282bl) 4282 (4283al|4283bl) 4283 (4284al|4284bl) 4284 (4285al|4285bl) 4285 (4286al|4286bl) 4286 (4287ar|4287br) 4287 (4287ar|4287br) 4288 (4287ar|4287br) 4289 (4287ar|4287br) 4290 (4291ar|4291br) 4291 (4292al|4292bl) 4292 (4293al|4293bl) 4293 (4294al|4294bl) 4294 (4295al|4295bl) 4295 (4296al|4296bl) 4296 (4297ar|4297br) 4297 (4297ar|4297br) 4298 (4298al|4298al) 4299 (4298al|4298al) 4300 (4299al|4299br) 4301 (4300ar|4300br) 4302 (4301al|4301br) 4303 (4302al|4302bl) 4304 (4303al|4303bl) 4305 (4304al|4304bl) 4306 (4305al|4305bl) 4307 (4306al|4306bl) 4308 (4307ar|4307br) 4309 (4308al|4308al) 4310 (4309ar|4309br) 4311 (4310ar|4310br) 4312 (4311al|4311br) 4313 (4312al|4312bl) 4314 (4313al|4313bl) 4315 (4314al|4314bl) 4316 (4315al|4315bl) 4317 (4316ar|4316br) 4318 (4317al|4317br) 4319 (4318ar|4318br) 4320 (4319ar|4319br) 4321 (4320al|4320bl) 4322 (4321al|4321bl) 4323 (4322al|4322bl) 4324 (4323al|4323bl) 4325 (4324al|4324bl) 4326 (4325al|4325bl) 4327 (4326al|4326bl) 4328 (4327ar|4327br) 4329 (4328al|4328al) 4330 (4329al|4329br) 4331 (4330ar|4330br) 4332 (4331al|4331br) 4333 (4332al|4332bl) 4334 (4333al|4333bl) 4335 (4334al|4334bl) 4336 (4335al|4335bl) 4337 (4336ar|4336br) 4338 (4337ar|4337br) 4339 (4338al|4338al) 4340 (4339ar|4339br) 4341 (4340ar|4340br) 4342 (4341al|4341br) 4343 (4342al|4342bl) 4344 (4343al|4343bl) 4345 (4344al|4344bl) 4346 (4345al|4345bl) 4347 (4346ar|4346br) 4348 (4347ar|4347br) 4349 (4348ar|4348br) 4350 (4349ar|4349br) 4351 (4350ar|4350br) 4352 (4351al|4351br) 4353 (4352al|4352bl) 4354 (4353al|4353bl) 4355 (4354al|4354bl) 4356 (4355al|4355bl) 4357 (4356ar|4356br) 4358 (4357ar|4357br) 4359 (4358al|4358al) 4360 (4359ar|4359br) 4361 (4360ar|4360br) 4362 (4361al|4361br) 4363 (4362al|4362bl) 4364 (4363al|4363bl) 4365 (4364al|4364bl) 4366 (4365al|4365bl) 4367 (4366ar|4366br) 4368 (4367ar|4367br) 4369 (4368al|4368al) 4370 (4371al|4371br) 4371 (4366al|4366bl) 4372 (4373al|4373bl) 4373 (4374al|4374bl) 4374 (4375al|4375bl) 4375 (4376al|4376bl) 4376 (4377ar|4377br) 4377 (4378ar|4378br) 4378 (4379al|4379br) 4379 (4380ar|4380br) 4380 (4381al|4381br) 4381 (4382al|4382bl) 4382 (4383al|4383bl) 4383 (4384al|4384bl) 4384 (4385al|4385bl) 4385 (4386ar|4386br) 4386 (4387ar|4387br) 4387 (4388al|4388al) 4388 (4389ar|4389br) 4389 (4390ar|4390br) 4390 (4391al|4391br) 4391 (4392al|4392bl) 4392 (4393al|4393bl) 4393 (4394al|4394bl) 4394 (4395al|4395bl) 4395 (4396ar|4396br) 4396 (4397ar|4397br) 4397 (4398al|4398al) 4398 (4399ar|4399br) 4399 (4400al|4400bl) 4400 (4401al|4401br) 4401 (4402al|4402bl) 4402 (4403al|4403bl) 4403 (4404al|4404bl) 4404 (4405al|4405bl) 4405 (4406ar|4406br) 4406 (4407al|4407br) 4407 (4408al|4408al) 4408 (4409ar|4409br) 4409 (4410al|4410br) 4410 (4401al|4401br) 4411 (4412ar|4412ar) 4412 (4413al|4413bl) 4413 (4414al|4414bl) 4414 (4415al|4415bl) 4415 (4416al|4416bl) 4416 (4417ar|4417br) 4417 (4418al|4418al) 4418 (4419ar|4419br) 4419 (4420al|4420bl) 4420 (4421al|4421bl) 4421 (4422ar|4422ar) 4422 (4423al|4423bl) 4423 (4424al|4424bl) 4424 (4425al|4425bl) 4425 (4426al|4426bl) 4426 (4427ar|4427br) 4427 (4428ar|4428br) 4428 (4428ar|4428br) 4429 (4428ar|4428br) 4430 (4429al|4429br) 4431 (4430ar|4430br) 4432 (4431al|4431br) 4433 (4432al|4432bl) 4434 (4433al|4433bl) 4435 (4434al|4434bl) 4436 (4435al|4435bl) 4437 (4436ar|4436br) 4438 (4437ar|4437br) 4439 (4438al|4438al) 4440 (4439ar|4439br) 4441 (4440ar|4440br) 4442 (4441al|4441br) 4443 (4442al|4442bl) 4444 (4443al|4443bl) 4445 (4444al|4444bl) 4446 (4445al|4445bl) 4447 (4446ar|4446br) 4448 (4447ar|4447br) 4449 (4448ar|4448br) 4450 (4449ar|4449br) 4451 (4450ar|4450br) 4452 (4451al|4451br) 4453 (4452al|4452bl) 4454 (4453al|4453bl) 4455 (4454al|4454bl) 4456 (4455al|4455bl) 4457 (4456ar|4456br) 4458 (4457ar|4457br) 4459 (4458al|4458al) 4460 (4459ar|4459br) 4461 (4460ar|4460br) 4462 (4461al|4461br) 4463 (4462al|4462bl) 4464 (4463al|4463bl) 4465 (4464al|4464bl) 4466 (4465al|4465bl) 4467 (4466ar|4466br) 4468 (4467ar|4467br) 4469 (4468ar|4468br) 4470 (4469ar|4469br) 4471 (4470ar|4470br) 4472 (4471al|4471br) 4473 (4472al|4472bl) 4474 (4473al|4473bl) 4475 (4474al|4474bl) 4476 (4475al|4475bl) 4477 (4476ar|4476br) 4478 (4477ar|4477br) 4479 (4478ar|4478br) 4480 (4479ar|4479br) 4481 (4480ar|4480br) 4482 (4481al|4481br) 4483 (4482al|4482bl) 4484 (4483al|4483bl) 4485 (4484al|4484bl) 4486 (4485al|4485bl) 4487 (4486ar|4486br) 4488 (4487ar|4487br) 4489 (4488ar|4488br) 4490 (4489ar|4489br) 4491 (4490ar|4490br) 4492 (4491al|4491br) 4493 (4492al|4492bl) 4494 (4493al|4493bl) 4495 (4494al|4494bl) 4496 (4495al|4495bl) 4497 (4496ar|4496br) 4498 (4497ar|4497br) 4499 (4498ar|4498br) 4500 (4499ar|4499br) 4501 (4500ar|4500br) 4502 (4501al|4501br) 4503 (4502al|4502bl) 4504 (4503al|4503bl) 4505 (4504al|4504bl) 4506 (4505al|4505bl) 4507 (4506ar|4506br) 4508 (4507ar|4507br) 4509 (4508al|4508al) 4510 (4509ar|4509br) 4511 (4510ar|4510br) 4512 (4511al|4511br) 4513 (4512al|4512bl) 4514 (4513al|4513bl) 4515 (4514al|4514bl) 4516 (4515al|4515bl) 4517 (4516ar|4516br) 4518 (4517ar|4517br) 4519 (4518ar|4518br) 4520 (4519ar|4519br) 4521 (4520al|4520bl) 4522 (4521al|4521bl) 4523 (4522al|4522bl) 4524 (4523al|4523bl) 4525 (4524al|4524bl) 4526 (4525al|4525bl) 4527 (4526ar|4526br) 4528 (4527ar|4527br) 4529 (4528al|4528al) 4530 (4529al|4529br) 4531 (4530ar|4530br) 4532 (4531al|4531br) 4533 (4532al|4532bl) 4534 (4533al|4533bl) 4535 (4534al|4534bl) 4536 (4535al|4535bl) 4537 (4536ar|4536br) 4538 (4537ar|4537br) 4539 (4538al|4538al) 4540 (4539ar|4539br) 4541 (4540ar|4540br) 4542 (4541al|4541br) 4543 (4542al|4542bl) 4544 (4543al|4543bl) 4545 (4544al|4544bl) 4546 (4545al|4545bl) 4547 (4546ar|4546br) 4548 (4547ar|4547br) 4549 (4548ar|4548br) 4550 (4549ar|4549br) 4551 (4550ar|4550br) 4552 (4551al|4551br) 4553 (4552al|4552bl) 4554 (4553al|4553bl) 4555 (4554al|4554bl) 4556 (4555al|4555bl) 4557 (4556ar|4556br) 4558 (4557ar|4557br) 4559 (4558al|4558al) 4560 (4559ar|4559br) 4561 (4560ar|4560br) 4562 (4561al|4561br) 4563 (4562al|4562bl) 4564 (4563al|4563bl) 4565 (4564al|4564bl) 4566 (4565al|4565bl) 4567 (4566ar|4566br) 4568 (4567ar|4567br) 4569 (4568al|4568al) 4570 (4569ar|4569br) 4571 (4570ar|4570br) 4572 (4571al|4571br) 4573 (4572al|4572bl) 4574 (4573al|4573bl) 4575 (4574al|4574bl) 4576 (4575al|4575bl) 4577 (4576ar|4576br) 4578 (4577ar|4577br) 4579 (4578ar|4578br) 4580 (4579ar|4579br) 4581 (4580ar|4580br) 4582 (4581al|4581br) 4583 (4582al|4582bl) 4584 (4583al|4583bl) 4585 (4584al|4584bl) 4586 (4585al|4585bl) 4587 (4586ar|4586br) 4588 (4587ar|4587br) 4589 (4588al|4588al) 4590 (4589ar|4589br) 4591 (4590ar|4590br) 4592 (4591al|4591br) 4593 (4592al|4592bl) 4594 (4593al|4593bl) 4595 (4594al|4594bl) 4596 (4595al|4595bl) 4597 (4596ar|4596br) 4598 (4597ar|4597br) 4599 (4598ar|4598br) 4600 (4599ar|4599br) 4601 (4600ar|4600br) 4602 (4601al|4601br) 4603 (4602al|4602bl) 4604 (4603al|4603bl) 4605 (4604al|4604bl) 4606 (4605al|4605bl) 4607 (4606ar|4606br) 4608 (4607ar|4607br) 4609 (4608al|4608al) 4610 (4609ar|4609br) 4611 (4610ar|4610br) 4612 (4611al|4611br) 4613 (4612al|4612bl) 4614 (4613al|4613bl) 4615 (4614al|4614bl) 4616 (4615al|4615bl) 4617 (4616ar|4616br) 4618 (4617ar|4617br) 4619 (4618ar|4618br) 4620 (4619ar|4619br) 4621 (4620al|4620bl) 4622 (4621al|4621bl) 4623 (4622al|4622bl) 4624 (4623al|4623bl) 4625 (4624al|4624bl) 4626 (4625al|4625bl) 4627 (4626ar|4626br) 4628 (4627ar|4627br) 4629 (4628al|4628al) 4630 (4629ar|4629br) 4631 (4630ar|4630br) 4632 (4631al|4631br) 4633 (4632al|4632bl) 4634 (4633al|4633bl) 4635 (4634al|4634bl) 4636 (4635al|4635bl) 4637 (4636ar|4636br) 4638 (4637ar|4637br) 4639 (4638al|4638al) 4640 (4639ar|4639br) 4641 (4640ar|4640br) 4642 (4641al|4641br) 4643 (4642al|4642bl) 4644 (4643al|4643bl) 4645 (4644al|4644bl) 4646 (4645al|4645bl) 4647 (4646ar|4646br) 4648 (4647ar|4647br) 4649 (4648ar|4648br) 4650 (4649ar|4649br) 4651 (4650ar|4650br) 4652 (4651al|4651br) 4653 (4652al|4652bl) 4654 (4653al|4653bl) 4655 (4654al|4654bl) 4656 (4655al|4655bl) 4657 (4656ar|4656br) 4658 (4657ar|4657br) 4659 (4658al|4658al) 4660 (4659ar|4659br) 4661 (4660ar|4660br) 4662 (4661al|4661br) 4663 (4662al|4662bl) 4664 (4663al|4663bl) 4665 (4664al|4664bl) 4666 (4665al|4665bl) 4667 (4666ar|4666br) 4668 (4667ar|4667br) 4669 (4668ar|4668br) 4670 (4669ar|4669br) 4671 (4670ar|4670br) 4672 (4671al|4671br) 4673 (4672al|4672bl) 4674 (4673al|4673bl) 4675 (4674al|4674bl) 4676 (4675al|4675bl) 4677 (4676ar|4676br) 4678 (4677ar|4677br) 4679 (4678ar|4678br) 4680 (4679ar|4679br) 4681 (4680ar|4680br) 4682 (4681al|4681br) 4683 (4682al|4682bl) 4684 (4683al|4683bl) 4685 (4684al|4684bl) 4686 (4685al|4685bl) 4687 (4686ar|4686br) 4688 (4687ar|4687br) 4689 (4688ar|4688br) 4690 (4689ar|4689br) 4691 (4690ar|4690br) 4692 (4691al|4691br) 4693 (4692al|4692bl) 4694 (4693al|4693bl) 4695 (4694al|4694bl) 4696 (4695al|4695bl) 4697 (4696ar|4696br) 4698 (4697ar|4697br) 4699 (4698ar|4698br) 4700 (4699ar|4699br) 4701 (4700ar|4700br) 4702 (4701al|4701br) 4703 (4702al|4702bl) 4704 (4703al|4703bl) 4705 (4704al|4704bl) 4706 (4705al|4705bl) 4707 (4706ar|4706br) 4708 (4707ar|4707br) 4709 (4708ar|4708br) 4710 (4709ar|4709br) 4711 (4710ar|4710br) 4712 (4711al|4711br) 4713 (4712al|4712bl) 4714 (4713al|4713bl) 4715 (4714al|4714bl) 4716 (4715al|4715bl) 4717 (4716ar|4716br) 4718 (4717ar|4717br) 4719 (4718ar|4718br) 4720 (4719ar|4719br) 4721 (4720ar|4720br) 4722 (4721al|4721br) 4723 (4722al|4722bl) 4724 (4723al|4723bl) 4725 (4724al|4724bl) 4726 (4725al|4725bl) 4727 (4726ar|4726br) 4728 (4727ar|4727br) 4729 (4728ar|4728br) 4730 (4729ar|4729br) 4731 (4730ar|4730br) 4732 (4731al|4731br) 4733 (4732al|4732bl) 4734 (4733al|4733bl) 4735 (4734al|4734bl) 4736 (4735al|4735bl) 4737 (4736ar|4736br) 4738 (4737ar|4737br) 4739 (4738ar|4738br) 4740 (4739ar|4739br) 4741 (4740ar|4740br) 4742 (4741al|4741br) 4743 (4742al|4742bl) 4744 (4743al|4743bl) 4745 (4744al|4744bl) 4746 (4745al|4745bl) 4747 (4746ar|4746br) 4748 (4747ar|4747br) 4749 (4748ar|4748br) 4750 (4749ar|4749br) 4751 (4750ar|4750br) 4752 (4751al|4751br) 4753 (4752al|4752bl) 4754 (4753al|4753bl) 4755 (4754al|4754bl) 4756 (4755al|4755bl) 4757 (4756ar|4756br) 4758 (4757ar|4757br) 4759 (4758ar|4758br) 4760 (4759ar|4759br) 4761 (4760ar|4760br) 4762 (4761al|4761br) 4763 (4762al|4762bl) 4764 (4763al|4763bl) 4765 (4764al|4764bl) 4766 (4765al|4765bl) 4767 (4766ar|4766br) 4768 (4767ar|4767br) 4769 (4768ar|4768br) 4770 (4769ar|4769br) 4771 (4770ar|4770br) 4772 (4771al|4771br) 4773 (4772al|4772bl) 4774 (4773al|4773bl) 4775 (4774al|4774bl) 4776 (4775al|4775bl) 4777 (4776ar|4776br) 4778 (4777ar|4777br) 4779 (4778ar|4778br) 4780 (4779ar|4779br) 4781 (4780ar|4780br) 4782 (4781al|4781br) 4783 (4782al|4782bl) 4784 (4783al|4783bl) 4785 (4784al|4784bl) 4786 (4785al|4785bl) 4787 (4786ar|4786br) 4788 (4787ar|4787br) 4789 (4788ar|4788br) 4790 (4789ar|4789br) 4791 (4790ar|4790br) 4792 (4791al|4791br) 4793 (4792al|4792bl) 4794 (4793al|4793bl) 4795 (4794al|4794bl) 4796 (4795al|4795bl) 4797 (4796ar|4796br) 4798 (4797ar|4797br) 4799 (4798ar|4798br) 4800 (4799ar|4799br) 4801 (4800ar|4800br) 4802 (4801al|4801br) 4803 (4802al|4802bl) 4804 (4803al|4803bl) 4805 (4804al|4804bl) 4806 (4805al|4805bl) 4807 (4806ar|4806br) 4808 (4807ar|4807br) 4809 (4808ar|4808br) 4810 (4809ar|4809br) 4811 (4810ar|4810br) 4812 (4811al|4811br) 4813 (4812al|4812bl) 4814 (4813al|4813bl) 4815 (4814al|4814bl) 4816 (4815al|4815bl) 4817 (4816ar|4816br) 4818 (4817ar|4817br) 4819 (4818ar|4818br) 4820 (4819ar|4819br) 4821 (4820ar|4820br) 4822 (4821al|4821br) 4823 (4822al|4822bl) 4824 (4823al|4823bl) 4825 (4824al|4824bl) 4826 (4825al|4825bl) 4827 (4826ar|4826br) 4828 (4827ar|4827br) 4829 (4828al|4828al) 4830 (4829ar|4829br) 4831 (4830ar|4830br) 4832 (4831al|4831br) 4833 (4832al|4832bl) 4834 (4833al|4833bl) 4835 (4834al|4834bl) 4836 (

7210(7193aR|7193bR) 7211(7212aR|7215bR) 7212(7211aR|7213aL) 7213(7214aR|7214aR) 7214(7184aR|7184bR) 7215(7216aL|7218aL) 7216(7217aR|7217aR) 7217(7193aR|7193bR) 7218(7219aR|7219aR) 7219(7202aR|7202bR) 7220(721aL|7226bR) 7221(7222aL|7222aL) 7222(7223bR|7223bR) 7223(7211aR|7211bR) 7224(7225bL|7225bR) 7225(7184aR|7184bR) 7226(7229aR|7227aL) 7227(7228bR|7228bR) 7228(7202aR|7202bR) 7229(7230bR|7230bR) 7230(7231aL|7231bL) 7231(7232aR|7233bL) 7232(7233aL|7233bL) 7233(7234aR|7234bR) 7234(7235aL|7235bL) 7235(7236aR|7236bL) 7236(7237aL|7237bL) 7237(7238aR|7238bR) 7238(7239aR|7239bR) 7239(7240aL|7240bL) 7240(7241aL|7241bL) 7241(7238aL|7243bL) 7242(7238aL|7238bL) 7243(7238aL|7238bL) 7244(7238aL|7247bL) 7245(7238aL|7246bL) 7246(7138aL|7138bL) 7247(7248aL|7248bL) 7248(7238aL|7238bL) 7249(7250aR|7255bR) 7250(7251aL|7253bL) 7251(7252aL|7252bL) 7252(7160aL|7160bL) 7253(7254aL|7254bL) 7254(7149aL|7149bL) 7255(7256aL|7256bL) 7256(7257aL|7257bL) 7257(7149aL|7149bL) 7258(7259aL|7259bL) 7259(7149aL|7149bL) 7260(7261aR|7264bR) 7261(7171aR|7262bL) 7262(7263aL|7263bL) 7263(7160aL|7160bL) 7264(7265aL|7267bL) 7265(7266aL|7266bL) 7266(7160aL|7160bL) 7267(7268aL|7269bL) 7268(7160aL|7160bL) 7269(7270aR|7273bR) 7270(7276aR|7271bL) 7271(7272bR|7272bR) 7272(7283aL|7283bL) 7273(7283aL|7283bL) 7274(7272aL|7274bL) 7275(7269aL|7269bL) 7276(7277aR|7280bR) 7277(7277aR|7277aR) 7278(7279aR|7279aR) 7279(7283aR|7283bR) 7280(7281aL|7281bL) 7281(7282aR|7282aR) 7282(7307aR|7307bR) 7283(7284aR|7285bR) 7284(7285aR|7285bR) 7285(7286aR|7286bR) 7286(7287aR|7288bR) 7287(7286aR|7286bR) 7288(7289aL|7289bL) 7289(7290aR|7290aR) 7290(7301aR|7291bR) 7291(7292aR|7293bR) 7292(7293aL|7293bL) 7293(7294aR|7294bR) 7294(7295aL|7295bL) 7295(7296aR|7296bL) 7296(7297aR|7302bR) 7297(7296aL|7300bL) 7298(7299aL|7299bL) 7299(7296aL|7296bL) 7300(7301aL|7301bL) 7301(7296aL|7296bL) 7302(7303aL|7305bL) 7303(7304aL|7304bL) 7304(7296aL|7296bL) 7305(7296aL|7306bL) 7306(7296aL|7296bL) 7307(7308aR|7309bR) 7308(7307aL|7307bR) 7309(7310aL|7310bL) 7310(7312aR|7312bR) 7311(7312aR|7312bR) 7312(7313aR|7313bR) 7313(7314aL|7316aL) 7314(7316aR|7316bR) 7315(7305aL|7305bL) 7316(7317aR|7317bR) 7317(7321aR|7321bR) 7318(7318aR|7319aL) 7319(7320bR|7320bR) 7320(7326aR|7326bR) 7321(7322aR|7323bL) 7322(7321aL|7321bL) 7323(7324aL|7324bL) 7324(7325aL|7325bL) 7325(7326aL|7326bL) 7326(7327aL|7327bL) 7327(7328aR|7328bR) 7328(7329aL|7329bL) 7329(7330aL|7330bL) 7330(7331aL|7331bL) 7331(7332aL|7332bL) 7332(7333aL|7333bL) 7333(7334aL|7334bL) 7334(7335aL|7335bL) 7335(7336aL|7336bL) 7336(7337aL|7337bL) 7337(7338aR|7338bR) 7338(7339aL|7339bL) 7339(7340aL|7340bL) 7340(7341aL|7341bL) 7341(7342aL|7342bL) 7342(7343aL|7343bL) 7343(7344aL|7344bL) 7344(7345aL|7345bL) 7345(7346aL|7346bL) 7346(7347aL|7347bL) 7347(7348aR|7348bR) 7348(7349aL|7349bL) 7349(7350aL|7350bL) 7350(7351aL|7351bL) 7351(7352aL|7352bL) 7352(7353aL|7353bL) 7353(7354aL|7354bL) 7354(7355aL|7355bL) 7355(7356aL|7356bL) 7356(7357aL|7357bL) 7357(7358aR|7358bR) 7358(7359aL|7359bL) 7359(7360aL|7360bL) 7360(7361aR|7361bR) 7361(7362aL|7362bL) 7362(7363aL|7363bL) 7363(7364aL|7364bL) 7364(7365aL|7365bL) 7365(7366aL|7366bL) 7366(7367aL|7367bL) 7367(7368aR|7368bR) 7368(7369aL|7369bL) 7369(7370aL|7370bL) 7370(7371aL|7371bL) 7371(7372aL|7372bL) 7372(7373aL|7373bL) 7373(7374aL|7374bL) 7374(7375aL|7375bL) 7375(7376aL|7376bL) 7376(7377aR|7377bR) 7377(7378aR|7378aR) 7378(7379aL|7379bL) 7379(7380aL|7380bL) 7380(7381aR|7381aR) 7381(7382aR|7382aR) 7382(7383aR|7383aR) 7383(7384aR|7384aR) 7384(7385aR|7385aR) 7385(7386aR|7386aR) 7386(7387aR|7387aR) 7387(7388aR|7388aR) 7388(7389aR|7389aR) 7389(7390aR|7390aR) 7390(7391aR|7391aR) 7391(7392aR|7392aR) 7392(7393aR|7393aR) 7393(7394aR|7394aR) 7394(7395aL|7395bL) 7395(7396aR|7396aR) 7396(7397aR|7397aR) 7397(7398aR|7398aR) 7398(7399aR|7399aR) 7399(7400aR|7400aR) 7400(7401bR|7401bR) 7401(7384aR|7384aR) 7402(7403aR|7403aR) 7403(7404aR|7404aL) 7404(7405aR|7405aR) 7405(7375aR|7375bR) 7406(7407aL|7409aL) 7407(7408aR|7408aR) 7408(7384aR|7384aR) 7409(7410aR|7410aR) 7410(7393aR|7393bR) 7411(7412aR|7412aR) 7412(7413aL|7413aL) 7413(7414bR|7414bR) 7414(7402aR|7402bR) 7415(7416bR|7416bR) 7416(7375aR|7375bR) 7417(7420aR|7418aL) 7418(7419bR|7419bR) 7419(7393aR|7393bR) 7420(7421bR|7421bR) 7421(7422aL|7422bL) 7422(7423aR|7423aR) 7423(7424aL|7424bL) 7424(7425aR|7425bR) 7425(7426aL|7426bL) 7426(7427aL|7427bL) 7427(7428aR|7428aR) 7428(7384aL|7428bL) 7429(7430aR|7430bR) 7430(7431bL|7431bL) 7431(7432aR|7432aR) 7432(7433aL|7433aL) 7433(7434aL|7434bL) 7434(7435aL|7435bL) 7435(7436aL|7436bL) 7436(7437aL|7437bL) 7437(7438aR|7438aR) 7438(7439aR|7439aR) 7439(7440aR|7440aR) 7440(7441bL|7441bL) 7441(7442aL|7442aL) 7442(7443bR|7443bR) 7443(7444bL|7444bL) 7444(7445aR|7445aR) 7445(7446aR|7446aR) 7446(7447aR|7447aR) 7447(7448aR|7448aR) 7448(7449aL|7449aL) 7449(7450aR|7450aR) 7450(7451aL|7451aL) 7451(7452aR|7452aR) 7452(7453aL|7453aL) 7453(7454bL|7454bL) 7454(7455aR|7455aR) 7455(7456aR|7456aR) 7456(7457aR|7457aR) 7457(7458aL|7458aL) 7458(7459aR|7459aR) 7459(7460aR|7460aR) 7460(7461aL|7461aL) 7461(7462aR|7462aR) 7462(7463aL|7463aL) 7463(7464aR|7464aR) 7464(7465aL|7465aL) 7465(7466aR|7466aR) 7466(7467aL|7467aL) 7467(7468bR|7468bR) 7468(7469aL|7469aL) 7469(7470aR|7470aR) 7470(7471aR|7471aR) 7471(7472bR|7472bR) 7472(7473bR|7473bR) 7473(7474aR|7474aR) 7474(7475bR|7475bR) 7475(7476aL|7476aL) 7476(7477aR|7477aR) 7477(7478aL|7478aL) 7478(7479aL|7479aL) 7479(7480aR|7480aR) 7480(7481aL|7481aL) 7481(7482bL|7482bL) 7482(7483aL|7483aL) 7483(7484aR|7484aR) 7484(7485bL|7485bL) 7485(7486aL|7486aL) 7486(7487aL|7487aL) 7487(7488aL|7488aL) 7488(7489aL|7489aL) 7489(7490aL|7490aL) 7490(7491aL|7491bL) 7491(7500aL|7500bL) 7492(7493aL|7493bL) 7493(7494aL|7494bL) 7494(7495aL|7495bL) 7495(7496aL|7496bL) 7496(7497aL|7497bL) 7497(7498aL|7498bL) 7498(7499aL|7499bL) 7499(7500aL|7500bL) 7500(7501aL|7501bL) 7501(7502aL|7502bL) 7502(7503aL|7503bL) 7503(7504aL|7504bL) 7504(7505aL|7505bL) 7505(7506aL|7506bL) 7506(7507aL|7507bL) 7507(7508aL|7508bL) 7508(7509aL|7509bL) 7509(7510aL|7510bL) 7510(7511aL|7511bL) 7511(7512aL|7512bL) 7512(7513aL|7513bL) 7513(7514aL|7514bL) 7514(7515aL|7515bL) 7515(7516aL|7516bL) 7516(7517aL|7517bL) 7517(7518aL|7518bL) 7518(7519aL|7519bL) 7519(7520aL|7520bL) 7520(7521aL|7521bL) 7521(7522aL|7522bL) 7522(7523aL|7523bL) 7523(7524aL|7524bL) 7524(7525aL|7525bL) 7525(7526aL|7526bL) 7526(7527aL|7527bL) 7527(7528aL|7528bL) 7528(7529aL|7529bL) 7529(7530aL|7530bL) 7530(7531aL|7531bL) 7531(7532aL|7532bL) 7532(7533aL|7533bL) 7533(7534aL|7534bL) 7534(7535aL|7535bL) 7535(7536aL|7536bL) 7536(7537aL|7537bL) 7537(7538aL|7538bL) 7538(7539aL|7539bL) 7539(7540aL|7540bL) 7540(7541aL|7541bL) 7541(7542aL|7542bL) 7542(7543aL|7543bL) 7543(7544aL|7544bL) 7544(7545aL|7545bL) 7545(7546aL|7546bL) 7546(7547aL|7547bL) 7547(7548aL|7548bL) 7548(7549aL|7549bL) 7549(7550aL|7550bL) 7550(7551aL|7551bL) 7551(7552aL|7552bL) 7552(7553aL|7553bL) 7553(7554aL|7554bL) 7554(7555aL|7555bL) 7555(7556aL|7556bL) 7556(7557aL|7557bL) 7557(7558aL|7558bL) 7558(7559aL|7559bL) 7559(7560aL|7560bL) 7560(7561aR|7561bR) 7561(7562aL|7562bL) 7562(7563aL|7563bL) 7563(7564aL|7564bL) 7564(7565aL|7565bL) 7565(7566aL|7566bL) 7566(7567aL|7567bL) 7567(7568aL|7568bL) 7568(7569aL|7569bL) 7569(7570aL|7570bL) 7570(7571aL|7571bL) 7571(7572aL|7572bL) 7572(7573aL|7573bL) 7573(7574aL|7574bL) 7574(7575aL|7575bL) 7575(7576aL|7576bL) 7576(7577aL|7577bL) 7577(7578aL|7578bL) 7578(7579aL|7579bL) 7579(7580aL|7580bL) 7580(7581aL|7581bL) 7581(7582aL|7582bL) 7582(7583aL|7583bL) 7583(7584aL|7584bL) 7584(7585aL|7585bL) 7585(7586aL|7586bL) 7586(7587aL|7587bL) 7587(7588aL|7588bL) 7588(7589aL|7589bL) 7589(7590aL|7590bL) 7590(7591aL|7591bL) 7591(7592aL|7592bL) 7592(7593aL|7593bL) 7593(7594aL|7594bL) 7594(7595aL|7595bL) 7595(7596aL|7596bL) 7596(7597aL|7597bL) 7597(7598aL|7598bL) 7598(7599aL|7599bL) 7599(7600aL|7600bL) 7600(7601aR|7601bR) 7601(7602aL|7602bL) 7602(7603aL|7603bL) 7603(7604aR|7604bR) 7604(7605aL|7605bL) 7605(7606aL|7606bL) 7606(7607aL|7607bL) 7607(7608aR|7608bR) 7608(7609aL|7609bL) 7609(7610aL|7610bL) 7610(7611aL|7611bL) 7611(7612aL|7612bL) 7612(7613aL|7613bL) 7613(7614aL|7614bL) 7614(7615aL|7615bL) 7615(7616aL|7616bL) 7616(7617aL|7617bL) 7617(7618aL|7618bL) 7618(7619aL|7619bL) 7619(7620aL|7620bL) 7620(7621aL|7621bL) 7621(7622aL|7622bL) 7622(7623aL|7623bL) 7623(7624aL|7624bL) 7624(7625aL|7625bL) 7625(7626aL|7626bL) 7626(7627aL|7627bL) 7627(7628aL|7628bL) 7628(7629aL|7629bL) 7629(7630aL|7630bL) 7630(7631aR|7631bR) 7631(7632aL|7632bL) 7632(7633aL|7633bL) 7633(7634aL|7634bL) 7634(7635aL|7635bL) 7635(7636aL|7636bL) 7636(7637aL|7637bL) 7637(7638aL|7638bL) 7638(7639aL|7639bL) 7639(7640aL|7640bL) 7640(7639aR|7639bR) 7641(7639aL|7639bL) 7642(7640aL|7640bL) 7643(7641aL|7641bL) 7644(7642aL|7642bL) 7645(7643aL|7643bL) 7646(7644aL|7644bL) 7647(7645aL|7645bL) 7648(7646aL|7646bL) 7649(7647aL|7647bL) 7650(7648aL|7648bL) 7651(7649aL|7649bL) 7652(7650aL|7650bL) 7653(7651aL|7651bL) 7654(7652aL|7652bL) 7655(7653aL|7653bL) 7656(7654aL|7654bL) 7657(7655aL|7655bL) 7658(7656aL|7656bL) 7659(7657aL|7657bL) 7660(7658aL|7658bL) 7661(7659aL|7659bL) 7662(7660aL|7660bL) 7663(7661aR|7661bR) 7664(7662aL|7662bL) 7665(7663aL|7663bL) 7666(7664aL|7664bL) 7667(7665aL|7665bL) 7668(7666aL|7666bL) 7669(7667aL|7667bL) 7670(7668aL|7668bL) 7671(7669aL|7669bL) 7672(7670aL|7670bL) 7673(7671aL|7671bL) 7674(7672aL|7672bL) 7675(7673aL|7673bL) 7676(7674aL|7674bL) 7677(7675aL|7675bL) 7678(7676aL|7676bL) 7679(7677aL|7677bL) 7680(7678aL|7678bL) 7681(7679aL|7679bL) 7682(7680aL|7680bL) 7683(7681aL|7681bL) 7684(7682aL|7682bL) 7685(7683aL|7683bL) 7686(7684aL|7684bL) 7687(7685aL|7685bL) 7688(7686aL|7686bL) 7689(7687aL|7687bL) 7690(7688aL|7688bL) 7691(7689aL|7689bL) 7692(7690aL|7690bL) 7693(7691aL|7691bL) 7694(7692aL|7692bL) 7695(7693aL|7693bL) 7696(7694aL|7694bL) 7697(7695aL|7695bL) 7698(7696aL|7696bL) 7699(7697aL|7697bL) 7700(7701aL|7701bL) 7701(7696aL|7696bL) 7702(7703aL|7703bL) 7703(7704aL|7704bL) 7704(7697aL|7697bL) 7705(7706aL|7706bL) 7706(7698aL|7698bL) 7707(7708aL|7708bL) 7708(7709aL|7709bL) 7709(7710aL|7710bL) 7710(7711aL|7711bL) 7711(7712aL|7712bL) 7712(7713aL|7713bL) 7713(7714aL|7714bL) 7714(7715aL|7715bL) 7715(7716aL|7716bL) 7716(7717aL|7717bL) 7717(7718aL|7718bL) 7718(7719aL|7719bL) 7719(7720aL|7720bL) 7720(7721aL|7721bL) 7721(7718aL|7718bL) 7722(7723aL|7723bL) 7723(7724aL|7724bL) 7724(7725aL|7725bL) 7725(7726aL|7726bL) 7726(7727aL|7727bL) 7727(7728aL|7728bL) 7728(7729aL|7729bL) 7729(7730aL|7730bL) 7730(7731aL|7731bL) 7731(7732aL|7732bL) 7732(7733aL|7733bL) 7733(7734aL|7734bL) 7734(7735aL|7735bL) 7735(7736aL|7736bL) 7736(7737aL|7737bL) 7737(7738aL|7738bL) 7738(7739aL|7739bL) 7739(7740aL|7740bL) 7740(7741aL|7741bL) 7741(7738aL|7738bL) 7742(7743aL|7743bL) 7743(7744aL|7744bL) 7744(7745aL|7745bL) 7745(7746aL|7746bL) 7746(7747aL|7747bL) 7747(7748aL|7748bL) 7748(7749aL|7749bL) 7749(7750aL|7750bL) 7750(7751aL|7751bL) 7751(7752aL|7752bL) 7752(7753aL|7753bL) 7753(7754aL|7754bL) 7754(7755aL|7755bL) 7755(7756aL|7756bL) 7756(7757aL|7757bL) 7757(7758aL|7758bL) 7758(7759aL|7759bL) 7759(7760aL|7760bL) 7760(7761aL|7761bL) 7761(7762aL|7762bL) 7762(7763aL|7763bL) 7763(7764aL|7764bL) 7764(7765aL|7765bL) 7765(7766aL|7766bL) 7766(7767aL|7767bL) 7767(7768aL|7768bL) 7768(7769aL|7769bL) 7769(7770aL|7770bL) 7770(7771aL|7771bL) 7771(7772aL|7772bL) 7772(7773aL|7773bL) 7773(7774aL|7774bL) 7774(7775aL|7775bL) 7775(7776aL|7776bL) 7776(7777aL|7777bL) 7777(7778aL|7778bL) 7778(7779aL|7779bL) 7779(7780aL|7780bL) 7780(7781aL|7781bL) 7781(7782aL|7782bL) 7782(7783aL|7783bL) 7783(7784aL|7784bL) 7784(7785aL|7785bL) 7785(7786aL|7786bL) 7786(7787aL|7787bL) 7787(7788aL|7788bL) 7788(7789aL|7789bL) 7789(7790aL|7790bL) 7790(7791aL|7791bL) 7791(7792aL|7792bL) 7792(7793aL|7793bL) 7793(7794aL|7794bL) 7794(7795aL|7795bL) 7795(7796aL|7796bL) 7796(7797aL|7797bL) 7797(7798aL|7798bL) 7798(7799aL|7799bL) 7799(7800aL|7800bL) 7800(7801aL|7801bL) 7801(7802aL|7802bL) 7802(7803aL|7803bL) 7803(7804aL|7804bL) 7804(7805aL|7805bL) 7805(7806aL|7806bL) 7806(7807aL|7807bL) 7807(7808aL|7808bL) 7808(7809aL|7809bL) 7809(7810aL|7810bL) 7810(7811aL|7811bL) 7811(7812aL|7812bL) 7812(7813aL|7813bL) 7813(7814aL|7814bL) 7814(7815aL|7815bL) 7815(7816aL|7816bL) 7816(7817aL|7817bL) 7817(7818aL|7818bL) 7818(7819aL|7819bL) 7819(7820aL|7820bL) 7820(7821aL|7821bL) 7821(7822aL|7822bL) 7822(7823aL|7823bL) 7823(7824aL|7824bL) 7824(7825aL|7825bL) 7825(7826aL|7826bL) 7826(7827aL|7827bL) 7827(7828aL|78