Sample problem Bank

Very Easy

Output “hello world”
Output all numbers between 0 and given input
Output all odd numbers between 0 and given input
Make a switch case menu
Output the higher of two inputs
Output if given input is even or odd
Drinking Age validation

Easy

Bool isDivisbleBy(int,int)
Bool isLeapYear(int)
Int SumArray(int[] arr,int size)
Int FindIndex(int[],int)
Int GetHighest(int,int);
Report*

Medium

String toUpper(string)
String toLower(string)
string swapCase(string)
Bool containsUpperChar(string)
Bool containsChar(string,char)
Int GCD(int,int)
String outputRightAngleTriangle(int); //given height
Int inputHandling(string,string);
Bool isPalidrome(unsigned long long);
Int CountVowels(string s);
Int countConstants(string s);
Int GCDOfVowelsAndConstants(string);
Int PlaceBulkOrders(int);

Difficult

Bool containsSubstring(string,string,bool) //case sensitive
String outputTriangle(int height)
String ReverseString(string s)
Number guessing game
Int PlaceBulkOrdersWithLimits(int);

Hard

Sort an int array in ascending order
Binary search on sorted array
Given a string, order all letters alphabetically and output it
Make a program that uses Clock() to act as a timer
Make a function, that given an address of a struct, alters it.

Hard Word Problems

Make a program that allows a user to place a bulk order. It asks the user how many orders they would like to place, and then outputs the price. The calculation for the price is
"50$ for the first unit, and a 1$ increasing discount on each subsequent order".
So if I order 1 unit = 50$
2 units = 50$ + 49$ = 99$
3 units = 50$ + 49$ + 48$ = 147$
4 units = 50$ + 49$ + 48$ + 47$ = 194$
And so on….
You only need to output "The total price is: <totalPrice>$"

Make a function to determine if the current tic tac toe board setup wins or loses.
The function is given a 2D char array where each index has a 'x','o' or ' '. The function returns the char of the winner, if there is no winner, it returns the empty char ' ' or '\0'.
Example

```
char board[3][3] = {{'x','o','o'},          //    char winner = checkWin(board);
                    {'x',' ','o'},          //    cout << winner;
                    {'x',' ','x'}};         //     'x'  would be the winner here
```

Very advanced difficult hard problem

The concept of cellular automata can be seen in Conway's Game Of Life (see youtube). The algorithm in concept is rather simple, a 2D array exists as the world, a single value in this array, depending on the value of it's neighbors, becomes a new value after one frame. For example, in Conway's game of life, the following rule applies:
"Any live cell with fewer than two live neighbours dies, as if by underpopulation."
The values in conways game are simply booleans. The location contains a live cell or not. Each alive cell checks it's surroundings and see's if there are enough neighbors, else it dies.

These simple rules define Cellular automata, and can be used to build beauitful things. In the game "Notia" or "pixel sandbox" it is used to simulate a living world. A pixel's color determines it's value, if a red pixel (fire) is next to a brown pixel (wood), the brown pixel becomes red.

In the simulation Langstons Ant, it turns one color into another as well.
See if you can build a cellular automata, it can be anything, a 2D array or bools or 2D array of chars, which has a set of rules which are calculated each frame.