1a- El lenguaje assembly permite escribir estas instrucciones como un texto, y un compilador traducirá las instrucciones al lenguaje máquina que hemos comentado. Los lenguajes ensambladores ofrecen un control muy preciso de todas las instrucciones y tareas que realiza el procesador. El código fuente escrito en assembly está pensado para que se ejecute para un procesador concreto.

1b- D es un lenguaje de propósito general y para aplicaciones. El lenguaje D retiene la capacidad de escribir código de alto desempeño y poderlo ligar directamente con el API del sistema operativo y el hardware del equipo. Tiene además una optimización agresiva en el compilador.

2a- Assembly ayuda descubrir errores en el código de lanzamiento en escenarios de producción reales . También clasifica problemas en el código de producción de proveedores externos que se les enviaron para que los resolvieran. Dejando a un lado el código externo, también ha demostrado ser útil para depurar problemas que ocurrieron solo en compilaciones sin depuración de complementos de Maya, por ejemplo. El conocimiento de ensamblaje es de mucha utilidad para la depuraciuon de codigo, a veces un compliador crea codigo de ensamblaje incorrecto y poner el codigo en un depurador ayuda a encontrar la causa. Los optimizadores de codigo tambien cometen errores a veces. Otro uso para ensamblaje son las interfaces o reparaciones de codigo del cual no se tiene codigo de origen. Densamblar permite cambiar o reparar ejecutables existentes. Ensamblar es necesario si se desea saber como el lenguaje elegido funciona, porque algunas cosas son lentas y otras rapidas. Finalmente, el conocer sobre codigo de ensamblador es indispensable para diagnosticar malware.

2b- Las implementaciones actuales de D compilan el código directamente a código máquina para lograr una ejecución eficiente. Cambios al lenguaje no se hacen regularmente desde la versión 1.0. El diseño está virtualmente congelado, y nuevos lanzamientos se concentran en resolver bugs existentes. El compilador oficial de Walter Bright define el lenguaje. Esto permite incluir código específico de la máquina con código D, una técnica comúnmente usada por programadores de software de sistema para acceder a características de bajo nivel necesarias para interactuar directamente con el hardware, permitiendo escribir software como sistemas operativos y drivers. D incluye soporte para comentarios de documentación, pero hasta ahora sólo el compilador entregado por Digital Mars implementa un generador D

3a- ide para assembly Sasm, Sublime, Vim, Visual Studio Code, Atom, IntelliJ IDEA. (and other JetBrains IDEs)

3b- ide para D podemos utilizar visual studio y codeblocks

4- denolet x = 4;let y = 5;let z = x + y; consolge.log(z);pythonx=4y=5z=x+yprint(z)