2 Weeks
- OOP Concepts
- Class, Object

Language
- Syntax
- Rules

OOSD - Object Oriented Software Development
- OOA (Object Oriented Analysis)
- OOD (Object Oriented Design)
- OOP (Object Oriented Programming)
  - C++

8:00-9:50
10:10 - 12:00
12:10 - 1:30

Attendance System
- Employee
- Student
- Attendance{
Date
Time
Emp/Stu
}

Employee{
id
name
salary
mobile
}

student{
rollno
name
marks
mobile
}

POP (Procedure Oriented Programming)
- All the logic is kept in procedures/functions

OOP
- Major Pillars (Compulsary)
  1. Abstraction
  2. Encapsulation
  3. Modularity
  4. Hirerachy
     - Association(has-a)
     - Inheritance (is-a)
- Minor Pillars
  1. Typing/Polymorphism
  2. Concurrency
  3. Persistance

Abstraction

printf("");

Encapsulation
- Binding of data and code together

```
void mypintf(){

}
```

Polymorphism

printf("Heelo World");

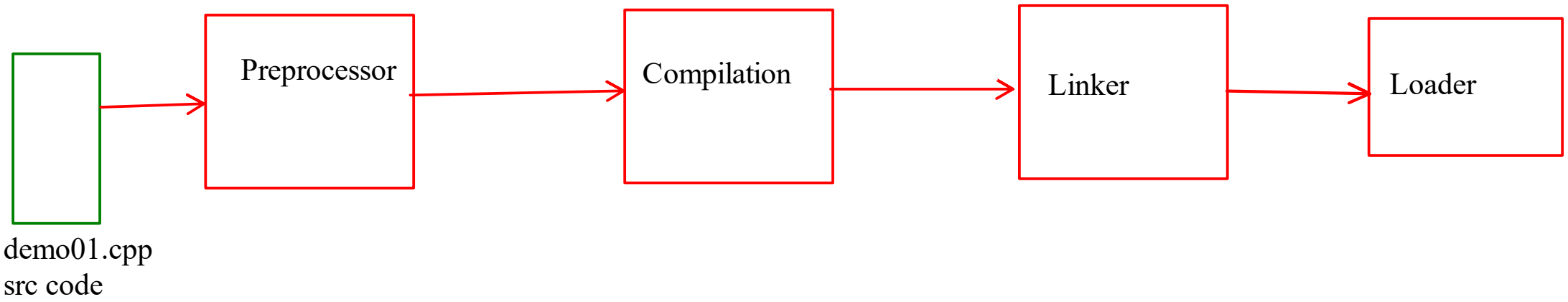
int num1=10;
printf("value%d",num1);

C with classes
- Bjarne Stroustrup
- class

C++ CPP
C++ 98

gcc/g++

IDE
Vs code
Editor



Datatype
- 1. Nature
  - What type of data can be stored
- 2. Memory
  - How much memory is required to store the data
- 3. Operations
  - Wthat type of operations can be performed on that data

int
- Number
- 4 bytes
- Arithmetic Operations

1. Fundamental DataTypes
   - void
   - int
   - float
   - double
   - char
   - bool (1 byte)
   - wchar_t (2 bytes)

2. Derived Datatypes
   - array
   - pointer
   - structure
   - class
   - function

Type Qualifiers

Type Modifiers
short, long, signed, unsigned

bool status = true
bool status = false

wchar_t

unicode
1. UTF-8
2. UTF-16
3. UTF-32

a=21/65/97 A=100

string name

```
struct Time{
int hrs;
int mins;
}

int main(){
int hrs = 2;
int mins=30;

int hrs2=
int mins2=
```

```
int main(){
struct Time start_time;
struct Time end_time;

}
```

```
}
```