A. A point-form list of the steps required to create a new project in Visual Studio, add new files, compile and run a command line program.

New Project

- File
- New
- Project

Add New Files

- In the solution explorer window on the right hand side.
- Right click on the heading of type of file you would like to add.
- Click "Add"
- Click "New Item"

Compile and Run Command Line Program

- Add a new source code file.
- Add the following code or any working code.

```
int main()

{

        return 0;

 }
```

- Click the debug menu.
- Click start without debugging.

B. A point-form list of the steps required to create a break-point location in the program, and run the program in using Visual Studio's Integrated debugging system so that program execution stops at the nominated point.

- On the left hand side of the editor, there will be a set of line numbers.
- To the left of the line numbers, there is a slight colour change(usually gray for standard white theme) which runs vertically with the numbers.
- Click in this section and a red filled circle will appear. This has set a break point in the code.
- For the program to stop at this line, ensure that you run with debugging.

C. Note how to inspect the value of variables during debugging

- Debug a program until it reaches a breakpoint.
- In the bottom left hand corner of the screen, there should be a window called "Autos". If there isn't, Press Ctrl+Alt+V then A. To enable this window.
- You can now see the value of the variables which are in scope.

Spike No. : 3

Title: Intro to Visual Studio

Author: Parth Madhani , 101901858

Goals / deliverables:

- Comparison Report

Technologies, Tools, and Resources used:

- Visual Studio 2017 IDE
- Canvas

Tasks undertaken:

- Writing down all the functionality with knowledge gaps (new project, new files, debugger etc.).
- Search help section to ensure the different ways
- Write down easy to understand steps for report.

What we found out:

I found out how to use the Visual Studio IDE effectively to create projects, work with different files/resources and use the debugger in order step through the software so as to watch how the code manipulates the underlying data so one can find errors.