BCA SEMESTER - II

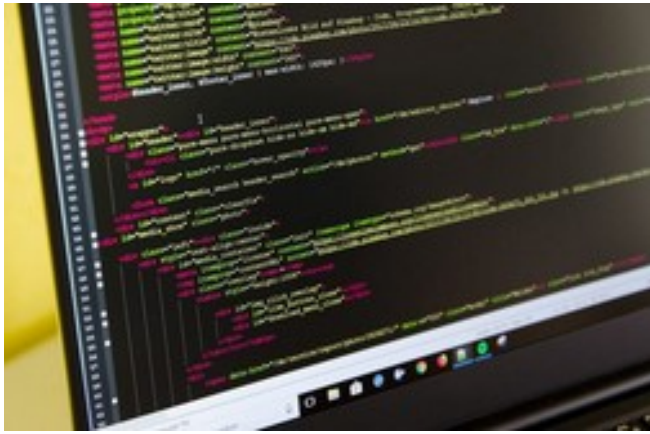
0302203
HISTORY OF COMPUTING

UNIT – 3
HISTORY OF PROGRAMMING LANGUAGES

- Dr. Disha Shah

Index

- **History of Programming Languages**
  - Ada Lovelace’s machine algorithm
  - Machine Language
  - Symbolic Programming Language
  - Lower Level Languages
  - Higher Level Languages
  - FORTRAN
  - ALGOL (Algorithmic Language)
  - LISP (List Processor)
  - COBOL (Common Business Oriented Language)



Index

- **Mid 1900s (Any 3)**
  - BASIC
  - PASCAL
  - Smalltalk,
  - C,
  - PROLOG
  - Ada
  - C++
  - Python
  - Ruby
  - Java,
  - PHP
  - Java Script
- **Mid 2000s (Any 3)**
  - Scala
  - Go
  - Dart
  - Swift
  - AlphaGo
  - Rust
  - Kotlin
  - Flutter
  - NLP

Go

- Go is a general-purpose language designed with systems programming in mind.
- It was initially developed at Google in the year 2007 by Robert Griesemer, Rob Pike, and Ken Thompson.
- It is strongly and statically typed, provides inbuilt support for garbage collection, and supports concurrent programming.
- Programs are constructed using packages, for efficient management of dependencies.
- Go programming implementations use a traditional compile and link model to generate executable binaries.
- The Go programming language was announced in November 2009 and is used in some of the Google's production systems.

Go

- Go is syntactically similar to C, but with memory safety, garbage collection, structural typing, and CSP-style concurrency.
- The language is often referred to as Golang because of its former domain name, golang.org , but the proper name is Go.

Features of Go Programming

- Support for environment adopting patterns similar to dynamic languages. For example, type inference (x := 0 is valid declaration of a variable x of type int)
- Compilation time is fast.
- Inbuilt concurrency support: lightweight processes (via go routines), channels, select statement.
- Go programs are simple, concise, and safe.
- Support for Interfaces and Type embedding.
- Production of statically linked native binaries without external dependencies.

Features Excluded Intentionally

- Support for type inheritance
- Support for method or operator overloading
- Support for circular dependencies among packages
- Support for pointer arithmetic
- Support for assertions
- Support for generic programming

Qwentic

Why Golang?

Concurrency:

Millions of platform users.

Go has many built-in features designed to handle several "concurrent web requests," due to which it is very efficient as opposed to legacy languages such as Python .

Scalability:

Grows with the business

As an enterprise grows the programs used will be required a do several things at the same time. Golang can easily scale due to its ability to handle several simultaneous tasks.

Error Checks:

Nil Malfunction

Go comes with a built-in error type. It uses error values to indicate an abnormal state.While writing the code, the developer can spot errors leading to nil malfunction.

Compiled Language:

Fast Performance

Go comes with a built-in error type. It uses error values to indicate an abnormal state.While writing the code, the developer can spot errors leading to nil malfunction.

Garbage Collection:

Boost App Speed

Golangs collection pauses are as low as 500 microseconds. As a result, it is predictable, better performance and fast loading time.

Cross Platform:

Low investment

Golang performs well across various platforms such as Windows, Linux, Unix, Android/ iOS and other operating systems as well as Cloud applications. This means Businesses don't have to spend much on ensuring Cross-Platform functionality.

What is the Golang used for?

- Go was originally built for programs related to networking and infrastructure.
- It was intended to replace popular high-performance server-side languages like Java and C++.
- Today, Go is used for a variety of applications:
  - Go is popular for cloud-based or server-side applications.
  - DevOps and site reliability automation are also popular ways to use Go.
  - Many command-line tools are written in Go.
  - Go is used in the world of artificial intelligence and data science.
  - Some use Go from micro-controller programming, robotics, and games.

Why Use Go?

- Go is fun and easy to learn
- Go has fast run time and compilation time
- Go supports concurrency
- Go has memory management
- Go works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.)

Go Compared to Python and C++

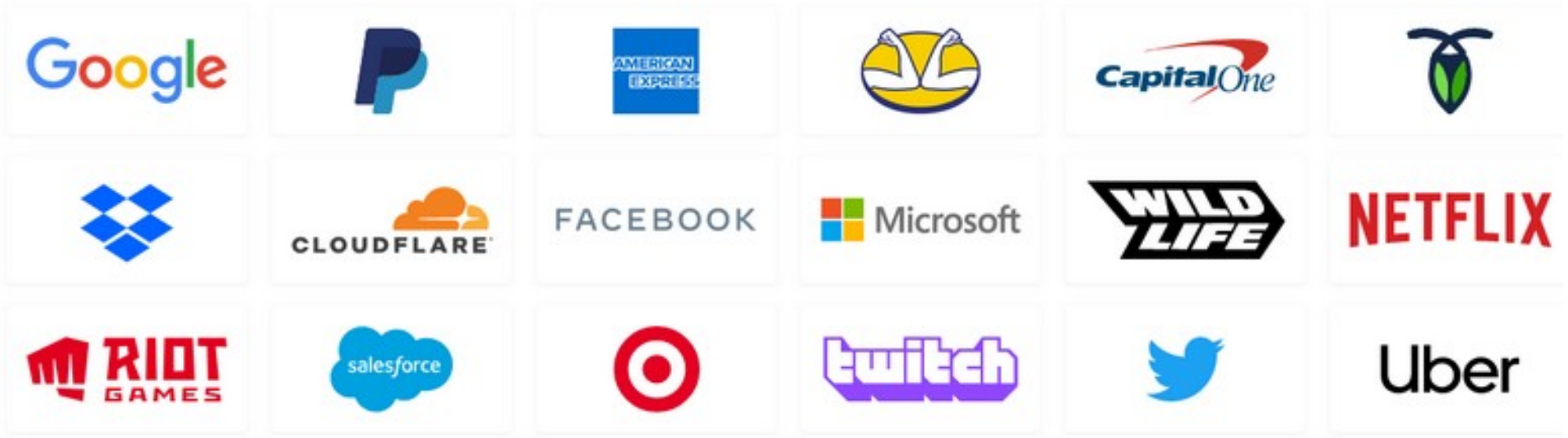
Go	Python	C++
Statically typed	Dynamically typed	Statically typed
Fast run time	Slow run time	Fast run time
Compiled	Interpreted	Compiled
Fast compile time	Interpreted	Slow compile time
Supports concurrency through goroutines and channel	No built-in concurrency mechanism	Supports concurrency through threads
Has automatic garbage collection	Has automatic garbage collection	Does not have automatic garbage collection
Does not support classes and objects	Has classes and objects	Has classes and objects
Does not support inheritance	Supports inheritance	Supports inheritance

- **To start using Go, you need two things:**
  - A text editor, like VS Code, to write Go code
  - A compiler, like GCC, to translate the Go code into a language that the computer will understand

- **Go Install**
  - You can find the relevant installation files at https://golang.org/dl/.

- **Text Editor**
  - You will require a text editor to type your programs. Examples of text editors include Windows Notepad, OS Edit command, Brief, Epsilon, EMACS, and vim or vi.
  - The name and version of text editors can vary on different operating systems. For example, Notepad is used on Windows, and vim or vi is used on Windows as well as Linux or UNIX.
  - The files you create with the text editor are called source files. They contain program source code. The source files for Go programs are typically named with the extension ".go".

Companies using Go



https://go.dev/solutions/#case-studies

Go screen

hello.go

```go
package main
import "fmt"

func main() {
    fmt.Printf("hello, world\n")
}
```

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.17763.615]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Anshul Aggarwal>go run first.go
!... Hello World ...!

C:\Users\Anshul Aggarwal>
```