

Welcome to the HBS programming test! First of all, thanks for taking the time to complete this step of our hiring process. We take the test quite seriously as it is an important way for us to evaluate your coding acumen. It's important to us that we hire engineers that are not only great problem solvers, but who can also put fingers to keyboard when it counts!

Even though the items on this test may be small in nature, please treat them as if they were problems in a large-scale, ship-quality product. This means you should exercise good engineering principles regarding performance, memory usage, and the time taken to implement a particular solution. Your code should be readable and well-documented, and if any assumptions are deemed necessary, please explain them clearly.

There is no explicit time limit, but we generally expect this to take less than a week. We will accept pretty much any deliverable format you would like to provide (within reason), but please do provide digital copies of your answers in the form of source code. If you would like to also provide a fully functioning executable per problem, that's okay too, just please remember to also include the source. The goal is for you to convey to us that you can concisely and clearly solve a problem using proper engineering principles, the specific choice of language or syntax is less important to us (unless a particular question calls for it).

Please include solution files so that we can compile and run your work.

All tests **must** be submitted as a single zip file, named '*First-Last-Test.zip*' with each answer in its own folder. Any coding questions should either be in a Visual Studio project, a Unity project, or include a Windows-compiled executable. (Other than the code review question, which we'll see on gitlab.com)

Good luck!

Question 1

Given a structure defined as follows:

```
struct Rect {  
    float minX;  
    float minY;  
    float maxX;  
    float maxY;  
};
```

```
float DistanceToRect(float x, float y, Rect& rect);
```

Please write a function that will return the distance from a single 2D point to the nearest edge of a rectangle, where a point that is inside the rectangle will return a distance of 0.

Question 2

Write a program in C# that will accept multiple CSV (comma separated values) files from the command line, parse them, and return both the sum and mean (average) value of all numbers across all files, in the column labeled "Cost, Initial". Please do not use any standard CSV parsing library, we would like to see your own implementation!

When submitting, include your solution files and any test cases you develop.

Assumptions

- The first row of each CSV file will contain the column headers that can be used to identify the column to extract values from (which may vary from file to file to file).
- The values themselves may or may not have punctuation and/or formatting (such as \$5,000.23 or 123). All values can be assumed to be in the same denomination (e.g. US dollars).
- Files may use either Windows or Unix style newlines, and you should write a parser that will accept both.

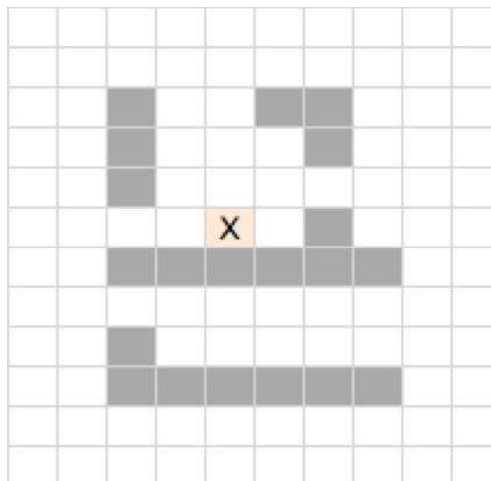
Question 3

The answer to this question **must** be supplied as a complete Unity 2019.3 (any release) project, ready to be built. Please compress and send your entire Unity project, you may want to delete the Library folder in order to reduce the final size.

The intent of this question is to express your skills in the Unity engine. Please use whatever systems you are comfortable with in order to present your strengths!

Given a 2D grid-based map (such as the example below), write a program that will find the shortest path and total movement cost to every *valid and accessible* location on the map for a single turn, starting at point **X**, using the following criteria:

- You may move only in the main 4 cardinal directions (north, east, south, west), not diagonally.
- Each move costs 1 movement point.
- You have 6 movement points to spend per turn.
- You may not move through (or onto) a shaded cell.



Question 4

A co-worker has just made their first submission for a new telemetry package and you've been asked to do a code review. There are many layers of issues here- some obvious mistakes, some more subtle architectural / maintainability problems, and some issues where there's no clear right or wrong answer and just might be an interesting discussion.

Remember, this is meant to be the base for a reusable module, so we're interested in your ability to spot both immediate bugs and long-term issues. Also, please imagine you're writing to an actual developer, not a robot. (i.e. Show some tact when calling out issues.)

In order to be added to the code review, you must have or create a (free) account at [GitLab.com](https://gitlab.com). Please send your username to giteval@hbs-studios.com and we'll promptly add you as a reviewer to a clone of that project.