(Video 20) C Programming (Important Question Set 3)

Q1.

```
#include <stdio.h>
int main(){
    int var = 0x43FF;
    printf("%x", var);
    return 0;
}
```

43ff

0x → HexaDecimal

%x → print H.D.

→ 0x or 0X both same

%X → 43FF

%x → 43ff

%d হলে Hexa to Decimal

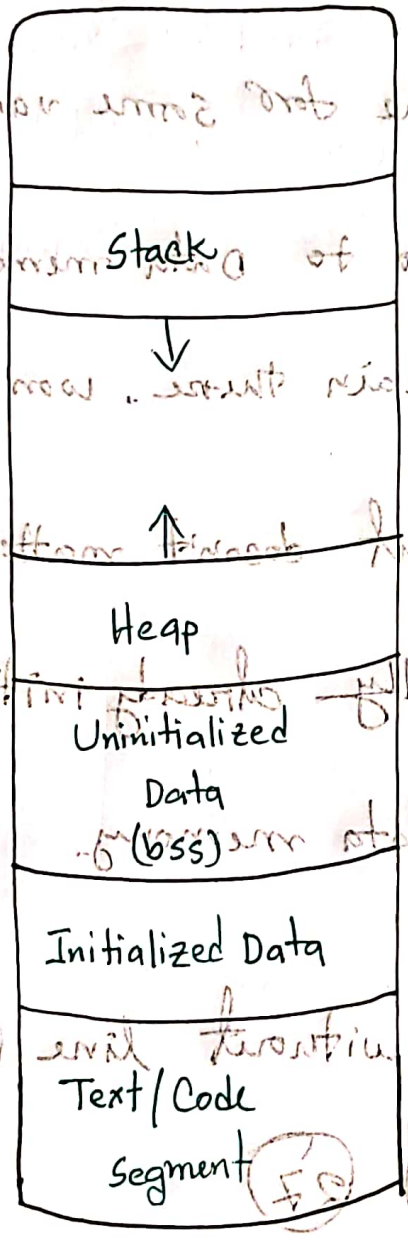
43FF → 17407

92.

```
#include <stdio.h> ①
static int i; ②
static int i = 27; ③
static int i; ④
int main(){ ⑤
    static int i; ⑥
    printf("%d", i); ⑦
    return 0; ⑧
} ⑨
```

Frist look at the memory layout

Memory Layout of C program



} Command line Arguments and environment variables.

two memory segment:

1) Text/code segment
2) Data segment
   a) Initialized
      i) Read Only
      ii) Read Write
   b) Uninitialized
      (bss - Block Started by Symbol)

Contain machine code of compiled program.

c) stack

d) Heap

② → Global variable declared but not initialized

so it will Go to bss memory.

③ → initialization done for some variable,

so Now it moves to Data memory

and will remain there. won't change.

④ Again declared but doesn't matter cos

the variable ~~already~~ already initialized and

won't move from data memory.

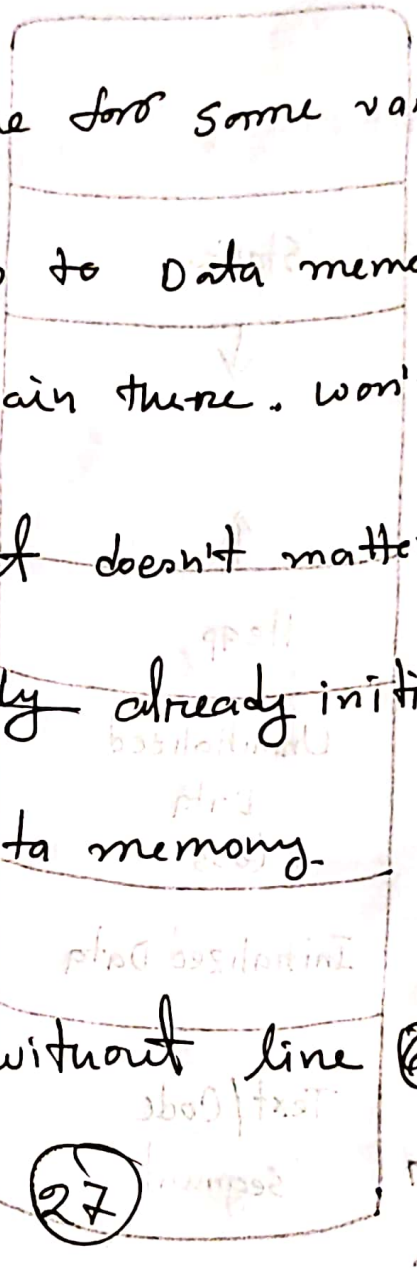
→ So, if we print without line ⑥ it will

print output 27 27

But if we have line ⑥ then it is in a

loop where the loop variable will get

more value/importance than Global variable



as the global variable is ~~declearad~~ initialized = 27

but in loop it is decleand without initializing

So, "i" will contain a value of 0.

and so, if we print with line (6)

the output will be "0".

```
#include<stdio.h>
~~int~~ static int i;
static int i = 27;
Static int i;
int main() {
    printf("%d", i);
    return 0;
}
```

[loop এর ভেতরে variable always বাইরে priority পাবে।]

27

[একবার initialized হয়ে গেলে আর কখনো হবে না।
i = 0 ও কখনো না।]