Continental Mathematics League

2016 - 2017

Computer Science Contest

Grades 6-8

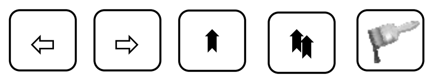
The contest consists of three "meets." Each meet has six questions for 30 minutes.

Note: All parts of the problem must be answered correctly for a student to receive credit for the problem. There is no partial credit.

Computer Science Grades 6-8

Meet 1 — January 26, 2017

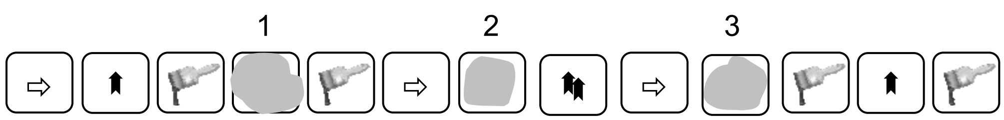
1. Tigger is a programmable robot that moves on the floor, facing a tiled wall, and can paint tiles on the wall. He can step to the left and to the right by one tile and paint the bottom tile on the wall. He can also jump up by one or two tiles and come down again. These five commands are represented by the following blocks and codes:



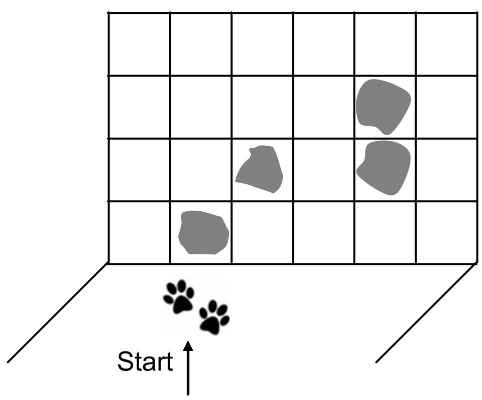
Code: L R J1 J2 P

A paint command immediately after a jump command tells Tigger to paint the tile he is facing, in the second or third row from the floor. After each jump (whether or not he has painted anything) Tigger lands in the same spot from which he jumped.

Tigger was given this program (of which three commands are hidden):



He painted this:



What are the commands not shown? Represent the commands using the corresponding codes in your answer.

Answer: 1: __________, 2: ___________, 3: ___________

2. Programs for some ancient computers were written using the octal (base 8) number system. The octal number system uses only eight digits: 0, 1, ..., 7. Here is how the numbers 0 through 10 are written in octal:

	Octal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	10
9	11
10	12

How do you write 35 in octal?

Answer: __________

3. In Python, `x // y` means the quotient and `x % y` means the remainder when `x` is divided by `y`. Consider the following *generator function*:

```
def collatz(x):
    while True:
        yield x
        if x % 2 is 0:
            x = x // 2
        else:
            x = 3*x + 1
```

It yields the numbers in what is known as the *Collatz sequence*, starting at a given number. For example,

```
collatz_sequence = collatz(1) # start with x = 1
print(next(collatz_sequence))
print(next(collatz_sequence))
print(next(collatz_sequence))
print(next(collatz_sequence))
```

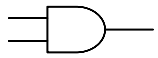
prints 1, 4, 2, and 1. What numbers will

```
collatz_sequence = collatz(6)
print(next(collatz_sequence))
print(next(collatz_sequence))
print(next(collatz_sequence))
print(next(collatz_sequence))
print(next(collatz_sequence))
```

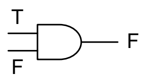
print?

Answer: _______, _______, _______, _______, and _______

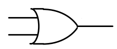
4. An electronic circuit can be made of *gates*. The AND gate is drawn like this:



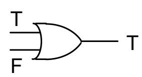
It takes two inputs, each of which can be T (true) or F (false), and produces one output. If both inputs are T, the output is T, otherwise the output is F. For example:



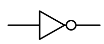
The OR gate is drawn like this:



If at least one of the inputs of the OR gate is T, then its output is T. If both inputs are F, then the output is F. For example:

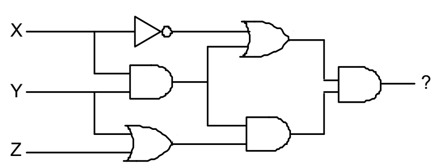


The NOT gate —



— takes one input and produces one output. When the input is T, the output is F; and when the input is F, the output is T.

This circuit —



— takes three inputs, *X*, *Y*, and *Z*. Give two sets of inputs for which the output is T.

Answer:

1. *X* = _______, *Y* = ______, *Z* = ______

and

2. *X* = _______, *Y* = ______, *Z* = ______

5. In Java, a popular programming language, the statement

```
System.out.println("2 + 2 = " + (2 + 2));
```

displays

```
2 + 2 = 4
```

But the statement

```
System.out.println("2 + 2 = " + 2 + 2);
```

displays

```
2 + 2 = 22
```

What does

```
System.out.println("2" + 2 + "=" + 2 + 2);
```

display?

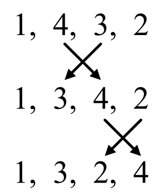
(A) `2+2=4`
(B) `4=4`
(C) `2+2=22`
(D) `22=22`

Answer: __________

6. Arranging a list of numbers in order is called *sorting*. One sorting *algorithm* (method) is called *Bubble Sort*. It works like this:

1. Set *n* to the number of *elements* (numbers) in the list
2. For each of the first *n*-1 numbers in the list, compare that number with the next one and swap the pair if they are out of order
3. Subtract 1 from *n*
4. Repeat Steps 2 and 3 until *n* becomes 1 or no elements were swapped in Step 2 (the list is in order)

For example, if we apply Bubble Sort to the list `[1, 4, 3, 2]`, three comparisons and two swaps will take place on the first pass through the list:



On the second pass, *n* is 3 and the first two pairs of numbers are worked on. Two comparisons and one swap will give `[1, 2, 3, 4]`. On the third pass, one comparison and no swaps take place; *n* becomes 1 and the list is sorted. If a comparison costs one cent and a swap costs three cents, the total cost of sorting this list will be 15 cents. What is the total cost of sorting the list `[3, 1, 6, 5, 2, 4]` using Bubble Sort?

Answer: _____________

Computer Science Grades 6-8

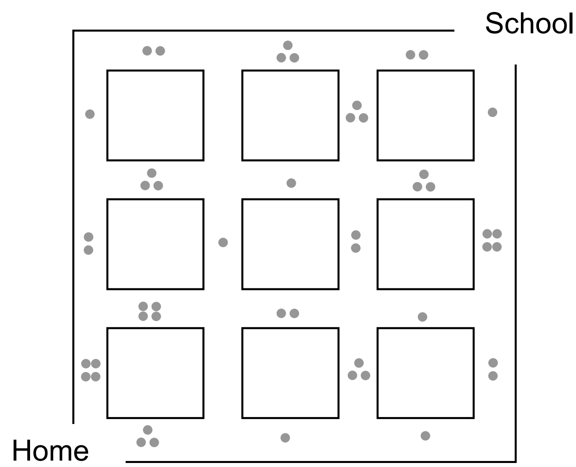
Meet 2 — February 16, 2017

1. In the theory of *formal grammars* you can apply a rule or several rules to a string of letters. For example, if you apply the rule `AAB=>A` to the string `CAABACBA`, you get `CAACBA`. If you now apply the rule `CB=>B` to the result, you get `CAABA`. If you apply the first rule again to the result, you get `CAA`. So using these two rules you can go in three steps from `CAABACBA` to `CAA`. Suppose you start with `ABCBAABCAACBC` and you want to get `BCABAC` with the above two rules, applied in any order. Which of the following is true?

   (A) This is not possible, because the difference between the number of `A`'s and the number of `B`'s always remains the same.
   (B) This is not possible, because the number of `A`'s always remains odd.
   (C) This is possible but you need at least 7 steps.
   (D) This is possible to do in 5 steps.

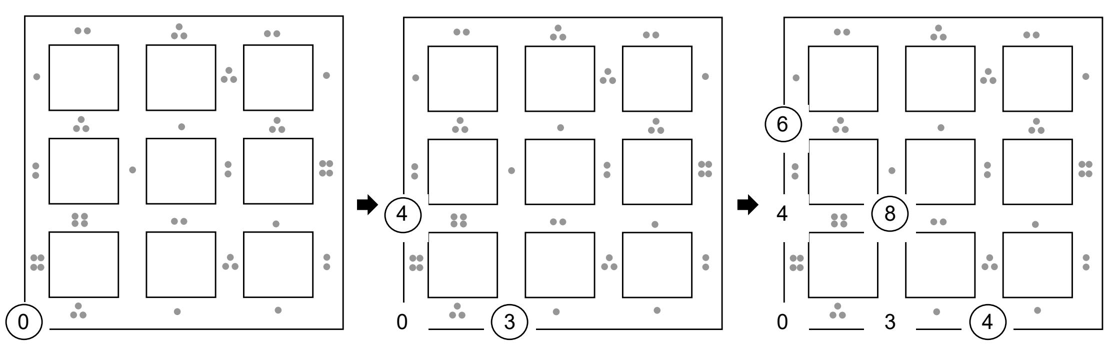
   Answer: __________

2. Penelope (Penny for short) likes to collect coins, so her mother created a board game in which Penny can collect coins on her way from "Home" to "School." The route to School is along the "streets" on the board. School is located three blocks east and three blocks north from Home. If coins are placed like this —



— what is the largest number of coins that Penny can collect on her way from Home to School, if on every step she moves only north or east?

<u>Hint</u>: The total number of possible paths here is 20, so an examination of all of them is time-consuming. A better approach is an algorithm called *dynamic programming*. Mark the start node as 0. Mark all the nodes that can be reached only from an already marked node with the value of the optimal path <u>to that node</u>. Here are the first two steps:



Repeat until all the nodes are marked.

Answer: ___________

3. Python is a popular programming language. You can have a *dialog* with the Python *interpreter*. For example (user input is shown in bold):

```
>>> 2 * 2
4
```

Fill in the blank in the following dialog:

```
>>> list(range(1, 7))
[1, 2, 3, 4, 5, 6]

>>> list(range(1, 7, 2))
[1, 3, 5]

>>> list(range(7, 1, -1))
[7, 6, 5, 4, 3, 2]

>>> list(range(7, 1, -2))
```

____

4. Computers represent numbers in the *binary number system*, which uses only two digits, 0 and 1 (as opposed to our usual decimal number system with ten digits):

Decimal	Binary
0	0
1	1
2	10
3	11
4	100
5	101
6	110
...	...

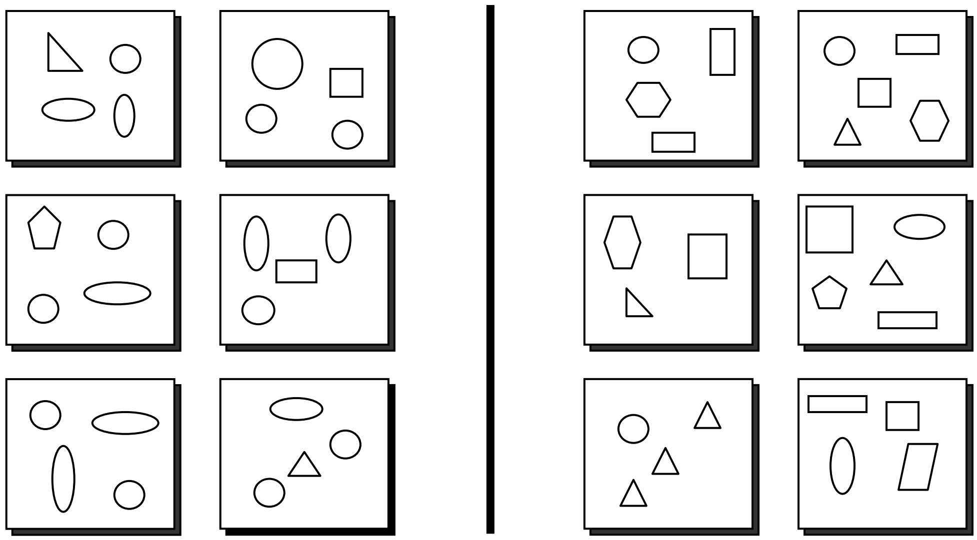
In many programming languages, `^` represents the *bit-wise XOR* operator, which works on corresponding digits ("bits") in the binary representation of two numbers (with zeros added on the left of the shorter number, if necessary). If the two digits in the same position are the same, the corresponding digit in the result is 0; otherwise the corresponding digit in the result is 1. For example:

```
 101
^011
----
 110
```

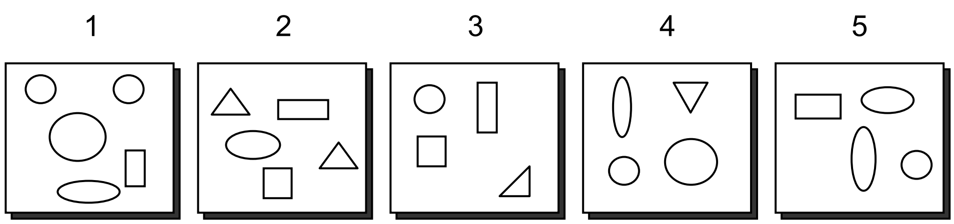
So `5^3` gives `6`. What is the value of `9^5`? Write your answer in the usual decimal number system.

Answer: ______________

5. Mikhail Bongard, a Russian computer scientist, invented a type of puzzle that a computer was supposed to solve. Here is one puzzle of this type:



The six pictures to the left of the dividing line are in some way different from the six pictures to the right. Here are five more pictures:



Three of them belong with the left-side pictures. Which ones?

Answer: _________, _________, and _________

6. Most programming languages let programmers define *recursive functions*. A recursive function includes a call or several calls to itself. Consider this recursive function:

```
def mystery(n):
  if n is 1:
    return 1
  else:
    return n*(mystery(n-1) + 1)
```

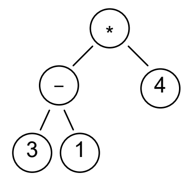
`mystery(1)` returns 1. `mystery(2)` returns 2*(1 + 1) = 4. What number will `mystery(5)` return?

Answer: __________

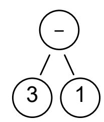
Computer Science Grades 6-8

Meet 3 — March 16, 2017

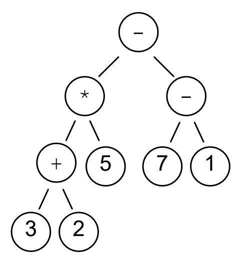
1. An arithmetic expression can be represented as an *expression tree*. For example, (3 − 1)*4 can be represented as



For each *node* that holds an arithmetic operation sign (+, −, *, or /), the left and right branches under that node are *evaluated* (computed) first, then the specified operation is performed. In the above tree,



gives 2, then 2*4 gives 8. What is the result of evaluating the following tree?



Answer: __________

2. Penelope (Penny for short) wants to arrange nine identical coins in a square, some heads up, others heads down. For example:

How many different arrangements are possible?

Answer: __________

3. C is a programming language. In C, `==` means "equals," `!=` means "not equal," `++` means "add one," and `&&` means "and." `x % y` gives the remainder when `x` is divided by `y`. The code in C

```
for (int n = 1; n <= 12; n++)
  if (n % 2 == 1)
    cout << n << " ";
```

produces the output

```
1 3 5 7 9 11
```

The code

```
for (int n = 1; n <= 12; n++)
  if (n % 3 == 2)
    cout << n << " ";
```

produces the output

```
2 5 8 11
```

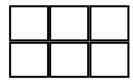
What output is produced by the following?

```
for (int n = 1; n <= 12; n++)
  if (n % 2 == 1 && n % 3 != 2)
    cout << n << " ";
```

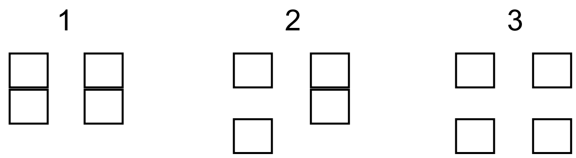
(A) `5 11`
(B) `1 3 7 9`
(C) `1 2 3 5 7 8 9 11`
(D) `1 2 3 5 5 7 8 9 11 11`

Answer: __________

4. A chocolate bar has "grooves" that separate squares:



You want to break the bar into small squares. If you are allowed to break any piece of chocolate into two pieces along a groove, it will take three breaks to break a 2 by 2 bar into four squares:



How many breaks are needed to break a 5 by 10 chocolate bar into fifty small squares?

Answer: __________

5. In Java, a `LinkedList` holds a list of items. The snippet of Java code below *prompts* you to enter a positive integer number *n*, then creates and displays a list of *n* numbers. `LinkedList`'s `add(x)` *method* (function) adds `x` at the end of the list, and `getLast` method returns the last item in the list. `k++` adds 1 to `k`; `k*x` means `k` times `x`.

```
Scanner kboard = new Scanner(System.in);
System.out.print("Enter a positive integer: ");
int n = kboard.nextInt();

// Start with an empty list:
LinkedList<Integer> numbers = new LinkedList<Integer>();

numbers.add(1);

for (int k = 2; k <= n; k++)
{
  int x = numbers.getLast();
  numbers.add(k*x);
}

System.out.println(numbers);
```

If you enter 3, the code displays

```
[1, 2, 6]
```

What is displayed when you enter 5?

Answer: ____________________________________

6. Ximena took ten tests during the school year. Ximena's teacher told her parents that at first her test scores were improving, getting better and better until she scored 100 on one of the tests (not the first two). Unfortunately, from that point on, Ximena's test scores went downhill, getting worse and worse, with each score lower than the one before. Ximena's parents want to find out on which test she got 100 by asking her questions like this: "What was your score on test number ___?" (fill in the number of a particular test). Using an optimal *algorithm* (method), what is the smallest number of questions Ximena's parents have to ask her?

Answer: ______________

Computer Science Grades 6-8

Answers and Solutions

Meet 1

1. L, R, J2. Also acceptable: ⇦ ⇨ 
2. 43
3. 6, 3, 10, 5, and 16
4. X = T, Y = T, Z = T and X = T, Y = T, Z = F
5. D
6. 35

Solutions:

1. The second blob is painted first, so Tigger has to return to paint the first blob. The first double jump has no effect.

2. Notice that 8 is represented in octal as 10, 16 in octal is 20, and so on. So 32 = 4*8 in octal is 40. 35 = 32 + 3, so 35 in octal is 43.

3. 6 is even, so the next number is 6/2 = 3. Since 3 is odd, the next number is 3*3 + 1 = 10. The next number is 10/2 = 5. The next number is 3*5 + 1 = 16. Notice that the sequence continues with 8, 4, 2, 1, and then forever repeats 4, 2, 1. The *Collatz conjecture* states that no matter which number you start from, the sequence always converges to repeating 4, 2, 1. The conjecture has been verified by computer for very many starting numbers, but no mathematical proof of this conjecture has been found so far.

4. If X is F or Y is F, the output of the leftmost AND gate is F and the outputs of the other two AND gates are also F. On the other hand, if Y is T, the value of Z doesn't matter, because Z is "ORed" with Y.

5. A string in quotes plus a number *concatenates* (joins) the string and the number. So `"2" + 2` makes `"22"`. A string plus a string concatenates the two strings, so `"22" + "="` makes `"22="`. And so on.

6. On the first pass through the list (in Step 2), there will be 5 comparisons and 4 swaps. On the second pass, there will be 4 comparisons and 2 swaps. On the third pass, there will be 3 comparisons and 1 swap. On the last pass, there will be 2 comparisons and no swaps. The total is 14 comparisons and 7 swaps; $1 \cdot 14 + 3 \cdot 7 = 35$.

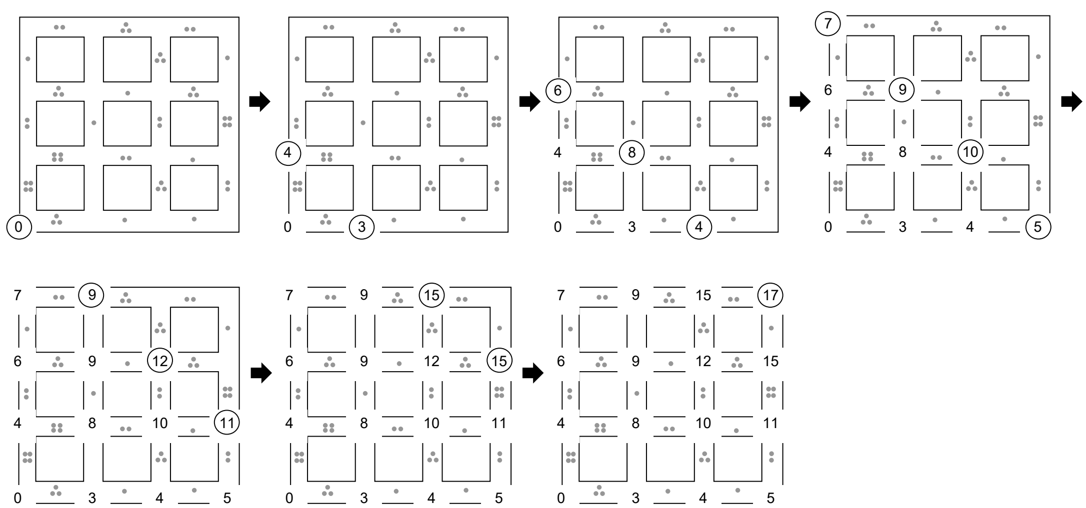
Meet 2

1. A
2. 17
3. `[7, 5, 3]`
4. 12
5. 1, 4, and 5
6. 325

Solutions:

1. `AAB=>A` reduces the number of `A`'s and `B`'s by one, so the difference of their counts remains the same. `CB=>B` does not change the number of `A`'s or `B`'s. In `ABCBAABCAACBC` the difference between the counts of `A`'s and `B`'s is 1, and in `BCABAC` it is 0. Choice B is false, because `AAB=>A` reduces the count of `A`'s by one.

2. This problem has to do with finding an optimal path on a *weighted directed graph*. In this graph the total number of paths is 20 (it is equal to the number of ways to choose three vertical segments out of six segments), so a brute-force examination of all possible paths is time-consuming. Dynamic programming is a much better strategy:



3. Python's `range` function returns a sequence of numbers within a certain range. In a call `range(a, b)`, `a` is the first number in the sequence; `b` and the numbers beyond `b` are outside the range — they are not included in the sequence. The third *parameter*, if given, is the size of the step; if not given it is 1 by default. When the step is negative, the numbers in the sequence are decreasing.

4. 9 is 1001 in binary; 5 is 0101 in binary. `9^5` makes 1100 in binary, which represents decimal 12.

5. The left-side pictures have more circles and ovals than polygons.

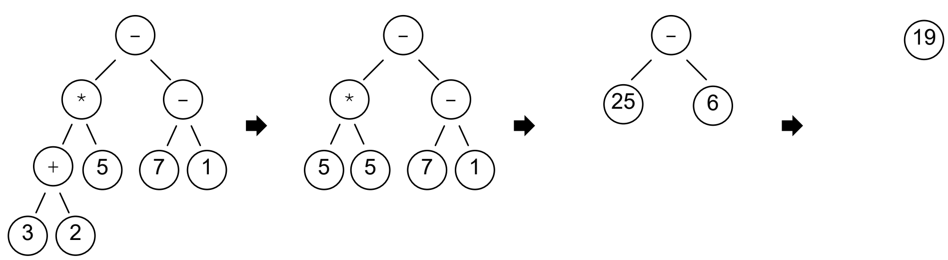
6. `mystery(3)` returns 3*(4 + 1) = 15. `mystery(4)` returns 4*(15 + 1) = 64. `mystery(5)` returns 5*(64 + 1) = 325.

Meet 3

1. 19
2. 512
3. B
4. 49
5. `[1, 2, 6, 24, 120]`
6. 3

Solutions:

1.



2. Each of the coins can be in two possible states, so the number of different arrangements of nine coins is $2^9 = 512$.

3. The output is odd numbers between 1 and 12 (`n % 2 == 1`), except those that give the remainder 2 when divided by 3 (`n % 3 != 2`), that is, excluding 5 and 11 .

4. How many breaks do you need to break a stick or any shape into fifty pieces of arbitrary lengths or shapes? Initially we have one piece. Each break adds one more piece. So no matter how you proceed, you will use *n*-1 breaks to get *n* pieces. This fact might not be so obvious at first for a chocolate bar with restrictions on how you can break it. (Sometimes it is easier to solve a more general problem than a more specific version of it.) You might start by experimenting with a small bar, for example, 2 by 3. No matter how you break it, you will use five breaks. For a 5 by 10 bar, if you first break it into five long strips and then each strip into ten pieces, you will use 4 + 5*9 = 49 breaks. If you break a 5 by 10 bar first into ten short strips then each strip into five pieces, you will use 9 + 10*4 = 49 breaks. If still not convinced, break the bar in half first, then break each half into 25 pieces (using 24 breaks) — the total number of breaks will be 1 + 2*24 = 49.

5. When $k = 2$, $1 \cdot 2 = 2$ is added to the list. The next value, when $k = 3$, is $2 \cdot 3 = 6$. The next value, when $k = 4$, is $6 \cdot 4 = 24$. And so on. The *k*-th number is $1 \cdot 2 \cdot 3 \cdot ... \cdot k$; this product is called *k factorial* and denoted $k!$.

6. Ximena's test results can be represented by the string <<??????>>. Her parents can first ask her about her scores on tests 5 and 6. If 100 is among these two, they've found it. Otherwise if the 5th test score is less than the 6th test score, then the 100 test is to the right (<<<<<<??>>), and asking Ximena about either test 7 or test 8 will locate it. If the 5th test score is greater than the 6th test score, then the 100 test is to the left (<<??>>>>>>), and asking about either test 3 or test 4 will locate the 100. The search method that always divides the remaining search range into two (approximately) equal parts is called *Binary Search*. Notice that only two more questions would be needed for 18 tests.