



# SYLLABUS



## C programming

- ✗ Variables
- ✗ Operators
- ✗ Conditionals and loops
- ✗ Functions
- ✗ Recursion
- ✗ Pointers and arrays
- ✗ Structure and union

## Data Structures

- ✗ Stacks
- ✗ Queues
- ✗ Linked Lists
- ✗ Trees
- ✗ Binary search trees
- ✗ Binary heaps
- ✗ Graphs



# FEATURES OF C



Procedural programming



Middle level language

System level features



Direct access to memory through pointers



Bit manipulation using bitwise operators



Writing assembly code within C code



Popular choice for system level apps

**Portability:** ability to run your code anywhere you want.

Assembly:

~~Portable~~

C:

Portable

# WHY C?



Portability



Less lines of code

## IN ASSEMBLY

```
Mov bl, 2  
Mov cl, 3  
ADD bl, cl  
Mov Result, bl
```

1001110010001000111....

## IN C

I Told you  
that I am  
better...

Result = 2 + 3

Compiler



# HIGH LEVEL VS LOW LEVEL



Less efforts

COBOL, FORTRAN,  
C++, LISP, Pascal,  
Prolog etc...



More efforts

Assembly language

Degree of Abstraction



# FEATURES OF C



Wide variety of built in functions, standard libraries and header files.

stdio.h

math.h

string.h

stdio.h

scanf

isdigit

sqrt

conio.h

stdlib.h

malloc

time.h

printf

ctype.h

strcat



```
# INCLUDE <STDIO.H>
```

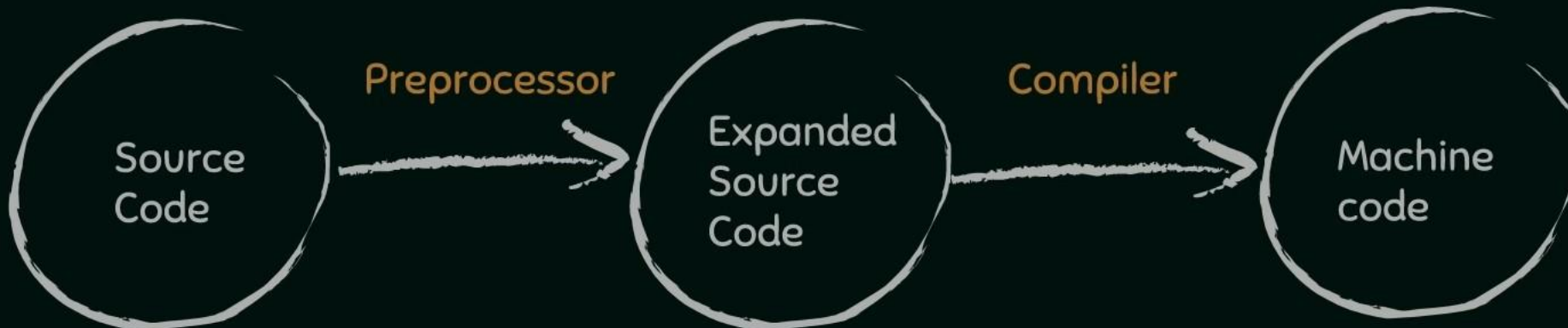
**Preprocessor:** replaces text (starting with #) with the actual content.



Replaces before the compilation begins




Output of preprocessing is expanded source code





## Stdio.h: standard input output file

- 
- Header file
  - Contains declarations (prototypes) of functions like printf, scanf etc.