

Epic Games, Programming Exercise

Thank you for taking the time to complete our test!

Instructions:

- Please give an approximate time that it took you to solve each question.
- State your assumptions when you solve each problem.
- We request that you keep the answers to these questions private. If you post them to a web link, github account etc. please do your best to make them private.

Format:

- If possible, please deliver your response in a plain text file or document format, zip file is fine
- We do not accept PDFs

All code is run through a test suite to test its production quality. These tests affect whether or not our interview process continues, so please take them seriously. Note that there may be multiple ways to solve some of the questions.

1) You have been given 2 special, extremely rugged Xboxes. You are in an office building that is 120 stories high. Using the fewest possible number of drops from windows in your office building, determine the highest floor you can drop an Xbox from and have it survive: for example, they might be able to take the drop from the 30th floor, but not the 31st. You can break both Xboxes in your search. State the worst case number of drops needed and explain how you arrived at that answer.

2) In C++, which class methods are automatically generated by the compiler, under what circumstances, and what is their access protection and signature? Explain why you may or may not want to provide your own implementations of these methods.

3) Write the code for the following function, without using any built-in functions except malloc or operator new..

```
char* itoa(int Value, int Base);
```

where the returned value is allocated on behalf of the caller, value is the integer to convert, and base is octal, decimal, or hex.

Your implementation is expected to be robust and production ready

4) Write a function with the following signature that, given a matrix of integers, builds a string with the entries of that matrix appended in clockwise order. For instance, the 3x4

matrix below:

2, 3, 4, 8

5, 7, 9, 12

1, 0, 6, 10

would make the string “2, 3, 4, 8, 12, 10, 6, 0, 1, 5, 7, 9”.

Your implementation is expected to be robust and production ready

```
void BuildStringFromMatrix(int* Matrix, int NumRows, int NumColumns,  
char* OutBuffer)  
{  
    // Your code goes here  
}
```

OutBuffer is guaranteed to be valid and large enough to hold all of the data.