

1.1

- a) Apple
- b) IBM Personal Computer
- c) Programas
- d) Unidade de entrada, unidade de saída, unidade de memória, ALU, CPU, armazenamento secundário
- e) Linguagem de alto nível, assembly e linguagem de máquina
- f) Compiladores
- g) UNIX
- h) Multitarefa

1.2

- a) Editor de textos
- b) Pré-processador
- c) inclusão de outros arquivos no arquivo a ser compilado e na realização de várias substituições de texto
- d) Linker
- e) Carregador
- f) ./a.out

1.3

- a) Hardware
- b) Software
- c) Hardware
- d) Software
- e) Hardware
- f) Software

1.4

Para proporcionar um melhor entendimento, evitando erros humanos e agilizando o processo de programação. Além disso, sua funcionalidade é abstraída das especificidades da máquina.

Pois excluiria a necessidade de compilar o código, processo esse que pode tomar um tempo considerável do computador e atrasar a execução do programa.

1.5

- a) CPU (Central Process Unit)
- b) Programação
- c) Linguagem simbólica (assembly)
- d) CPU (Central Process Unit)
- e) Memórias
- f) ALU (Arithmetic Logic Unit)
- g) ALU (Arithmetic Logic Unit)

- h)** Linguagem de alto nível
- i)** Linguagem de máquina
- j)** CPU (Central Process Unit)

1.6

- a)** Verdadeiro
- b)** Verdadeiro

1.7

Standard Input - Fluxo padrão de entrada de comandos inseridos. Significa que um dado (frequentemente texto) está indo para um programa.

Standard Output - Saída padrão de dados. O programa requisita a transferência de dados através de uma operação de "escrita". Nem todos os programas precisam de saída.

Standard Error - Saída padrão utilizada pelos programas para envios de mensagens de erro ou de diagnóstico. Este fluxo é independente da saída padrão e pode ser redirecionado separadamente.

1.8

Porque ela propõe uma representação mais fácil de compreender devido à relação de cada elemento em termos de um objeto, ou classe, podendo ser comparada ao mundo real. Além disso, o software que elas produzem é mais inteligível, mais bem organizado e mais fácil de manter, modificar e depurar, o que dá mais liberdade individual em um trabalho coletivo, reduz o tempo de desenvolvimento de sistemas complexos e realiza um processo mais natural do que a programação procedural, resultando em importantes ganhos de produtividade.

1.9

- a)** FORTRAN
- b)** COBOL
- c)** Pascal
- d)** Ada
- e)** BASIC
- f)** C#
- g)** C
- h)** C++
- i)** Java