Software development tools

Professor Hossein Saiedian

EECS 348: Software Engineering

Spring 2023

KU THE UNIVERSITY OF KANSAS

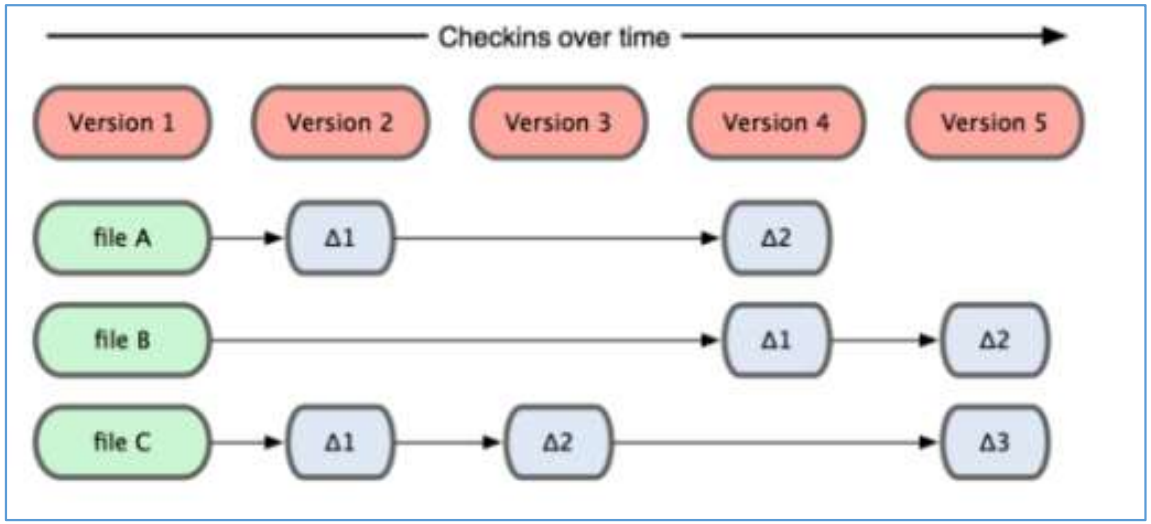
A brief intro to Git and GitHub

KU THE UNIVERSITY OF KANSAS

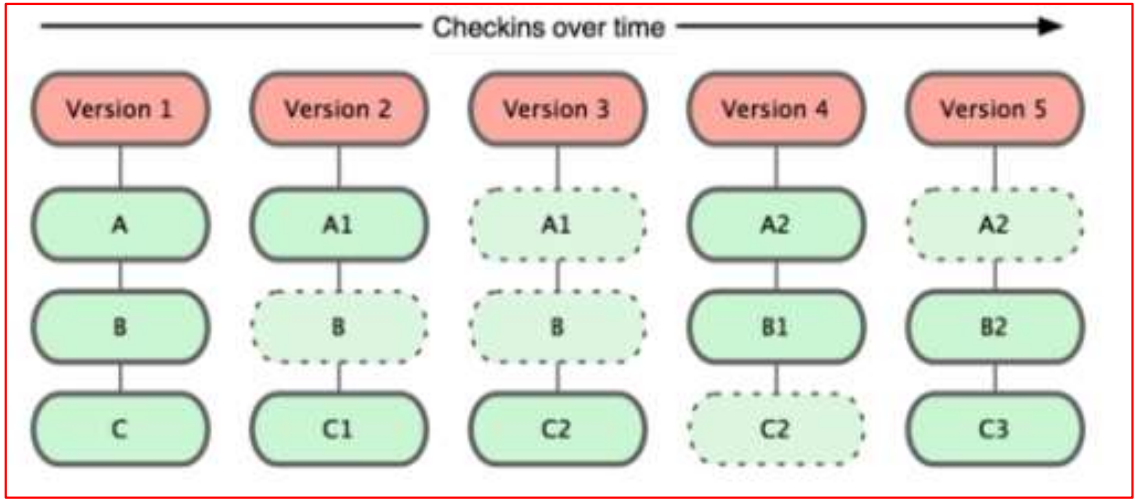
- Git is a version control system (VCS)
- Allows to maintain multiple versions of a code base
  - Keeps a history of previous changes
  - Let's you see the changes you make to your code and easily revert them
  - Sometimes across multiple developers
    * Collaborate with other developers
    * Push and pull code from repositories such as GitHub
- Available on Linux cycle servers
- Available for installation on Windows and Mac machines

Snapshots, not differences

- Older VCSs



- Git



What is GitHub

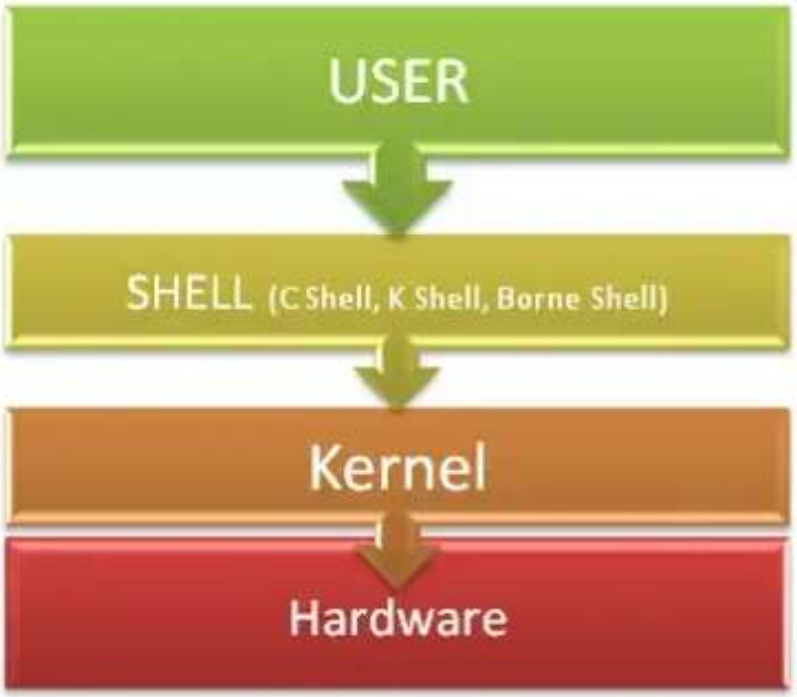
- GitHub.com is a website server that hosts git repositories
- Hosting repositories facilitates the sharing of codebases among teams by providing a GUI to easily clone repos to a local machine
- When you push your code repositories on GitHub, you will be creating your own developer's portfolio
- Lots of resources online
  - https://github.s3.amazonaws.com/media/progit.en.pdf

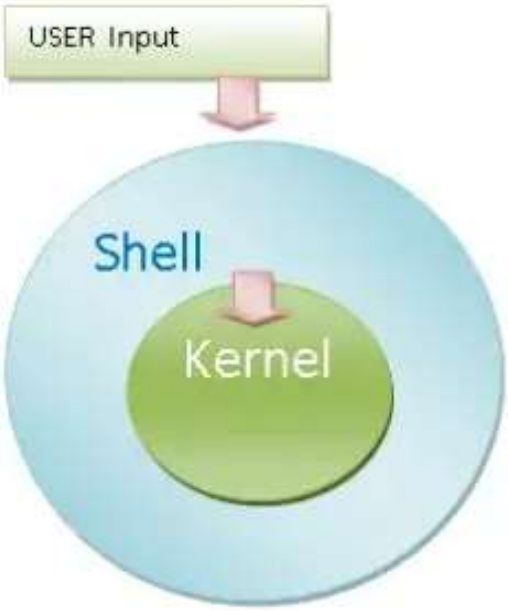
The shell (operating system)

- Operating systems provide a "command line" interface which allows the user to enter commands
  - These commands are translated by the shell into something the kernel can comprehend and execute
- Shell is not part of the operating system kernel
- It is a command line interpreter (CLI)
- In Unix, a user can pick their shell
  - Popular Unix (Linux) shell: sh, bash, ksh, csh, tcsh, …
  - Default shell on EECS Linux machines: bash

A simplified view of a shell

- A shell is the primary interface between a user sitting at the terminal and the operating system





https://medium.com/@clturner23

Default shell

- Linux default shell: most Linux systems default to the bash shell
  - I prefer csh (some similarity with the C language)
- Once you learn more about the shell options, you may want to change to another one
  - Command to change shell: `chsh`

A useful shell command

- Aliasing command
  - Assigns a command, possibly with many options and flags, to another name
  - Usually it is a shorter name or one that is easier to remember
- Setting up an alias
  - csh: alias alias-name original-command
  - bash: alias-name=original-command

  - Example in csh:

```
alias 348 ~/Teaching/2023/Spring/EECS348
```

Create an alias command every time?

- Each shell program normally comes with a configuration (login) file
- Bash configuration (login file): `.bashrc`
- What to include
  - User-defined aliases
  - User-defined environment variables
  - Can include conditional statements

Other useful shell commands

- Directory: `mkdir, rmdir, cd, ls, …`
- Files: `cp, cat, mv, rm, sort, wc, …`
- Search: `grep, find, …`
- Editor: `vi, vim, emacs, nano` (for the beginners)
- File/directory permission: `chmod, chown, …`
- Software development: `make, tar, git, vim, …`
- Many others

Software engineering uses

- Writing shell scripts
- Create a text file
- Include in the first line: `#!/bin/bash`
- Write scripts that do different tasks
  - A shell script: a text file that contains a sequence of commands
  - Command sequences in which a user has a need to use repeatedly in order to save time
  - Shell scripts contain ASCII text and are written using a text editor
  - Automating the code compiling process
  - Executing routine backups
  - Personal example: compiling LaTeX files

Software engineering uses

- What can be included in a shell script
  - Shell commands
  - Assignment statement
  - Loop statements (while, for)
  - If statement
  - System calls
  - …
- Some special symbols also have their own meanings
  - `#, %, $, |, [ ], …`

A simple shell script

- The following script has while statement
  - Iterate for 5 times
  - The value of count variable will increment by 1 in each step
  - When the value of count variable will be 5 the while loop will terminate

```
#!/bin/bash
valid=true
count=1
while [ $valid ]
do
echo $count
if [ $count -eq 5 ];
then
break
fi
((count++))
done
```

Summary

- It is a command line interpreter (CLI)
- Unix/Linux affectionate love the CLI
  - It is very powerful and provides a lot of control
  - It is simple (there is an initial learning curve)
  - Nevertheless, many GUI interfaces too (most Linux sys admins and power users do not use GUI)
- You most likely will learn a lot more in an OS course
- For now, you need to learn the purpose and the very basics of shell scripting

Software development IDEs

- Integrated Develop Environments (IDEs) are software packages that provide comprehensive support for coding, testing, and debugging
- The components of an IDE
  - Editor
  - Build support (link, compile)
  - Execute
  - Debug

Nice features of an IDE

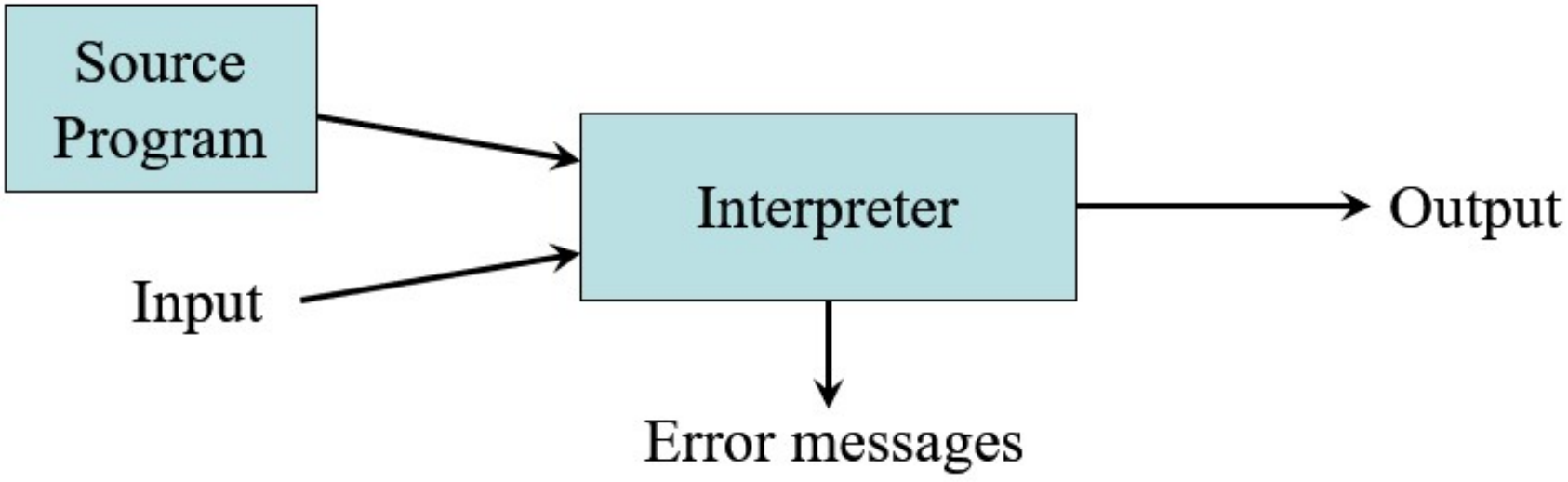
- Syntax highlighting and aid in editing (e.g., matching brackets)
- Packaging options (e.g., tar and zip archives)
- Posting to an online repository (e.g., GitHub)
- Configurable build support (e.g., multiple programming languages)
- Smart feedback
- Coding templates
- Documentation support/lookup: `scp( )`

Examples of popular IDEs

- Microsoft Visual Studio
- Eclipse
- AWS Cloud9
- Android Studio
- PyCharm
- Spyder
- More …

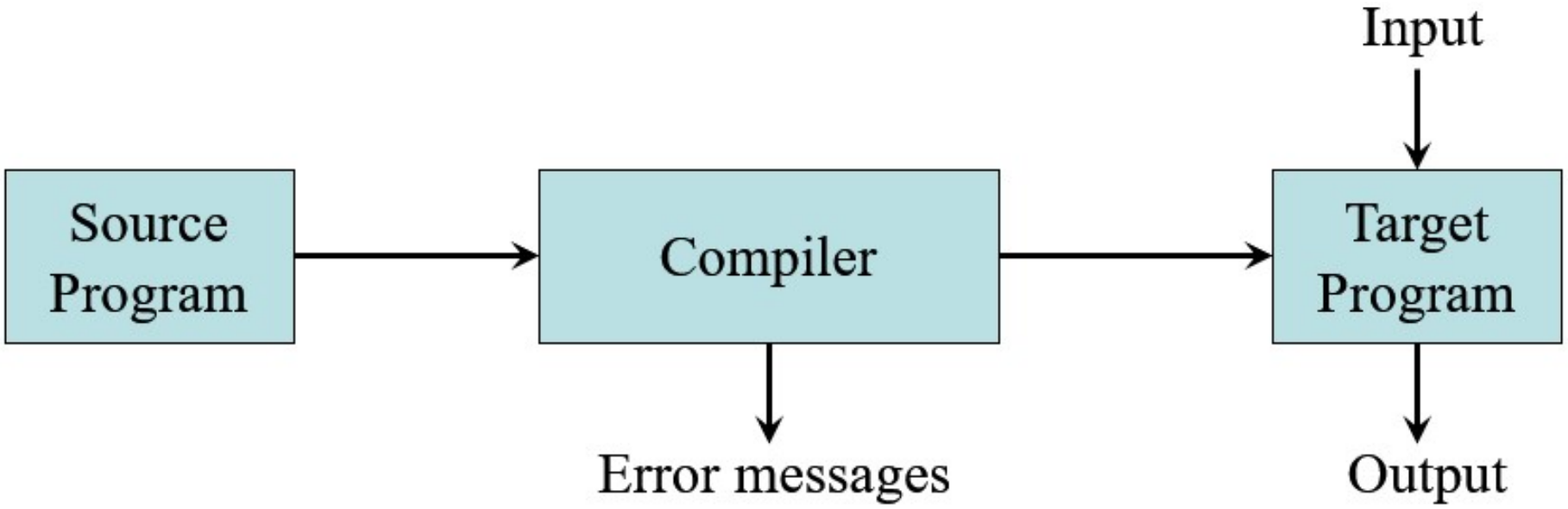
Compiled programs vs interpreters

- Interpretation
  - Performing the operations described by the source program
  - An extremely simplistic view



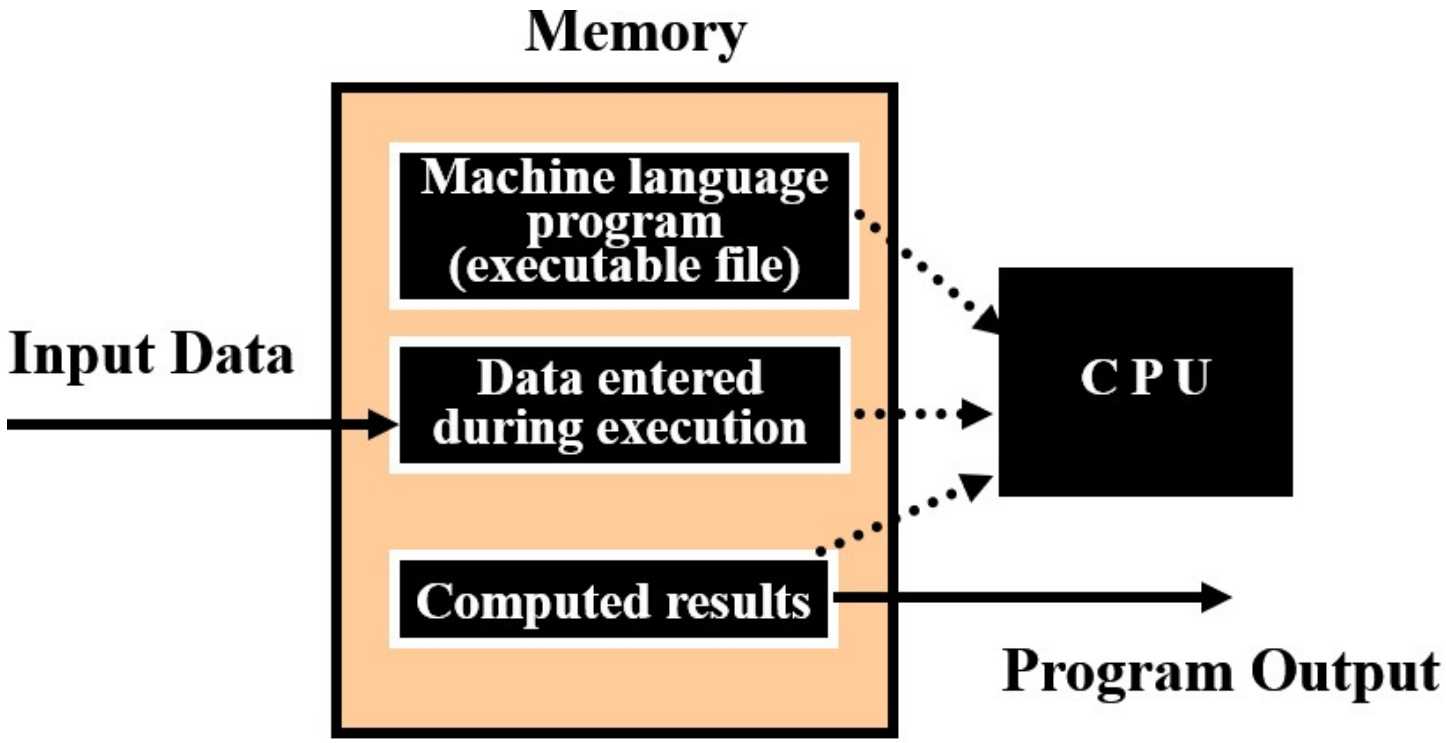
Compiled programs vs interpreters

- Compilation
  - Translation of a program written in a source language into a semantically equivalent program written in a target language
  - An extremely simplistic view

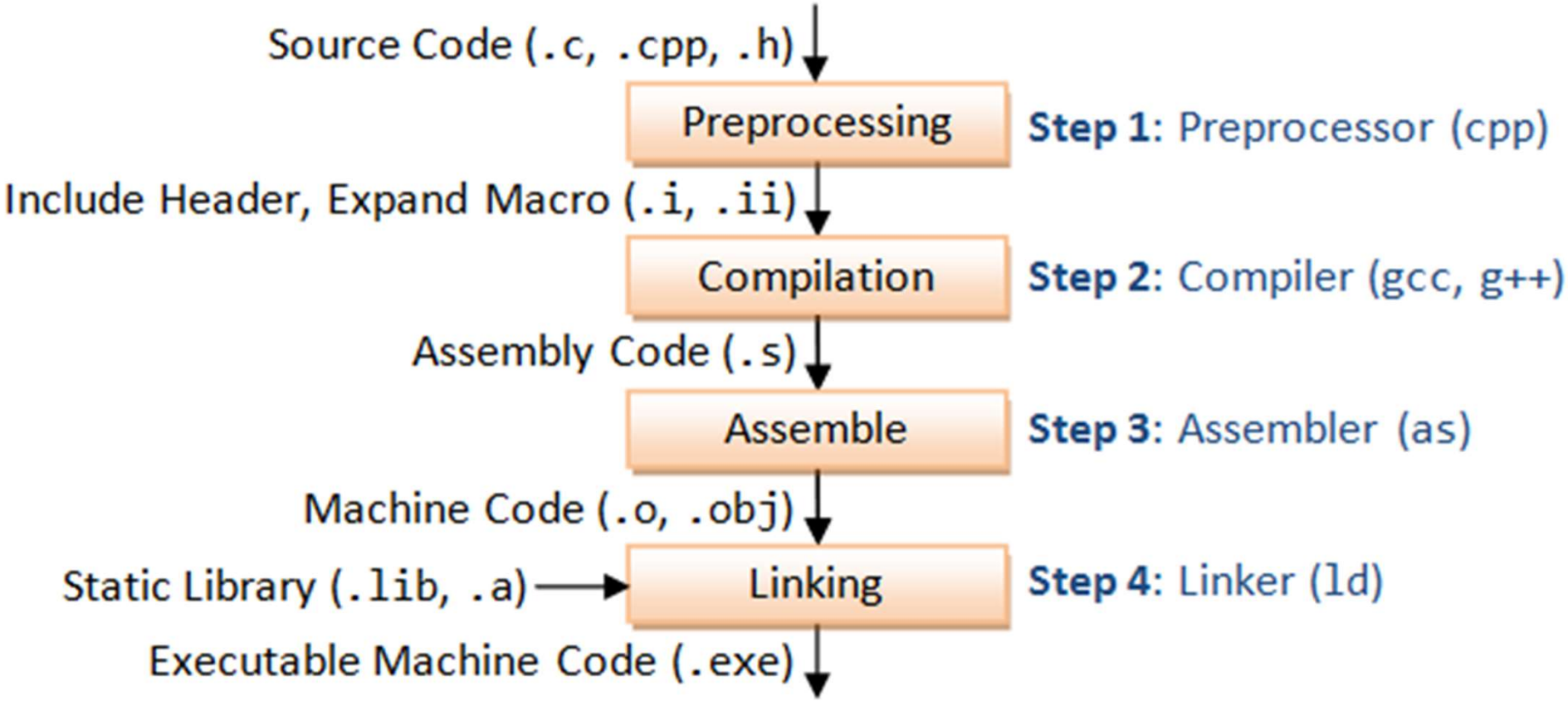


Compiled programs vs interpreters

- Compilation
  - Translation of a program written in a source language into a semantically equivalent program written in a target language
  - Another extremely simplistic view



Compilation: A simplified view



How to compile a program

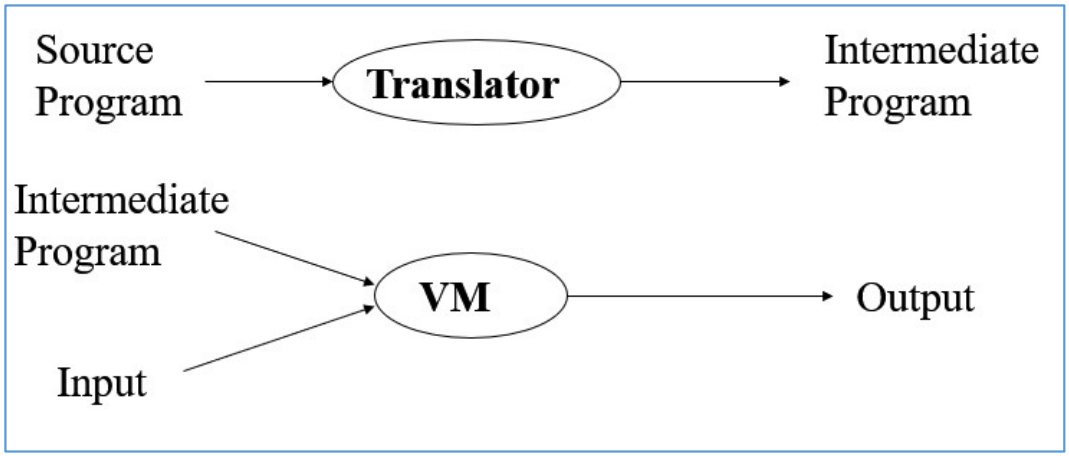
- Depends on the OS environment, platform, tools
- A very simplified approach on a Linux environment

```
$ gcc -c  myprogram.exe myprogram.c
```

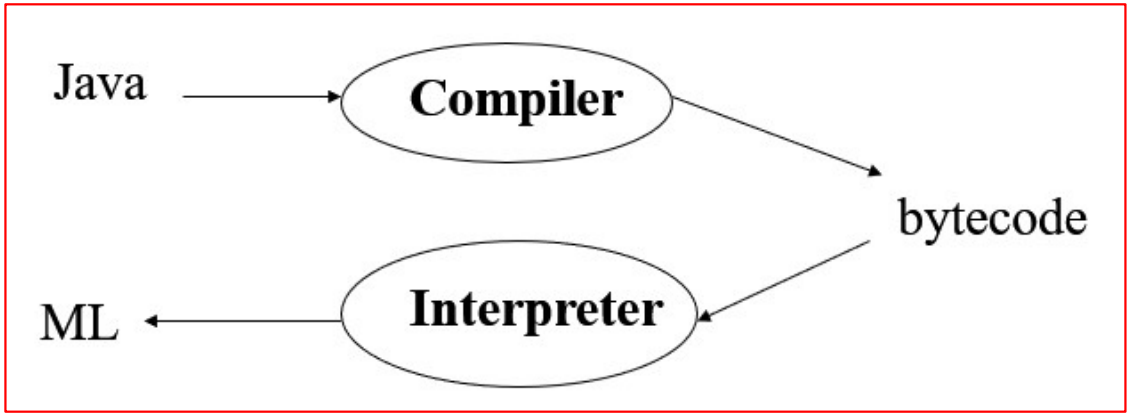
- `myprogram.exe` is now an executable program
  - Note: `.exe` extensions are not common in a Linux environment
- An excellent compilation manager on Linux: `make`
- Version management tools (e.g., `git`) and services (GitHub) are essential

Compilation and interpreters: a mixture

- Interpreter implements a Virtual Machine (VM)



- Java for portability



Summary

- An interpreter produces a result from a program
- A compiler produces a program in assembly language
  - The assembler of architecture then turns the resulting program into binary code
  - Assembly language varies for each individual computer, depending upon its architecture