



CS Problems

By [\[\[Reisen\]\]](#), Source: [0x32767.github.io](https://github.com/0x32767), Discord: [reisen](#)_____

Introduction

This is a document dedicated to a bunch of computer science problems. These problems range from “easy” to “phantasm” ([2hu](#) reference). They also include tips and solutions. I encourage you to minimize using the internet, and to write as much of the solution as possible by yourself. My only requirement is that **AI of any sort is prohibited** (GH Copilot, ChatGPT, et cetera...). Good luck and happy solving.



Problem 1, Easy

Create a program that checks if a user input would be a valid periodic table symbol. Said symbol would need to:

- Be either 1 or two characters long
- Have the first character be uppercase
- Have the last character (if there is one) be lowercase

Tip

You will need to check the length of the string at one point, check the case of the second character there.

Problem 2, Easy

Create a program that will find the n th term of a linear sequence of integers. The program must be a function that takes a list of integers and will return the n th term as a string.

Example: [7 13 19 25 31]

Expected: $6n+1$

Tip

You can actually find the n th term of any sequence given the first two numbers, just use the first two elements of the list.

Problem 3, Easy

Bob has a function that takes one parameter and returns the value of some sequence at that index. For example calling his function with 10 would return 20. Create a function that uses bob's function and plots the result. For example one of bob's functions produces the following output [1, 2, 3, 4, 5, 6, 7]. On the printed graph it should print something like:

```
|#  
| #  
|  #  
|   #  
|    #  
|     #  
|      #  
|       #  
+-----
```

Note that

- The plotting function must be called `plot`
- The program can hardcode bob's function



- Bob's function will never return a float
- The plot must be 100% accurate

Tip

Bob's sequence has infinitely many terms, you can call his function with the value of, `100000000000` and it would still return a value. Use a `for` loop.

Problem 4, Easy

A very trivial question in computer science is fizz buzz. Here you need to print the numbers 1 to 100, if the number is divisible by 3 print 'fizz' if it is divisible by 5 you print 'buzz' if it is divisible by both 3 and 5 print 'fizz buzz'.

Problem 5, Easy

Cirno is making a game about her perfect math class. In the game you are given an arithmetic question such as $[a] + [b] = ?$ (where $0 \leq a \leq 9$ and $0 \leq b \leq 9$). The user is then given a choice A and B, one of which is a random number and the other is correct. The user must select the wrong answer unless the answer is 9. For example,

a = 2
b = 4
A = 3
B = 6

In this scenario the user needs to select option A because $a + b \neq A$. However:

a = 7
b = 2
A = 8
B = 9

Here the user would need to select B because $a + b = 9$, and 9 is special.

Problem 6, Easy

George is working on $3x + 1$ or the Collatz Conjecture. The conjecture is as follows, choose any positive integer and apply the following rules:

- If the number is odd multiply it by 3 and add 1
- If the number is even divide the number by two

Every number should eventually end up in the following loop 4, 2, 1. Georg wants to brute force the numbers 1 to 1000. Your program should output the number of operations (every time you apply one of the rules) for every given number before it ends up as 1.



Problem 7, Normal

Mark is making the next big social media website. He wants to be able to display the number of views a video has. To make it easier to read the number of views he creates a number abbreviation system. Examples are: 89,288 being abbreviated as “89k” or 1,000,000 to “1M” and so on. The number should also be truncated, never rounded up.

Tip

Convert the number into a string, then group the string in chunks of three. For example the number `4357978345345` could be converted into `[4 357 978 345 345]` (the number has been converted into an array of string).

Problem 8, Normal

John has a website where users can make an account. Using an official captcha would be very difficult to implement so John decides to make his own. John decides to have the users of his website recognise a string as a word. Similar to passwords, random letters should be replaced to fool bots. For example, an `@` can also represent an `a`, and a `5` can replace a `s`. Additionally, random letters should be capitalized to make it even harder. John’s program should choose a word at random, mangle it and print it to the user. If the user inputs the same word (the original version as lowercase letters) then they get access, otherwise they are denied access.

Example words:

- python
- computer
- science

These are the mangled words:

- PyT0n
- c0|V|pu+er
- \$c!ENce

Problem 9, Normal

One of the cornerstones of mathematics is multiplication. Let's just forget that it exists! Create a function called `multiply` that takes in two numbers and multiplies them together. However, you may not, in any form, use multiplication or any function that multiplies numbers. You may want to use bit shifts and repeated addition to accomplish your goal. In this section you also need to try to compute the answer the fastest.



Problem 10, Normal

Mike has a function that returns the value of a quadratic given an input, assume this function is called `get_quad` and takes in `x`. Use a brute force algorithm to find the lowest point of the graph. You may assume that.

$$ax^2 + bx + c$$

$$a > 0$$

$$b > 0$$

$$c > 0$$

(This means that all parabolas formed given these rules will always have a **U** shape) The algorithm should take as little time as possible to find the lowest point on the graph. Additionally the error should also be reduced to a reasonably small number, (`0.01`).

Problem 11, Normal

Owen is working on a website. In a url you can have http parameters, these parameters can provide specific information to the website. These parameters come after a '?' in the URL of a web page. For example, a website with not parameters:

```
https://github.com/0x32767
```

And a website with parameters:

```
https://www.youtube.com/watch?v=xX5iOYCJmBI&list=PLZPZq0r_RZON1eaqfafTnEexRzuHbfZX8&index=1
```

A http parameter is defined by a name, followed by an equals ('=') followed by the value, if there are more parameters there will be a '&' character. In the youtube example, there are three parameters.

```
v=xX5iOYCJmBI
```

```
list=PLZPZq0r_RZON1eaqfafTnEexRzuHbfZX8
```

```
index=1
```

NOTE: In the url there are '&' characters separating the parameters.

Here you can also see the '=' character separating the name and value.

Your job is to create a function called `parse_http_params` that takes in a http parameter string and returns a 2d array. The array will have the name of the parameter as the first element and the value as the second element. For example:

```
v=FtutLA63Cp8
```

Would return:

```
["v", "FtutLA63Cp8"]
```

And:



v=XKHETdqhLK8&t=2s

Would return:

["v", "XKHETdqhLK8"]
["t", "2s"]



Problem 12: Hard

Fred needs to use ERL (Exam Reference Language) for his homework, but he doesn't like ERL. So he decides to write a program that will translate python to ERL. To make his life easier he decides that he will only use functions, procedures while loops and if statements. To help him he will also leave comments at the end of loops such as `# END LOOP` to tell his translator that it is the end of a loop, he also does the same with functions and procedures with `# END FUNC`, `# END PROC`, `# END IF`, `# START PROC`, `# START FUNC` respectively. An example of one of the python programs is below:

```
# START PROC
def test_func(name):
    print("Hello " + name)
# END PROC

# START FUNC
def add_one(number):
    return number + 1
# END FUNC

i = 0

while i < 10:
    print(i)
    i += 1
# END WHILE
```

And the program should output the following code:

```
procedure test_func(name)
    print("Hello "+name)
end procedure

function add_one(number):
    return number + 1
endfunction

i = 0

while i < 10
    print(i)
    i = i + 1
endwhile
```



Tip

The easy approach should be to replace all instances of a keyword with its counterpart. This works but is sloppy. Instead you will need to write a lexer, a program that will break down another program into smaller bits, such as strings and numbers. You could then iterate through the generated tokens and then replace them. Finally you could then convert the tokens back into a string. This approach handles most possible edge cases that raw string replacement wouldn't. Such as having a variable name with "def" in it.

Problem 13, Hard

Jeff wants to check his math homework. He is studying lines on graphs. Jeff wants the graphing calculator to create graphs for setting the X axis on a graph to a constant ($x = 1$), setting the Y axis as a constant ($y = 5$). He wants to be able to input at least two of these expressions into the graph. He wants the output to be printed out to him. Because, Jeff is also rather fussy; if two lines cross, (for example, $y = 1, x = 2$ will cross at $[2 \ 1]$), he wants a +, for all other instances he wants either a hyphen (-) or a pipe printed (|).

Note that

- Bob will always input his lines in the following, `[AXIS] = [NUMBER]`
- Bob doesn't need to deal with negative numbers
- You just need to plot the graph up to a 10 by 10 display

Tip

Generate a 2d array with X, Y and character. X and Y are self-explanatory (X and Y coordinates on the graph where a line goes through) character is either the hyphen or the pipe. When generating the graph use a nested for loop and check if the current coordinates match up. Then perform a linear search through the 2d array; if there are multiple entries with different characters e.g. `[... [2 3 '-'] ... [2 3 '|'] ...]` then print the '+' otherwise just print the stored character.

Problem 14, Hard

Paul is learning how to play black jack. In blackjack the aim is to get as close to 21 points as possible without going over. The number of points is the sum of the numbers on the cards, where picture cards are worth 10 points, and an ace can either be worth 1 or 10 points. Paul wants to create a program that calculates the probability of the next card not being brought, causing him to go over 21. To make the program more accurate he inputs a string of cards he has in his hand where a 10 is represented by a 0, an ace by an A and a picture card by a P and all other cards are their respective numbers.



Problem 15, Lunatic

Joe has a bunch of long algebraic divisions. She wants to make a program that will collect the like terms and simplify the expression. Her expression will only include multiplications in the numerator and denominator. Below is an example of an input:

```
a^2 b^4 a^2 c c / a^3 b b c^2
```

Here is the expected output:

```
a b^2
```



Problem 16, Phantasm

Lucy is preparing for a DofE expedition. She has a list of coordinates of checkpoints she needs to go to. For health and safety she needs to arrive at each point 2 hours from when she departed from her last point. She needs to go to at least 5 checkpoints. Lucy has used the grid figures on a map to represent the location of each checkpoint. Lucy starts at the point (0 0) on the map and her objective is at (15 15). The length of one square on her map in real life is 0.5km. Assuming that Lucy can walk 0.5 km/h. The example below shows some example data:

```
[ 1 2 'A' ]  
[ 2 2 'B' ]  
[ 4 4 'C' ]  
[ 5 7 'D' ]  
[ 7 10 'E' ]  
[ 7 7 'F' ]  
[ 14 13 'G' ]  
[ 12 13 'H' ]  
[ 9 7 'I' ]  
[ 10 9 'J' ]  
[ 11 11 'L' ]  
[ 15 15 'GOAL!' ]
```

An example program would output:

```
A -> B -> C -> E -> Z -> J -> GOAL!
```

Problem 17, Phantasm

Jess has a 2d array with letters, where the array is a fixed 26 by 2 array. Here is an example:

```
[ ['A', 'M'],  
  ['B', 'J'],  
  ['C', 'Q'],  
  ['D', 'L'],  
  ['E', 'I'],  
  ['F', 'D'],  
  ['G', 'S'],  
  ['H', 'O'],  
  ['I', 'A'],  
  ['J', 'G'],  
  ['K', 'U'],  
  ['L', 'K'],
```



```
['M', 'F'],  
['N', 'B'],  
['O', 'P'],  
['P', 'N'],  
['Q', 'X'],  
['R', 'Z'],  
['S', 'E'],  
['T', 'H'],  
['U', 'C'],  
['V', 'R'],  
['W', 'T'],  
['X', 'V'],  
['Y', 'Y'],  
['Z', 'W']]
```

The array contains all letters of the alphabet on the left, and a scrambled version on the right. In each sub array Jess calls the first element the key and the second element the value. For example, in the array the first subarray ['A', 'M'] has 'A' as the key and 'M' as the value. Within this pattern there are loops. If you pick a key, take the value and find a new value with the key as that value, you can form a loop. In the example above:

```
A -> M -> F -> I -> A
```

So we have a loop. Create an algorithm that takes in one letter as an input and finds the loop associated with that letter, the algorithm should also output what letters the loop consists of.

Problem 18, Phantasm

The roman empire has returned in the 20th century and are looking to make some infrastructure. The Romans have created a 2d array of heights in an area, from sea level. These Romans have found a large lake on the top of a mountain and want to get it to their settlement. The settlement is one meter above sea level and the mountain is 100 meters above sea level. In between the settlement and the mountain is a lot of terrain of varying heights, represented by the height above sea level in their 2d array. Create a program that finds the optimum path from the lake on the mountain to the settlement. Do note that:

- Water can only move down hill
- Romans are not a big fan of their drinking water being used to make waterfalls, the drop from one position to the next should be a maximum of 5 meters.
- The Romans are short of building materials so the path should be the optimum one.
- Romans don't know how to construct diagonal bridges

Here is an example of their 2d array:

```
[[21, 20, 1, 2, 55, 180],
```



```
[22, 19, 5 , 9 , 55, 101],  
[23, 20, 15, 10, 55, 100],  
[25, 77, 80, 81, 40, 98 ],  
[27, 75, 34, 85, 94, 97 ],  
[31, 71, 34, 89, 90, 50 ],  
[35, 69, 65, 61, 60, 55 ],  
[37, 40, 43, 46, 50, 54 ]]
```

The program should output an array with directions as to where to build. These directions are given as compass directions (North, South, East, West). The example would have the solution:

```
['S', 'S', 'W', 'S', 'W', 'N', 'N', 'W', 'W', 'S', 'S', 'S', 'E', 'E',  
'E', 'E', 'S', 'W', 'W', 'W', 'W', 'W', 'W', 'W', 'N', 'N', 'N', 'N',  
'N', 'E', 'E', 'E', 'N', 'W', 'N']
```

You may assume that:

- There is only one mountain that is 100 meters high.

Problem 19, Phantasm

Lee has a friend who can beat anyone at double OXX. Lee has arranged a competition with his friend where we will try and beat his friend at a 10 by 10 board. Create a program that takes the current board as a 2d array and computes the best move. The game board is taken as a 10 by 10 array where a ' ' symbolizes no move there and a 'X' and 'O' correspond to the respective players.

Problem 20, Phantasm

Carl is working on a robot. The robot has specific preplanned directions it can go in. Given two points, a start and target, find the most efficient combination of vectors to get the robot from start to finish (or as close to the finish). The robot can only move in these directions:

- (+5, -9)
- (-8, +1)
- (+1, +1)
- (-4, -6)



Last word

Hello dear reader and thank you for doing these questions. I really hope that you found at least some of the questions fun, and feel as if your programming skills have improved as a result of doing them. Best of luck, [[Reisen]]

License

CS Questions © 2023 by [[Reisen]] is licensed under **CC BY-SA 4.0** (cc)

- Royalty free
- The work may be used for commercial use
- Please provide me ([[Reisen]]) with original credit for the work
- You may adapt/build upon and take inspiration from said work
- Any derivative direct works must be under the creative commons license

When I referred to `direct` I was referring to building upon the document, not using it. If you intend to use the document to make videos, blog, educate, et cetera, you may use the document under any conditions.