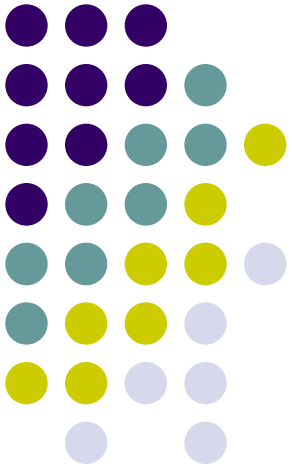
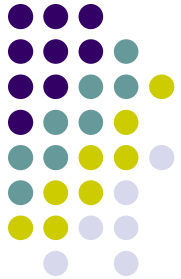
Paper 102: Programming & Problem solving through C

Lecture-05:Unit-I
C Fundamentals



Loop Constructs:while

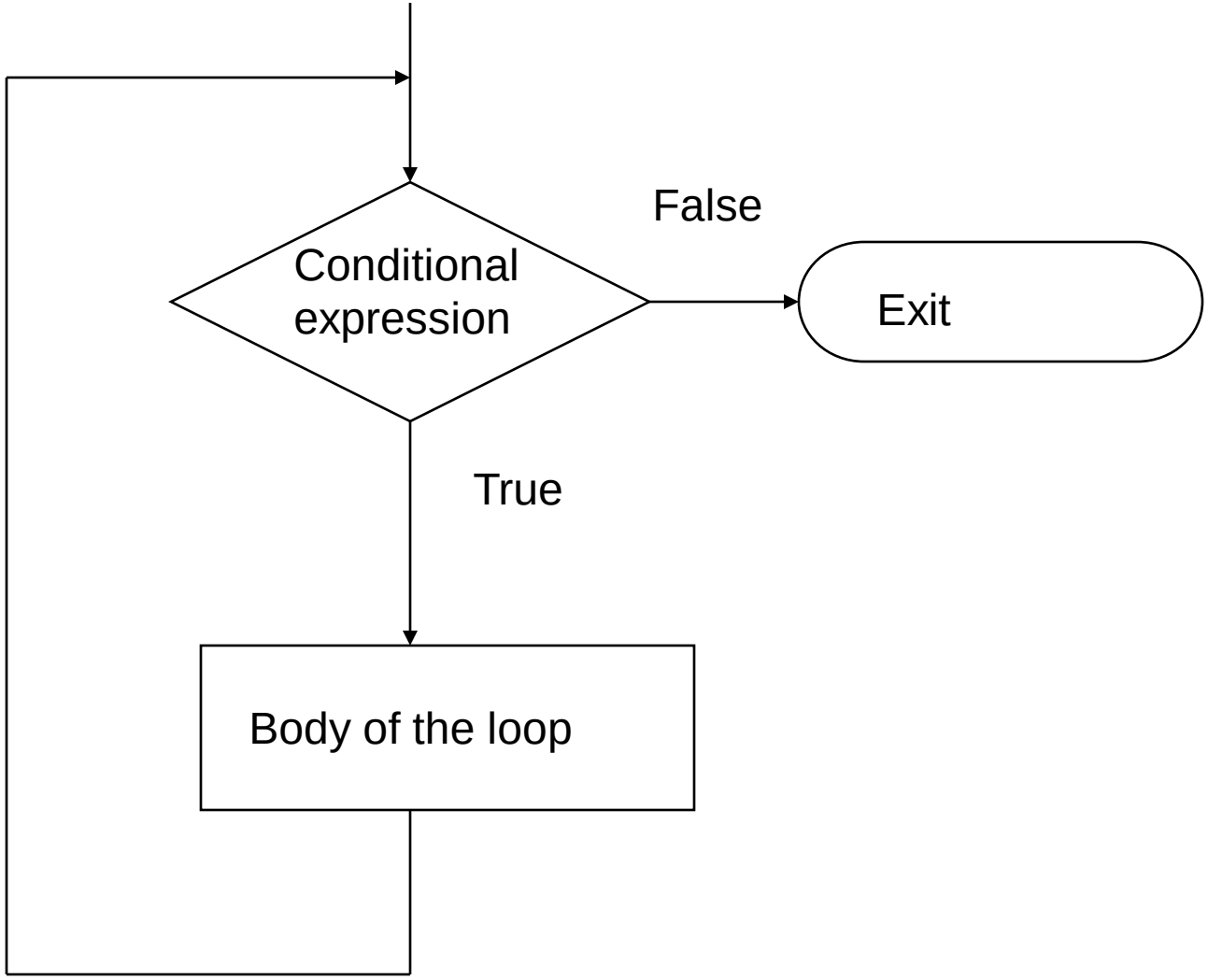


The while loop has the general form:

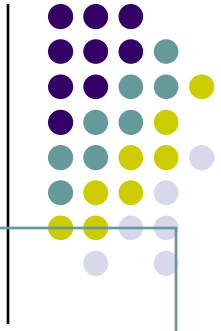
```
while(condition expression)
{
    statement block
}
```

- If a single statement is the object of the **while**, the braces may be omitted.
- The loop will repeat as long as the condition is true.
- The while tests its condition at the top of the loops.
  - Therefore, if the condition is false to begin with, the loop will not execute at all.
- The condition may be any expression.

Flowchart of the while loop



while loop example



```
#include<stdio.h>  /* to sum N numbers */
void main()
{   int num,n,i=1,sum=0;
    printf("Enter how many numbers to sum:");
    scanf("%d",&n);
    while(i<=n)
    {
        printf("\n Enter the number:");
        scanf("%d",&num);
        sum=sum+num;
        i=i+1;
    }
    printf("\nThe sum of all the %d numbers is=%d",n,sum);
}
```

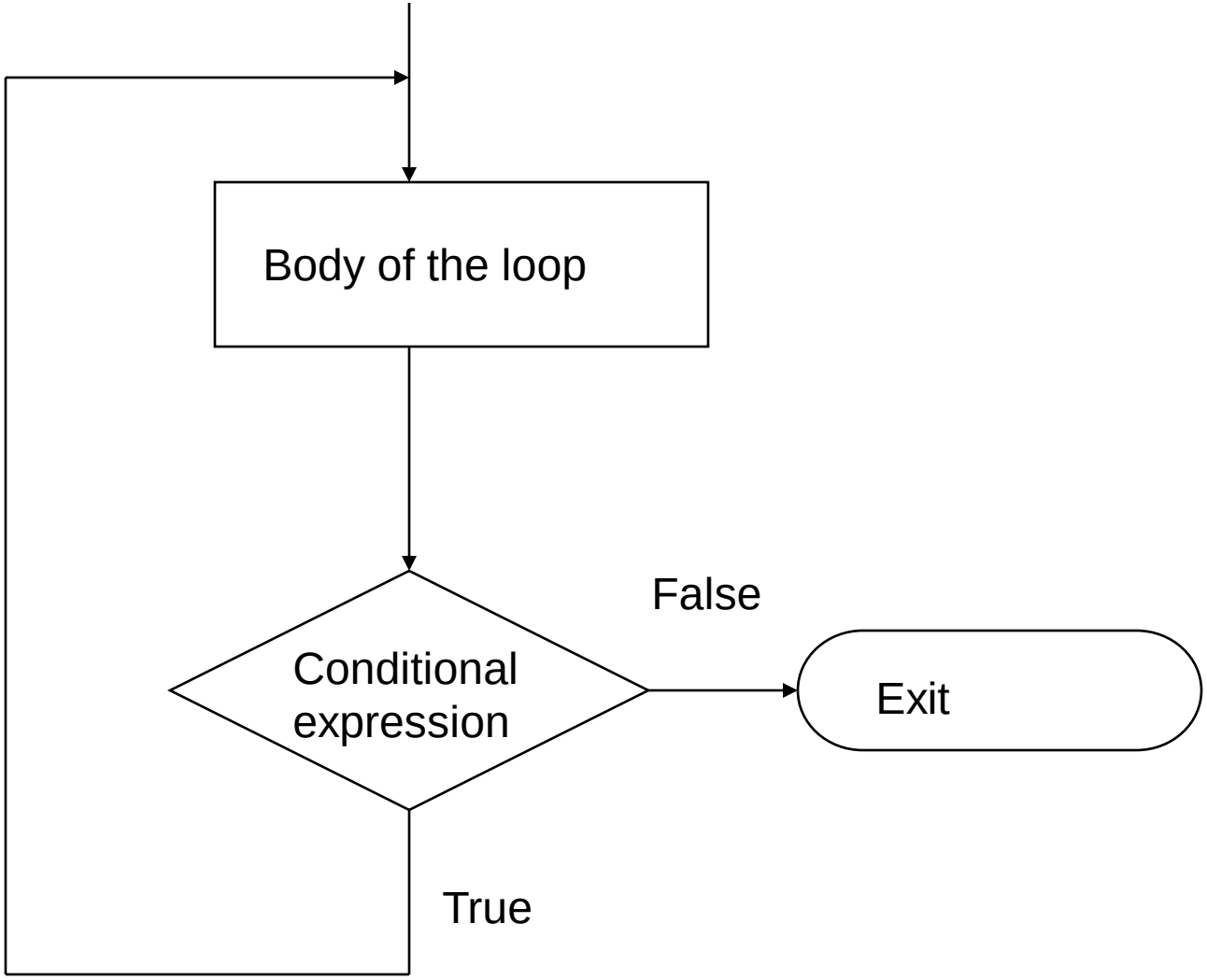
Loop Constructs: do_while loop

- The do loop is one of three loop constructs available in C. It test the condition at the bottom of the loop.
- The general form of the do loop is

```
do
{
    statement block
} while (condition expression);
```

- If only one statement is repeated, the braces are not necessary, but they add clarity to the statement.
- The do loop repeats as long as the condition is true.
- The do loop is the only loop in C that will always have at least one iteration
  - the condition is tested at the bottom of the loop.

Flowchart of the do-while loop



do-while loop example

```
#include<stdio.h>  /* to sum N numbers */
void main()
{    int num,n,i=1,sum=0;
     printf("Enter how many numbers to sum:");
     scanf("%d",&n);
     do    {
           printf("\n Enter the number:");
           scanf("%d",&num);
           sum=sum+num;
           i++;
           } while(i<=n);
     printf("\n The sum of all the %d numbers is=%d",n,sum);
}
```

Class assignment

1. Write a program to input a number and count the digits of the number.
2. Write a program that determine if a number entered by the user at run time is an Armstrong number or not.
3. Write a program to input a number and determine how many digits are Odd numbers.