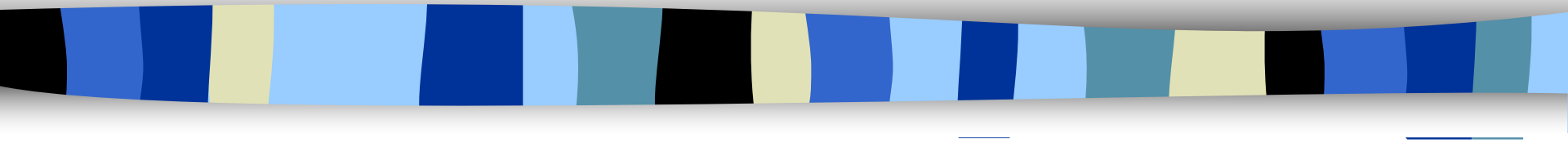
# *If-else*



# *Control/ Control flow statements*

- In C, the semicolon is a statement terminator.
- Braces { and } are used to group declarations and statements together into a *compound statement,* or *block,* so that they are syntactically equivalent to a single statement.
- The braces that surround the statements of a 'main' function are one obvious example; braces around multiple statements after an if, else, while, or for are another.
- There is no semicolon after the right brace that ends a block.

- C programming language assumes any **non-zero** and **non-null** values as **true**, and if it is either **zero** or **null**, then it is assumed as **false** value.

# *Control Instruction*

- To alter our actions by circumstances.
- If the weather is fine, then I will go for a stroll
- If the highway is busy I would take a diversion
- Notice that all these decisions depend on some condition being met
- C language too must be able to perform different sets of actions depending on the circumstances
- C has three major decision making or conditional statements.

1. **if-else** statement
2. **switch** statement
3. **conditional operator**(**?:**) statement

# *The if Statement*

- The **if - else** statement is used to carry out a logical test and then take one of two possible actions, depending on the outcome of the test (i.e., whether the outcome is true or false).
- The **else** portion of the **if - else** statement is optional. Thus, in its simplest general form, the statement can be written as
- **if *(expression)***

  ***statement***
- The ***expression*** must be placed in parentheses, as shown.
- In this form, the ***statement*** will be executed only if the ***expression*** has a nonzero value (i.e., if ***expression*** is true).
- If the ***expression*** has a value of zero (i.e., if ***expression*** is false), then the ***statement*** will be ignored.
- The ***statement*** can be either simple or compound.

# *The expression in if*

- The expression can be any valid expression:

```
if ( (3 + 3) % 5 )
        printf ( "This works" ) ;
if ( a = 10 )
        printf ( "Even this works" ) ;
if ( -5 )
        printf ( "Surprisingly even this works" ) ;
```

- C a non-zero value is considered to be true, whereas a 0 is considered to be false.

# The if Statement

- Several representative of if statements are shown below.

- if ( x < 0) printf ( "%f", x );

- ```
  if (due > 0)
     credit = 0;
  ```

- ```
  if (x <= 3.0)    {
        y = 3 * pow(x, 2 ) ;
        printf( "%f\n", y) ;
   }
  ```

# The if Statement

- if ((balance < 1000) || (status == ‘R’ ))
  printf( "%f”, balance);

- if (( a >= 0) && (b <= 5 ) )  {
  xmid = (a + b) / 2;
  ymid = sqrt(xmid);
  }

# *The if else Statement*

- The general form of an **if** statement which includes the **else** clause is

- **if *(expression)***

  ***statement* 1**

  **else**

  ***statement 2***

- If the ***expression*** has a nonzero value (i.e., if ***expression*** is true), then ***statement 1*** will be executed.

- Otherwise (i.e., if ***expression*** is false), ***statement 2*** will be executed.

# *The if-else Statement contd…*

- The group of statements after the **if** upto and not including the **else** is called an ‘if block’. Similarly, the statements after the **else** form the ‘else block**’.**
- Notice that the **else is written exactly below the if.**
- Had there been only one statement to be executed in the if block and only one statement in the else block we could have dropped the pair of braces.
- As with the **if** statement, the default scope of **else** is also the statement immediately after the **else.**

# The if else Statement

- Here are several examples illustrating the use of **if-else** statement.
- if(status == 'S')

```
        tax = 0.20 * pay;
else
        tax = 0.14 * pay;
```

- if (due > 0)  {

```
        printf("account number %d is overdue", accountno);
        credit= 0;
}
else
        credit = 1000.0;
```

# The if else Statement

```
if (x <= 3)
        y = 3 * pow(x, 2);
else
        y = 2 * pow((x – 3) , 2 );
 printf( "%f \n”, balance);
```

# The if else Statement

```
if (circle)  {
        scanf("%f", &radius);
        area = 3.14159 * radius * radius;
        printf( “Area of circle= %f”,area);
 }
 else {
        scanf ("%f %f” , &length, &width) ;
        area = length * width;
        printf( " Area of rectangle = %f ”,area);
 }
```

# The if Statement- Example

```
#include <stdio.h>
int main ()  {
   /* variable definition */
   int a = 10;

   /* check the boolean condition using if statement */
   if( a < 20 )   {
      /* if condition is true then print the following */
      printf("a is less than 20\n" );
   }

  printf("value of a is : %d\n", a);
  return 0;
}
```

# The if-else Statement Example

```
#include <stdio.h>
int main () {
   // variable definition
   int a = 100;
   // check the boolean condition
   if( a < 20 )  {
      // if condition is true then print the following
      printf("a is less than 20\n" );
    }
    else   {
      // if condition is false then print the following
      printf("a is not less than 20\n" );
    }
   printf("value of a is : %d\n", a);
   return 0;
}
```

# *The if-else Statement Example*

Output:

a is not less than 20;

value of a is : 100

# *The if else Statement (dangling else problem)*

- Because the else part of an if-else is optional, there is an ambiguity when an else is omitted from a nested if sequence.
- This is resolved by associating the else with the closest previous else-less if.
- For example, in

```
if (n > 0)
   if (a > b)
         z = a;
   else
         z = b;
```

- the else goes with the inner if, as we have shown by indentation.
- If that isn't what you want, braces must be used to force the proper association:

# The nested if Statement

- It is possible to nest (i.e., embed) **if** statements, one within another.
- There are several different forms that nested **if** statements can take.
- The syntax for a nested if statement is as follows −

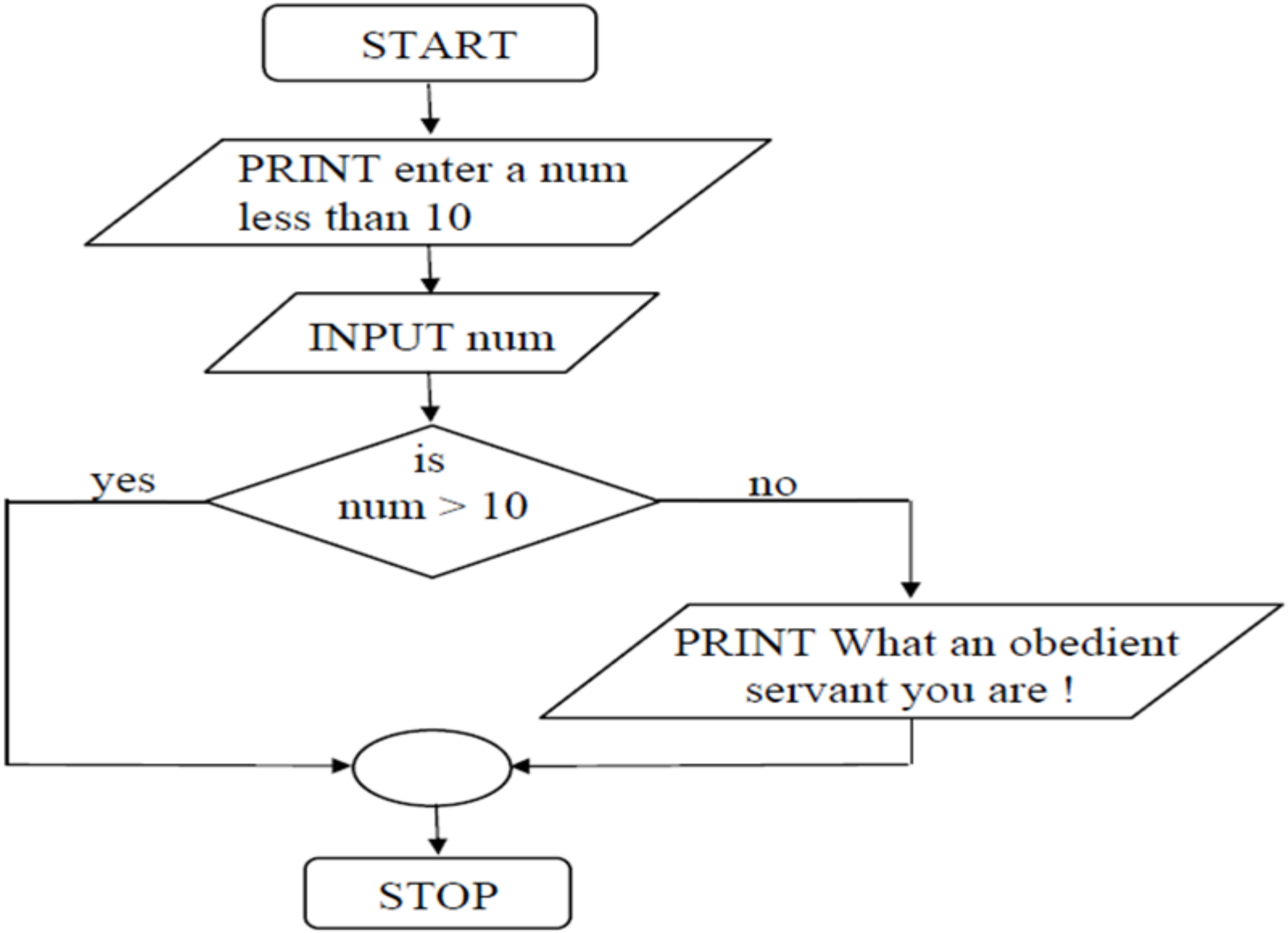
```
if( expression 1)     {
   /* Executes when the  expression 1 is true */
   if(expression 2)    {
      /* Executes when the expression 2 is true */
    }
}
else {
    //else statement block
}
```

# The nested if Statement

```
#include<stdio.h>
void main() {
    int x=20,y=30;
    if(x==20)     {
        if(y==30)   {
            printf("value of x is 20, and value of y is 30.");
         }
         printf(“y is not equal to 30”);
     }
     printf(“this will always print”);
}
```

# The if Statement – Example

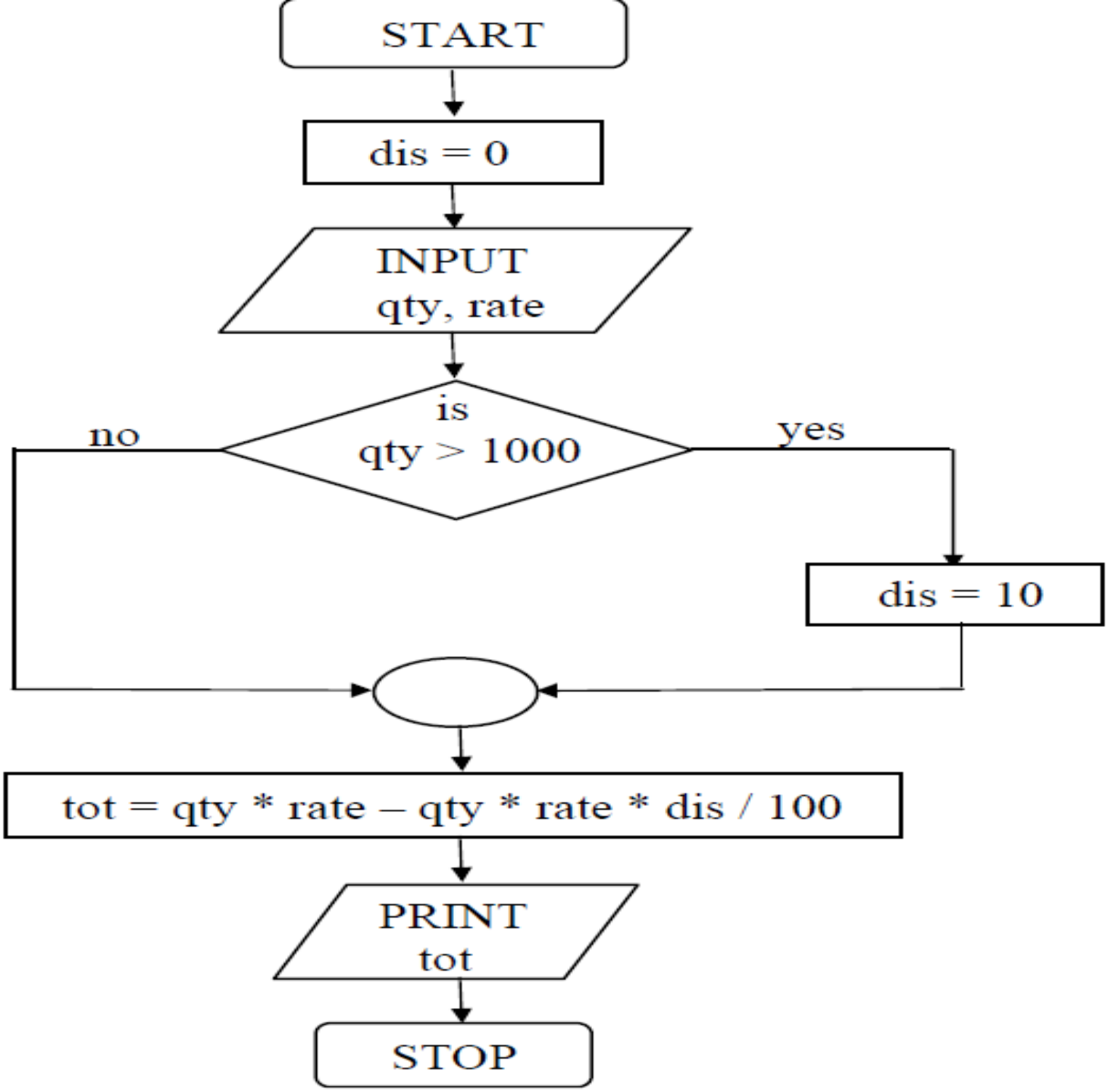
```
/* Demonstration of if statement */
#include<stdio.h>
void main( ) {
    int num ;
    printf ( "Enter a number less than 10 " ) ;
    scanf ( "%d", &num ) ;
    if ( num <= 10 )
        printf ( "What an obedient servant you are !" ) ;
}
```


START
PRINT enter a num
less than 10
INPUT num
is
num > 10
yes
no
PRINT What an obedient
servant you are !
STOP

# The if Statement – Example

While purchasing certain items, a discount of 10% is offered if the quantity purchased is more than 1000. If quantity and price per item are input through the keyboard, write a program to calculate the total expenses.

```
void main( ) {
int qty, dis = 0 ;
float rate, tot ;
printf ( "Enter quantity and rate " ) ;
scanf ( "%d %f", &qty, &rate) ;
if ( qty > 1000 )
        dis = 10 ;
tot = ( qty * rate ) - ( qty * rate * dis / 100 ) ;
printf ( "Total expenses = Rs. %f", tot ) ;
}
```


START
dis = 0
INPUT
qty, rate
is
qty > 1000
no
yes
dis = 10
tot = qty * rate – qty * rate * dis / 100
PRINT
tot
STOP

# *The if Statement – Example*

- The current year and the year in which the employee joined the organization are entered through the keyboard. If the number of years for which the employee has served the organization is greater than 3 then a bonus of Rs. 2500/- is given to the employee. If the years of service are not greater than 3, then the program should do nothing.

```
#include<stdio.h>
void main( )
{
int bonus, cy, yoj, yr_of_ser ;
printf ( "Enter current year and year of joining " ) ;
scanf ( "%d %d", &cy, &yoj ) ;
yr_of_ser = cy - yoj ;
if ( yr_of_ser > 3 )
        {
                bonus = 2500 ;
                printf ( "Bonus = Rs. %d", bonus ) ;
        }
}
```

# *The if-else Statement Example*

- **Example.** In a company an employee is paid as under: If his basic salary is less than Rs. 1500, then HRA = 10% of basic salary and DA = 90% of basic salary. If his salary is either equal to or above Rs. 1500, then HRA = Rs. 500 and DA = 98% of basic salary. If the employee's salary is input through the keyboard write a program to find his gross salary.

```
#include<stdio.h>
void main( )
{
float bs, gs, da, hra ;
printf ( "Enter basic salary " ) ;
scanf ( "%f", &bs ) ;
if ( bs < 1500 )
        {
                hra = bs * 10 / 100 ;
                da = bs * 90 / 100 ;
        }
else
        {
                hra = 500 ;
                da = bs * 98 / 100 ;
        }
gs = bs + hra + da ;
printf ( "gross salary = Rs. %f", gs ) ;
}
```

# Nested if...else statement (if...else if....else Statement)

- The nested if...else statement allows you to check for multiple test expressions and execute different codes for more than two conditions.
- When a logical expression is encountered whose value is nonzero (true), the corresponding statement will be executed and the remainder of the nested **if - else** statements will be bypassed.
- Thus, control will be transferred out of the entire nest once a true condition is encountered.
- The final **else** clause will apply if none of the expressions is true. It can be used to provide a default condition or an error message.

# *The if…else if…else Statement*

- An **if** statement can be followed by an optional **else if...else** statement, which is very useful to test various conditions .
- else if statements in C is like another if condition, it's used in program when if statement having multiple decisions.
- When using if...else if…else statements, there are few points to keep in mind −

- An if can have zero or one else's and it must come after any else if's.
- An if can have zero to many else if's and they must come before the else.
- Once an else if succeeds, none of the remaining else if's or else's will be tested.

# Nested if...else statement (if...else if....else Statement)

```
if (testExpression1) {
// statements to be executed if testExpression1 is true
}
else if(testExpression2)  {
  // statements to be executed if testExpression1 is false and testExpression2 is true
}
else if (testExpression 3) {
 // statements to be executed if testExpression1 and testExpression2 is false and testExpression3 is true
}
……..
else {
   // statements to be executed if all test expressions are false
}
```

# Nested if...else statement (if...else if....else Statement) Example

```
#include <stdio.h>
int main () {
   int a = 100;
   if( a == 10 )    {
     printf("Value of a is 10\n" );
   }
   else if( a == 20 )    {
      printf("Value of a is 20\n" );
   }
  else if( a == 30 )    {
    printf("Value of a is 30\n" );
 }
  else   {
      printf("None of the values is matching\n" );
   }
 printf("Exact value of a is: %d\n", a );
   return 0;
}
```

Output:

None of the values is matching

Exact value of a is: 100

# Nested if...else statement (if...else if....else Statement) Example

```
// Program to relate two integers using =, > or <

#include <stdio.h>
int main() {
    int number1, number2;
    printf("Enter two integers: ");
    scanf("%d%d",&number1, &number2);
    //checks if two integers are equal.
    if(number1 == number2)    {
        printf("Result:%d= %d",number1,number2);
    }
    //checks if number1 is greater than number2.
    else if (number1 > number2)    {
        printf("Result:%d > %d", number1, number2);
    }

    // if both test expression is false
    else    {
        printf("Result: %d < %d",number1, number2);
    }
    return 0;
}
```

# Nested if-else Example

```
#include<stdio.h>.
void main( ) {
        int i ;
        printf ( "Enter either 1 or 2 " ) ;
        scanf ( "%d", &i ) ;
        if ( i == 1 )
                printf ( "You would go to heaven !" ) ;
        else {
              if ( i == 2 )
                    printf ( "Hell was created with you in mind" ) ;
              else
                    printf ( "How about mother earth !" ) ;
        }
}
```

# *Forms of if*

- The **if statement can take any of the following forms:**

```
(a) if ( condition )
                    do this ;
(b) if ( condition ) {
                    do this ;
                    and this ;
          }
(c) if ( condition )
                    do this ;
     else
                    do this ;
```

# *Forms of if*

```
(d) if ( condition )     {
                    do this ;
                    and this ;
          }
   else {
                    do this ;
                    and this ;
          }
```

# *Forms of if*

```
(e) if ( condition )
                    do this ;
    else {
                    if ( condition )
                              do this ;
                    else      {
                              do this ;
                              and this ;
                    }
    }
```

```
(f) if ( condition ) {
                    if ( condition )
                              do this ;
                    else {
                              do this ;
                              and this ;
                    }
  }
   else
                    do this ;
```

# *Nested if-else Example*

- The marks obtained by a student in 5 different subjects are input through the keyboard. The student gets a division as per the following rules:

  Percentage above or equal to 60 - First division

  Percentage between 50 and 59 - Second division

  Percentage between 40 and 49 - Third division

  Percentage less than 40 - Fail

  Write a program to calculate the division obtained by the student.

```
#include<stdio.h>
void main( )
{
int m1, m2, m3, m4, m5, per ;
printf ( "Enter marks in five subjects " ) ;
scanf ( "%d %d %d %d %d", &m1, &m2, &m3, &m4, &m5 ) ;
per = ( m1 + m2 + m3 + m4 + m5 ) / 5 ;
        if ( per >= 60 )
                printf ( "First division ") ;
        else
        {
                if ( per >= 50 )
                        printf ( "Second division" ) ;
                else
                        {
                                if ( per >= 40 )
                                    printf ( "Third division" ) ;
                                else
                                    printf ( "Fail" ) ;
                        }
        }
}
```

```
#include<stdio.h>                    // PRGRAM WITH nested if else.
void main( )
{
int m1, m2, m3, m4, m5, per ;
printf ( "Enter marks in five subjects " ) ;
scanf ( "%d %d %d %d %d", &m1, &m2, &m3, &m4, &m5 ) ;
per = ( m1 + m2 + m3 + m4 + m5 ) / 5 ;
if ( per >= 60 )
        printf ( "First division" ) ;
else if ( per >= 50 )
        printf ( "Second division" ) ;
else if ( per >= 40 )
        printf ( "Third division" ) ;
else
        printf ( "Fail" ) ;
}
```