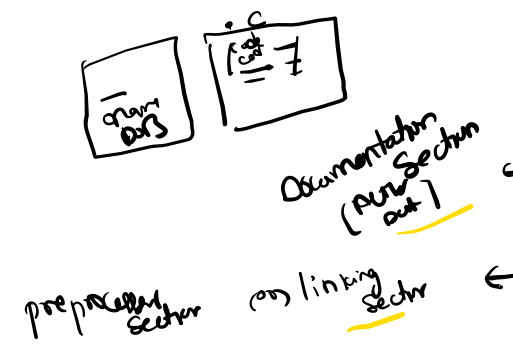


Basic Structure of c program

Save → compile → Run
 locale → Thxpc → TurboC3 → Bin →

```

/* Documentation Section */
/* linking Section / preprocessor Section */
/* Definition Section */
/* Global declaration Section */
int main()
{
    // body logic
    return 0;
}
/* Sub program Section */
  
```



online C Compiler

```

Online C Compiler.
Code, Compile, Run and Debug C program online.
Write your code in this editor and press "Run" button to compile and execute it.

#include <stdio.h>

int main()
{
    printf("Hello World");
    return 0;
}
  
```

Documentation

Linking
Definition

Global
var

body
logic

Function
Subprogram
Section

```

/* Program : Simple C Program Example
 * Author : Your Name
 * Date : 29-09-2025
 * Purpose : Demonstrates all sections of C program
 */

#include <stdio.h>

int globalVar = 100;
void greet();

int main() {
    int localVar = 50;

    printf("Hello, World!\n");
    printf("Global Variable: %d\n", globalVar);
    printf("Local Variable: %d\n", localVar);

    greet();

    return 0;
}

void greet() {
    printf("Welcome to C Programming!\n");
}
  
```

Exam

Before Compiling

→ preprocessor

Linker

stdio.h
↳ [standard input and output]

C library

compiler

{ 0, 1 }

stdio.h
- printf (args)
- return

```
Online C Compiler.  
Code, Compile, Run and Debug C program online.  
Write your code in this editor and press "Run" button to compile and execute it.  
#include <stdio.h>  
int main()  
{  
    printf("Hello World");  
    return 0;  
}
```

printf
args
return

C program # → stdio.h
replace
Linker
C.lib → actual replace.

```
extern int printf(const char *format, ...);  
extern int scanf(const char *format, ...);  
  
int main() {  
    printf("Hello, World Hai !\n");  
    return 0;  
}
```

#include <stdio.h>
int main()
{
 //
 }
}

```
extern int printf(const char *format, ...);  
extern int scanf(const char *format, ...);  
  
int main() {  
    int printf(const char *format, ...) {  
        va_list args;  
        int result;  
        va_start(args, format);  
        result = vprintf(stdout, format, args);  
        va_end(args);  
        return result;  
    }  
    return 0;  
}
```

stdio.h (vs)

Declaration

↳ printf()

↳ scanf()

C-lib

printf()

scanf()

C-program

↳ #

stdio.h

code replace

Linker

Clib

actual
code

replace

#include <stdio.h>



C lib
stand
code

replace

#include

-stdio.h



current directory
etc.



C lib stand (actual code)