# C&! Entrance Exam

**Read This First**

*(Parents: These problems can be explored without end. You won't be helpful on the answers, directly, but please help your child navigate the process: (1) Give your child the test and let them consider it for a few days. (2) After a few days, ask for an explanation of the most interesting question, and approaches for tackling it. Provide paper and other materials. (3) Help them assemble their final thoughts and answers.*

*We are interested in your child's thinking and engagement. Excited, confused and eager answers worked out over a few days (or even weeks!), on just a fragment of the test, is what we hope for. One well-thought-through answer is great, and answers to five or more problems are somewhat too many.*

*And maybe nothing on the test is very engaging. That's fine! But it is a sign that C&! might not be very engaging, either.)*

**Hey Kids!** Here are some problems for you to play with. Read them over and see what looks interesting. Pick something.

Then dig in! Play around!! Experiment with examples!! Change things up!!

Put the problem away for a few days and then play some more!

Then tell us what you found out.

**Take Note!**

- These problems take some real thought and experimentation! Don't expect just to write down a good answer right away like you might in school or on a regular test. That just won't work at C&!
- You can't do them all. Well, we can't but maybe you can. You may just have to pick a few, and really play with them.
- You'll need to use a lot more paper than we gave you!
- We want to see your all of your work! It's all great, even the ideas that didn't work out!
- But also write down your final answer and reasoning in a nice neat way so we can understand what you mean. If you just can't get enough of this stuff, C&!'s the place for you!!

We hope you find some of these problems as fun and interesting as we do! When you explore them, you'll get to know us a little bit. Send us your work and we'll get to know you!

**Problem -1.**

This isn't really a problem, so we didn't want to call it problem one.

Visit https://projecteuler.net and look through some of their problems. They are challenging math problems that require some thoughtful programming to solve. OK, maybe one or two of them you can solve without a computer - if so tell us about it! - but many of them require deep thinking and careful, efficient programming.

Go. Try a problem. Can you solve 10 of them? 50? Pick one or more problems that you particularly like and tell us about your solution. Send us your code, any language is fine!

**Problem 0.**

Again, not really a problem, so we won't call it problem one. Hopefully we'll get to a real problem soon.

Visit the NACLO website and try some of their practice problems. What is that I hear you say? Linguistics problems for a math camp? You bet. The patient and systematic reasoning you need to solve these problems is exactly the sort of patient and systematic reasoning you need to do mathematics: steady accumulation of knowledge through logical deduction.

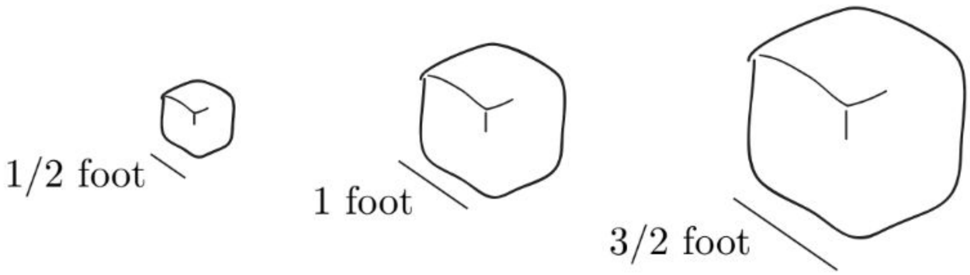
Plus, languages are awesome.

So. Go. Try a problem. Try a few more. Pick one or two that are your favorites, write them up, and send them to us.

**Problem 1.**

Finally a real problem. Whew.

We have a collection of weird rocks: each one is a perfect cube and its width, length and height are all equal to each other— let's call this (obviously) the "size" of the cubical rock. The first rock has size 1/2 foot, the second has size 1 foot, and the third has size 3/2 foot. The average of all these sizes is 1 foot.




But what about the average of their volumes? Calculate this. Is this the average volume the same as the volume of the rock with the average size? More? Less? Can you explain this in general? Which of the two averages is almost always smaller? Can it be larger? When are they ever equal? What is the size of a rock of average volume? Try out a bunch of examples! What's going on?

**Problem 2**.

Expand 1/7 as a decimal. How long does it take to repeat?

Compute

- 142857 * 1 =
- 142857 * 2 =
- 142857 * 3 =
- 142857 * 4 =
- 142857 * 5 =
- 142857 * 6 =
- 142857 * 7 =
- What do you notice about your answers?

Expand 1/13 as a decimal. What 13 multiplication problems do you think we will ask next? Write them, solve them, and tell us what you see.

What is the prime factorization of 999,999? Any coincidences?

Write some other problems like these. Use a computer or calculator if you want! (Looking at 1/17 is an interesting place to start.) (Looking at the prime factorization of 9,999,999 leads to craziness.)

**Problem 3.**

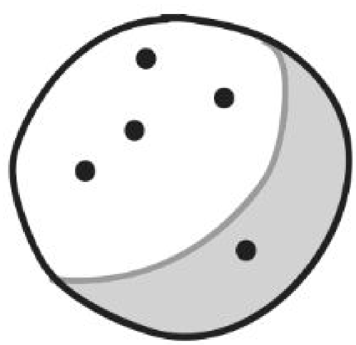
Your friend showed you the following trick. She gave you numbers 1 to 100, each written on a separate card, and asked you to hide one card of your choice. After that, she asked you to show her remaining cards, one at a time, in any order you want. When you were done, she thought for a short time and told you which number was missing. You asked her if she actually remembered all 99 numbers you showed her. She answered that she only had to remember one number after each card shown to her. After each card, she would compute the new number and forget the old one, and the number she had to remember never had more than 4 digits!

(a) Can you come up with a strategy to perform your friend's trick? It should work, no matter what number is hidden. Specify what you would do after each card is shown and how you would figure out the missing number.

(b) What if now your friend hides two numbers of her choice? Can you come up with a strategy to figure out what these numbers are? Again, after each card is shown, you shouldn't have to remember too much. The less you have to remember, the better!

**Problem 4.**

Imagine a perfect sphere. (No real sphere is perfect, but any real sphere, maybe a plain rubber ball, is really helpful for learning about a perfect one.) If we cut a perfect sphere exactly in half, we have two "hemispheres" — and there are infinitely many ways to do this. When we include the points on the cut we have a "closed" hemisphere. (What, actually, are the ways you can cut a perfect sphere exactly in half? Draw some of your own! Try this out on a ball!)



It is pretty surprising, but true: No matter how five points are selected on the sphere, amazingly, there is always some way to find a closed hemisphere that has at least four of the points!! Prove that this is true, always.

5. At C&!, the teachers are always trying to scam the kids.

Too bad the kids usually see what's coming from a mile away... But we're pretty sure you'll go for this!

*You pay $1 to play this game. You pick a number from 1 to 6 and roll three (perfectly fair! six-sided!) dice. If your number comes up on any of the dice, you get back your $1 and you get a prize as well: — another $1 if your number shows once, — another $2 if your number shows twice, — or another $3 if your number shows on all three dice!*

That's got to be a great deal! There are three dice, and there's a 1 in 6 chance your number will appear on any of them. So about half the time you'll end up paying $1, but the other half of the time you'll get back your money and get at least that much again on top, right? So it really is a great deal isn't it? Is it really? Explain.

Got any great deals for us?

**Problem 6**.

Complete the following, by filling numbers in the blanks so that the statements

are all true:

The number of 1's in this puzzle is: ________

the number of 2's is: ________

the number of 3's is: ________

and the number of 4's is: ________

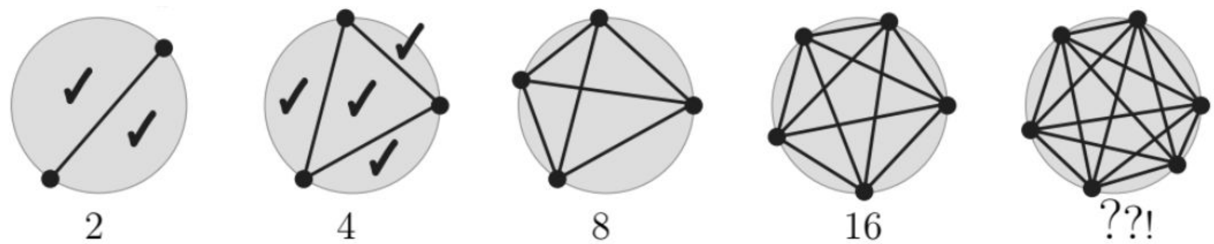
.

Of course the numbers that are printed in the puzzle, and the numbers that you fill in, both are counted up!

(Is there only one way to do this? What if you make a longer paragraph or skip some of the numbers? What puzzles can you make, and solve? What puzzles can you make and prove cannot ever be solved?)

**Problem 7.**

You think you see the pattern in the numbers in the figure below. In each drawing, lines connect some number of points on a circle, and the number at the bottom of the drawing is the number of regions they divide the circle into, the most possible.




But what is the number of regions in the last drawing? (Redraw it a lot and keep on counting, to be sure.) Sometimes you just have to take things a little further. Do some more examples, with 7,8,9 etc. points[1] around a circle, counting the most possible number of regions divided by lines between them. (In your drawings, it doesn't even matter if your lines are very straight.) So: What is the correct pattern? Why?

**Problem 8**.

We're going to give you a little programming language. Your mission, should you choose to accept it, is to use this language to write the largest number you can in 140 characters or less.

Here are the rules of the language:

- You can write mathematical expressions with numbers and the symbols +, -, *, /, (, ), and ^ (exponentiation) and they have the usual meaning. So, for example, you could write this little program:
    - `(3^2)*(2^3)-144/2+1`
    - If we run this program we'll get the value `1`.
- You can define new *functions* that act on *variables*, then apply the function to a number. Here's how it looks:
    - `double(x)|x+x;double(12)`
    - This little program defines a function called "`double`" that doubles a number, then it uses the function on the number 12. If we run this program we'll get the

[1] Oh, hey, what about the number of regions when there is just one point? Does the "pattern" hold?

value `24`.

- We can test if something is true and output a different number if it is. This one is a little tricky, but try to understand this example:
  - `f(x)|if(x=0) then 1 else 1000;f(2)`
  - This program will say "hey x is 2, does 2 equal zero? I don't think so! So I'll return 1000".
  - If we asked for `f(0)` we would have gotten the value 1.

Those aren't very big numbers. Let's do something more fun:

```
factorial(n)|if(x<1)then 1 else
x*factorial(x-1);factorial(5)
```

What does this do? What about this:

```
f(n)|if(x<1)then 1 else x*f(x-1);f(f(f(f(f(9)))))
```

Are we getting big yet?

OK, now you are ready. Can you change the factorial example to make it grow faster? Learn about Knuth's "up arrow notation" and see if that gives you any ideas. How about the Ackermann function?

Write the biggest numbers you can! Give a couple of tries and tell us why one is bigger than the other, or tell us that you don't know! Did you find the biggest possible number? Can you prove it?

You'd rather write your program in Haskell? Python? Assembler? Go for it! You can even make up your own programming language if you like.