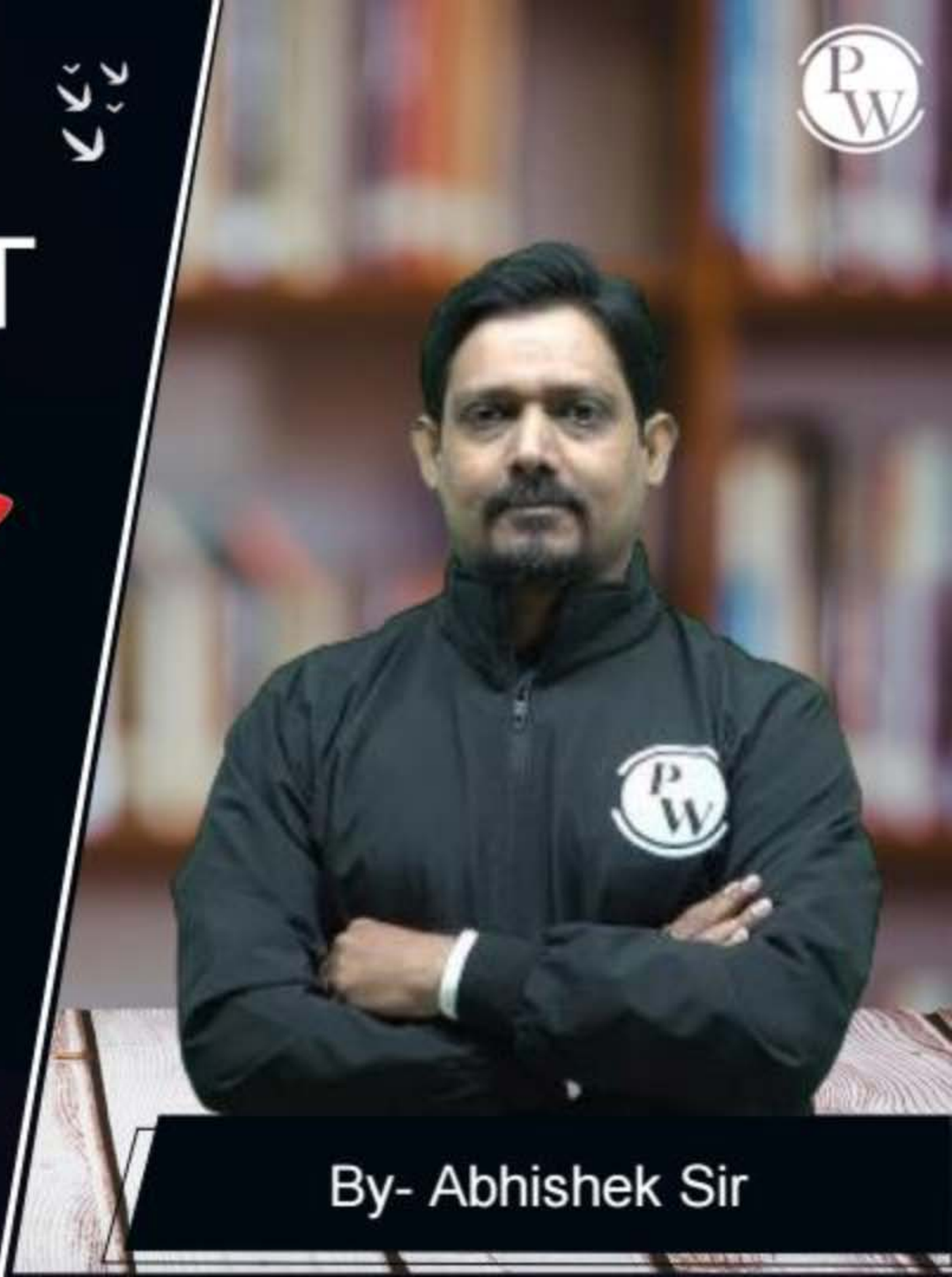
Computer Science & IT

C programming

Control Flow Statement

Lecture No. 05



By- Abhishek Sir

Recap of Previous Lecture

- Topic — for loop
- Topic — while loop
- Topic — practice problem
- Topic
- Topic

Slide

Topics to be Covered

- Topic — do while
- Topic — break
- Topic — Continue
- Topic — practice
- Topic — function

Slide

Home Work

```
int i;                                  ← ⌊log2 n⌋
for (i=2; i<=n; i=i*2)
    stmt;
```

How many times loop will Run $2^k = n$

$\Rightarrow$ $\log_2$ both side

$\log_2 2^k = \log_2 n$

$k \log_2 2 = \log_2 n$

$k = \log_2 n$

Iteration No	
1	$2^1 <= n$
2	$2^2 <= n$
3	$2^3 <= n$
4	$2^4 <= n$
⋮	
k	$2^k <= n$ ← $2^k = n$

Slide

Home Work

```
for (i=2; i<=8; i=i*2)
    stmt;
```

2 <= 8 ①

4 <= 8 ②

8 <= 8 ③

```
for (i=2; i<=11; i=i*2)
    stmt
```

2 <= 11 ①

4 <= 11 ②

8 <= 11 ③

16 <= 11 - ✗

Slide

Home Work

```
for (i=2; i<=8; i=i*2)
    stmt;
```

2 <= 8 ①

4 <= 8 ②

8 <= 8 ③

$\lfloor \log_2 n \rfloor$

$\lfloor 2.5 \rfloor = 2$

```
for (i=2; i<=17; i=i*2)
    stmt
```

2 <= 17 ①

4 <= 17 ②

8 <= 17 ③

16 <= 17 ④

Slide

Home Work

```
for (i=1; i<=8; i=i*2)
    stmt;
```

2^0
2^1
2^2
$\vdots$
2^k

2^1
2^2
2^3
$\vdots$
2^k

$k+1$

$\lfloor \log_2 n \rfloor + 1$

$3 + 1 = 4$

Slide

GATE 2014

t.me/Abhishekshaarmapw

#Q. Consider the ~~function func shown below~~:

```
#include<stdio.h>
int main() {
    int num = 65, count = 0;
    while (num) {
        count++;
        num>>= 1;
    }
    return (count);
}
```

output

printf("%d", count)

1 Right shift Divide by 2

num = num >> 1

	counter	num >>= 1	
65T	1	65/2 = 32	
T	2	32/2 = 16	
	3	16/2 = 8	
T	4	8/2 = 4	Nonzero
	5	4/2 = 2	Nonzero
	#6	2/2 = 1	Nonzero
(1)	7✓	1/2 = 0	Zero

The value returned by `func(65)` is _____.

Slide

do-while loop

* do-while ✓
* break ✓
* continue ✓

```
# include<stdio.h>
 int main(){
    int num=65, count=0;
    do {
      count++;
      num>>=1
    }while(num)
  printf("%d", count);
}
```

do while
condition at time
of exit.

* After executing the block one's, condition is tested.
* at least one time block will execute.

Slide

do-while loop

* do-while ✓
* break ✓
* continue ✓

if num=0 taken
count will be printed 1

```
# include<stdio.h>
int main(){
   int num=65, count=0;
   do {
      count++;
      num>>=1
   }while(num)
printf("%d", count);
}
```

count	num	condition
1	65/2=32	T
2	32/2=16	T
3	16/2=8	T
4	8/2=4	T
5	4/2=2	T
6	2/2=1	T
7	1/2=0	F

7

Slide

break

* break is loop control keyword
* break is used within block of loop & switch.
* it takes the control outside of switch & loop block where it is used.

Slide

break

Example :

i=1 - 10 time

i=2 - 10 times

20 times

```
#include<stdio.h>
int main(){
int i, j;
for(i=1; i<=10; i++){
    for(j=1; j<=10; j++){
        printf("Life is friends");
    }
if(i==2) break;
}
return 0;
}
```

10 times
2 times break

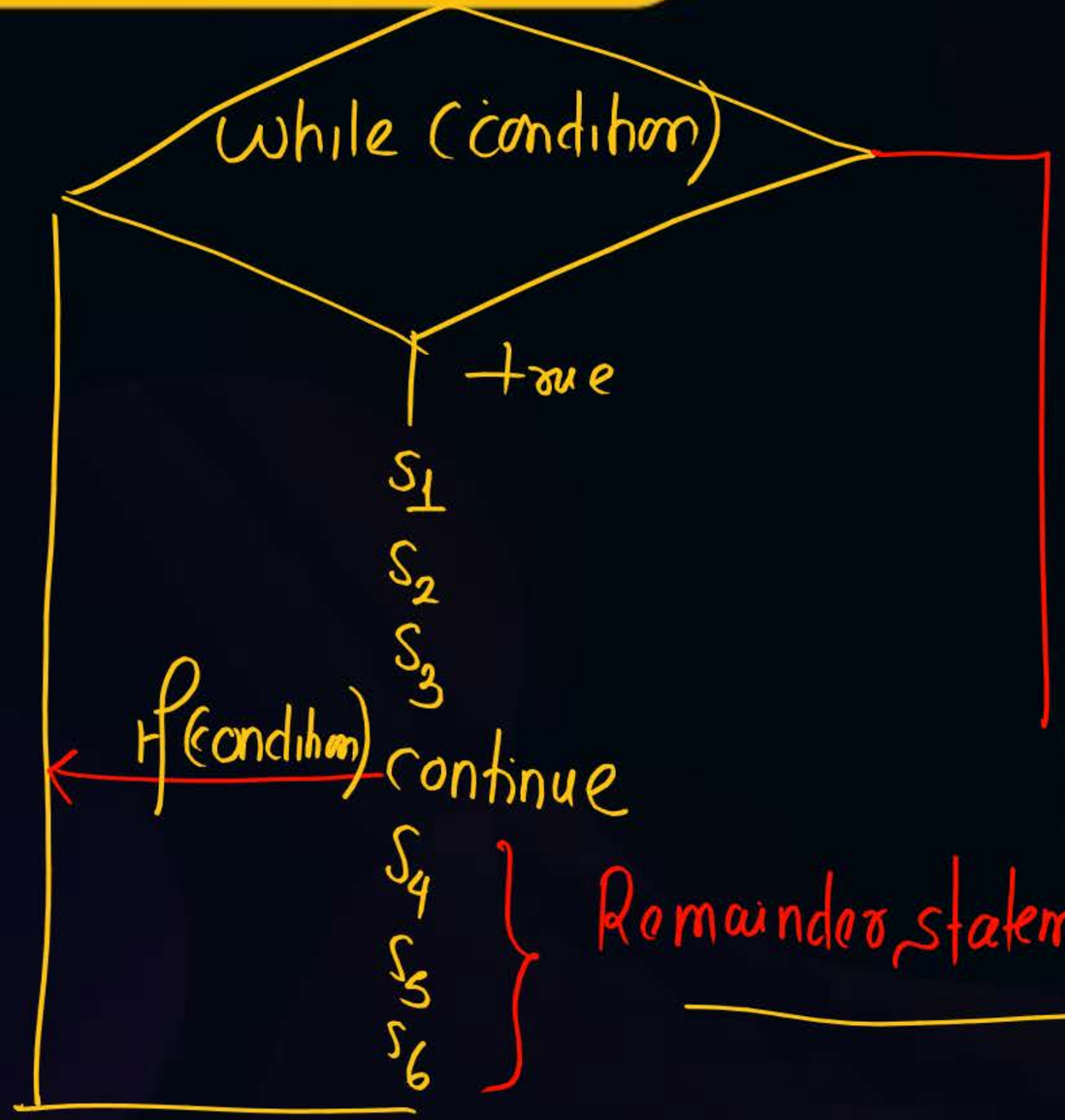
Slide

Continue

* Continue is also loop control statement.
* if continue keyword occurs then Remaining statement of loop block is Not executed and condition is retested.

Slide

Continue



false

Re test the condition

Remainder statement will Not execute

Slide

Question: Continue

poll N4T

#Q.
```
#include <stdio.h>
int main() {
    int i,j;
    int count=0;
    for(i =1;i<=3;i++){
        for(j=1;j<=20;j++){
            printf("I am a good student");
            if(i==2) break;
        }
    }
    return 0 ;
}
```

No. of time printf function will I am a good student;

i = 1 = 20 times

i = 2 = 1 time

i = 3 = 20 times

41

41

Slide

Question: Continue

#Q.
```
#include<stdio.h>
void main () {
for (int i = 1; i <= 10; i++) {
if (i == 5||i==6) {
        continue;
}
        printf("%d\n", i);
}
}
```

A 4
b.5
C.8
d. 11

Question

output

1 2 3 4 7 8 9 10

no. of values printed

8 value.

7

Slide

```
int main() {
    int i;
    for (i=1; i<=10, i++) {
        if (i==5 || i==6) {
            continue
        }
    }
    printf("%d", i);
}
```

Retest

Increement

Question

#Q.
```
#include <stdio.h>
int main() {
        int i=2+4%6*2+9/10;
        while (i<20){
            printf("I am good student");
            i++;
            if(i%2) continue;
        }
        return 0 ;
        }
```

The number of times printf statement executed is

A 4

b.5

C.10

d. 11

For loop

· HW problem

```
#include<stdio.h>
int main(){
int i,j=1,sum=0;
      for (i=1; i<=10;i++){
            sum= sum+j*j;
            j=j+1;
      }
      printf("%d",sum);
      return 0 ;
}
```

(A) 385
(B) 55
(C) 0100
(D) 10000

Slide

Function

function + storage class + Recursion > 1 modular programming
2. Reusability

function collection of Instruction considered as one logical unit

Slide

Function

prototype of function

return type function name (parameters) ; declaration

int fun (int, int);

return

char

float

void ← - No return.

Slide

Function

fun is a function taking two integer as arguments and Return an integer

Slide

Function

```
# include<stdio.h>
int fun (int, int);
int main(){
   int i=10, j=20, k;
   k= fun(i, j);
   printf("%d", k);
   return 0;
}
```

declaration

```
int fun ( int x, int y){
   return x+y;
}
```

x=10, y=20

10+20

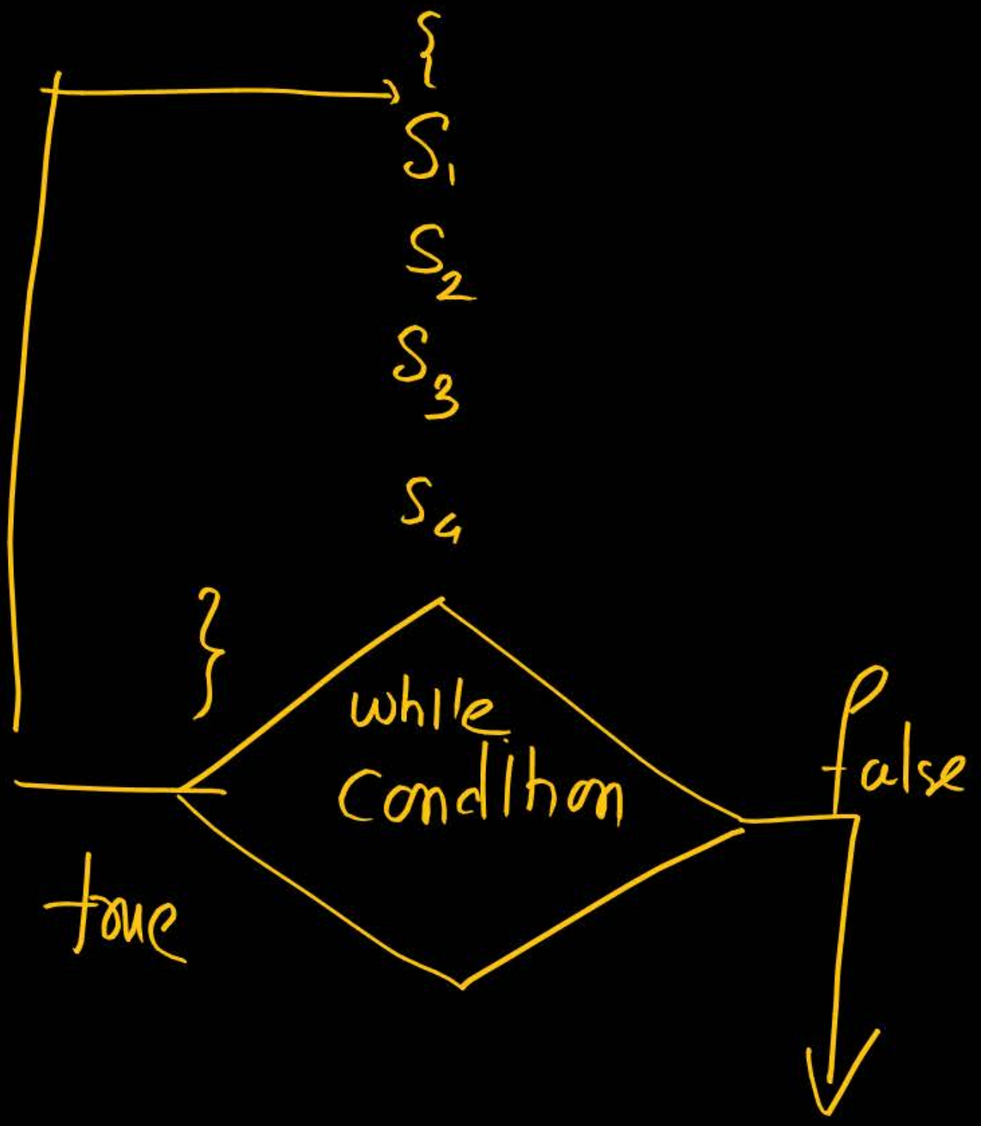
Definition

Control transfer

Control Return

30

Slide


{
S1
S2
S3
S4
}
while
condition
false
true

2 mins Summary

Topic do while

Topic break;

Topic continue

Topic function

Topic

t.me/Abhishekshaarmapw

Slide

THANK - YOU