# Coding Challenge Project Notes

Robert Hansen  
13 August 2023

**Program Overview and Setup**

The software is written in C# and runs in the Visual Studio environment. I am using Visual Studio 2022 and .Net 6.0.

I am also using the NUnit testing extension which is available from NuGet; this is not required to build and run the application.

**Assumptions**

1. The world X and Y locations are integers.
2. The number of central fill facilities to create is between 5 and 180. This was chosen to make the simulation more interesting. These values can be changed by modifying the constants at the beginning of the program.
3. The maximum medication cost is $2000. This is a limit placed on the simulation data; it is not a limitation of the processing software. This can be changed using the constant definition at the beginning of the program.
4. I have hard-coded 3 medications for each central fill facility, named A, B, and C. In a real application, we would probably create a medication class and use a list of the medications available at each facility. This would allow different medication names and any number of medications.
5. The central fill facility IDs are integers generated starting from 1.
6. The world map display assumes the central fill facility ID is not bigger than 3 digits. (There is no limitation in the processing software.)
7. When adding central fill facilities to the world array, I do not check to see if there is already a facility at that location; the previous one will be overwritten. In a real application, this would need to be handled appropriately. (see comments related to this below)
8. If there are more than three nearby central fill facilities near the target user location (with the same distance), the software will report the first one(s) found.
9. If there are medications with the same cost, A will be reported over B and C, and B will be reported over C.

**Changes to support multiple central fills at same location  
 and  
Changes to support a larger world size.**

Both of these issues can be addressed with the same change. The current implementation uses a two dimensional array to represent the world coordinates. A better approach would be to simply have a list of central fill facility objects (with each one having properties indicating its location). This way, the world size is irrelevant. It also allows multiple central fill facilities to be at the same location.

Since the software would sort the entire list of central fill facilities to compute the distances to the target, this could be slow if there was a large number of facilities. The sorting process could be short-circuited since we only need to find the 3 shortest values. Also, there may be a way to preselect a subset of facilities based on location. For example, I would expect the central fill facility information to be in a database, and there may be additional information related to location that could be used.