**EXPERIMENT-4: Conditional Branching**

**Objective:** To understand conditional logic of execution; applying conditional branching structures in C (if, if-else, if-else-if ladder, nested-if, switch-case).

**List of Lab Activities:**

Write algorithm and C program, compile, execute and test the code using Linux C compiler with suitable test cases.

-------------------------------------------------------------------

**Experiment 4.1** - “Find the biggest of 3 numbers.”

**ALOGRITHM: ---**

**Step 1::** Take three random numbers and put in

variables a, b, c

Step 2: if a>b go to Step 3, otherwise go to Step 4,

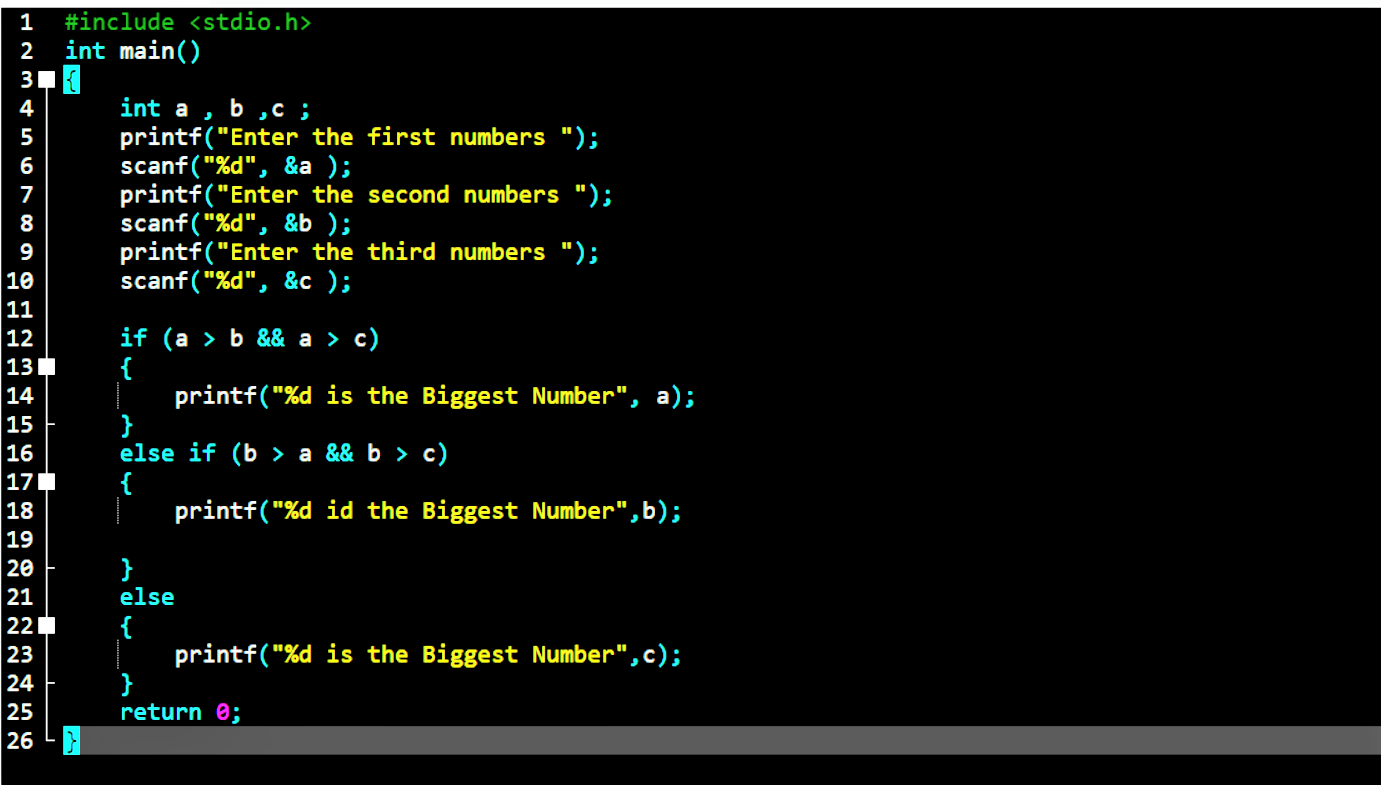
Step 3: if a>c go to Step 5, otherwise go to Step 7,

Step 4: if b>c go to Step 6,otherwise go to Step 7,

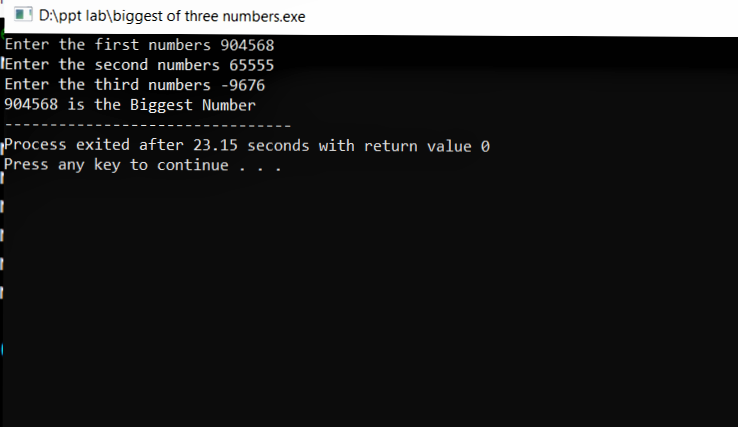
Step 5: Display output "a is the largest "

Start 6: Display output "b is the largest"

Start 7: Display output “ c is the largest”

C program ---

Execution---



**Experiment 4.2** – “Check whether a given year is

leap year or not.”

**ALOGRITHM**: ---

Step 1::: Take the year you want to check and put in

variables a,

Step 2: if a%400==0 and a%4==0 then go to Step 4,

otherwise go to Step 3

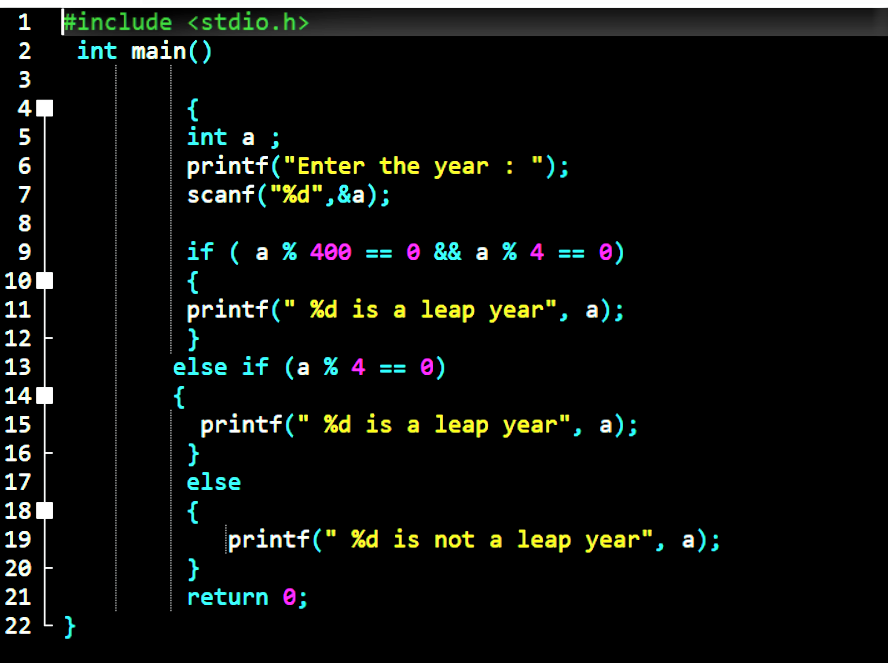
Step 3:if a%4==0 then go to Step 4, otherwise go to step6

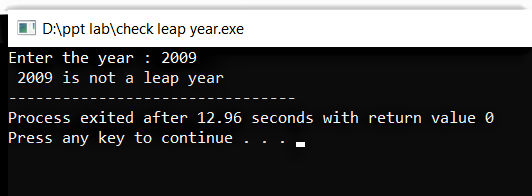
Step 4: Display output "a is a leap year " and Stop the

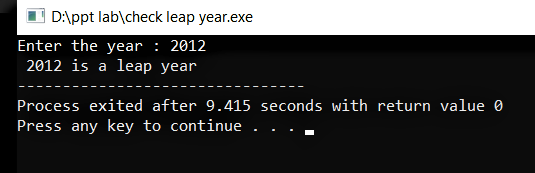
program

Start 5: Display output “a is not a leap year " and Stop

the program

C program---

Execution---



**Experiment 4.3**- “Find the roots of quadratic equation.”

**ALOGRITHM**: ---

Step 1:write the equation in form of aX2 + bX + c form

Step 2: Enter value of a, b ,c

Step 3:Find d = (b2 – 4ac) to decides the nature of roots

Step 4: If d> 0, then roots are real and go to Step 5,

If d=0, then roots are same and go to Step 7,

If d < 0, then roots are imaginary and go to step 9

Step 5: Calculate

root 1= (-b + sqrt (d ) / ( 2 \* a ));

root 2= (-b - sqrt (d ) / ( 2 \* a ));

Step 6: Print root 1 and root 2 and Stop the program

Step 7: calculate

root1 = root 2 = (-b / ( 2 \* a ))

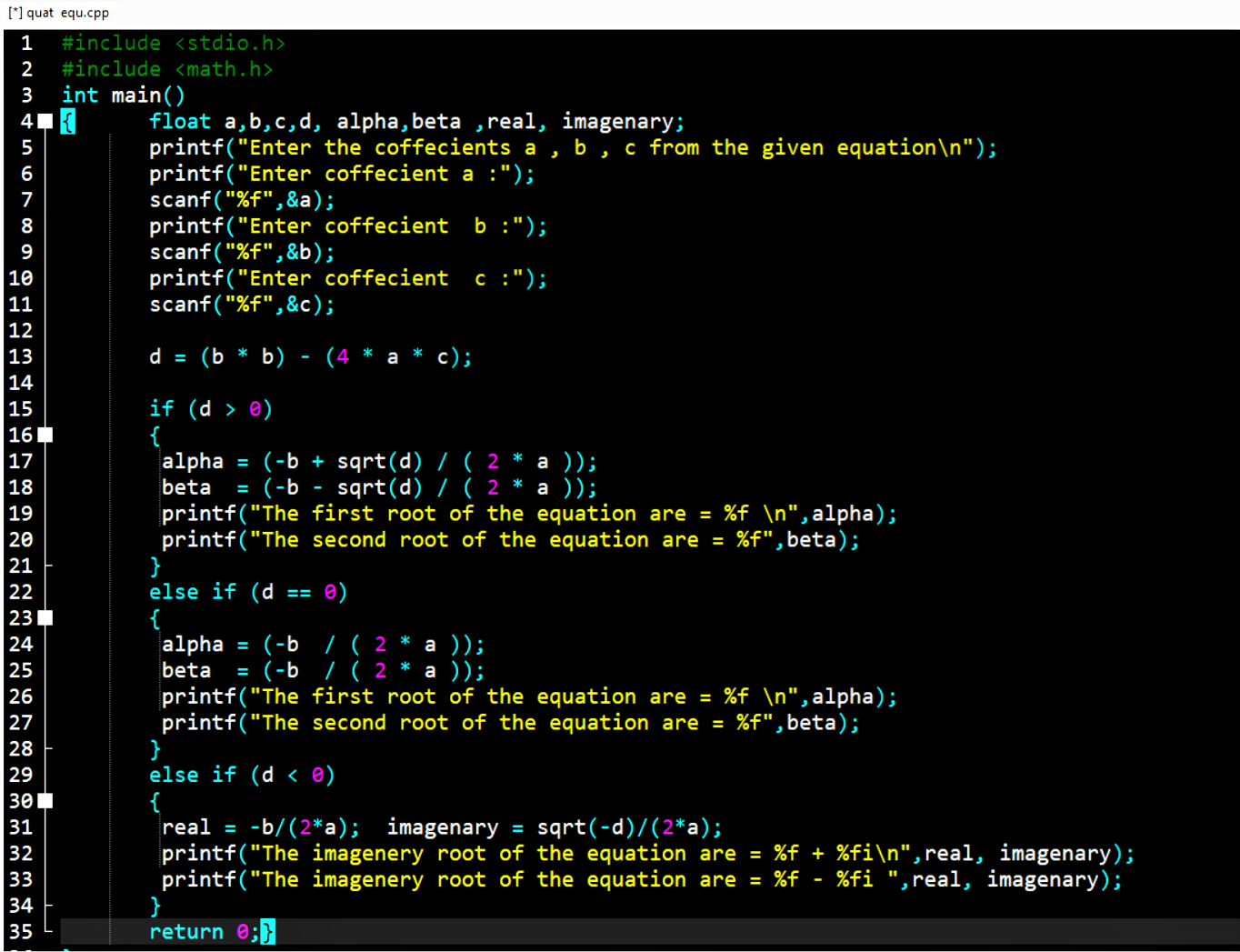
Step 8:Print root 1 and root 2 and Stop the program

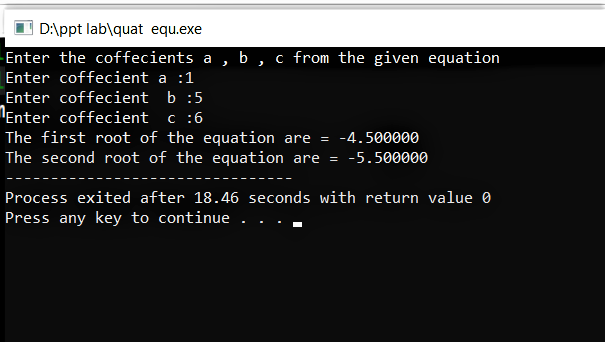
Step 9: calculate

root 1 = -b/(2\*a) +sqrt(-d)/(2\*a)

root 2 = -b/(2\*a) +sqrt(-d)/(2\*a)

Step 10: Print root 1 and root 2 and Stop the program

C-program---

Execution---

**Experiment 4.4** – “Check if a given character is a

Vowel or consonant using Switch-Case statement.”

**ALOGRITHM:** ----   
Step 1: Read the alphabet entered and store in a declared

variable ‘x’ of character datatype.

Step 2: then compare ‘x’ with the vowels in both upper

and lower case. If the cases matches the go to

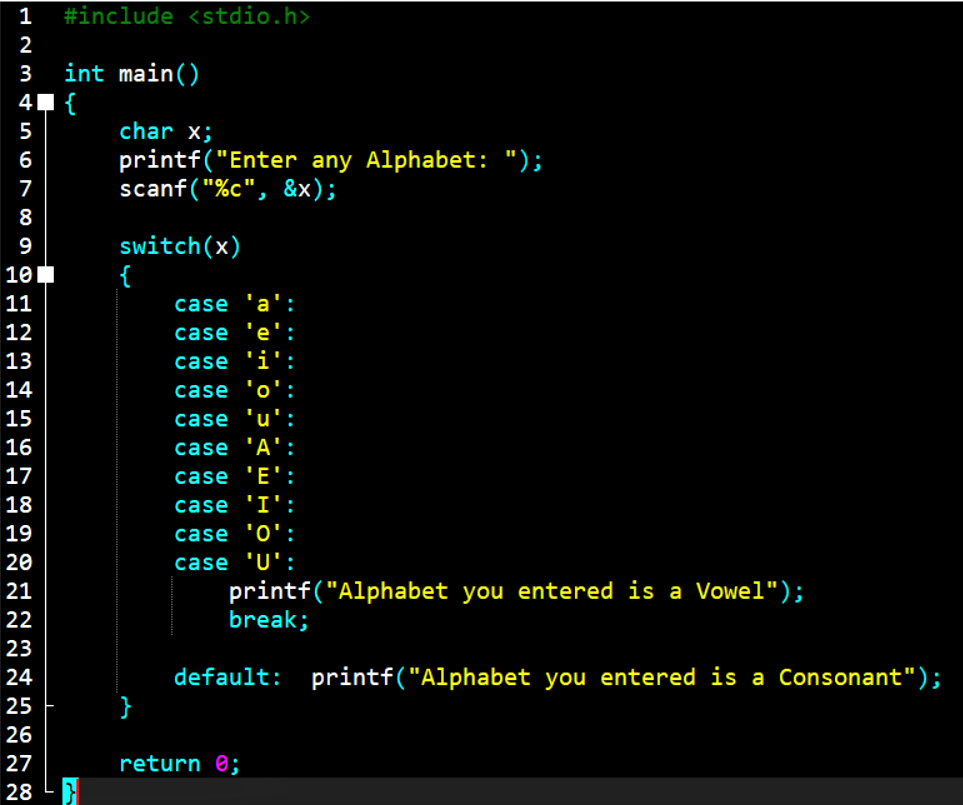
step3 otherwise go to step 4

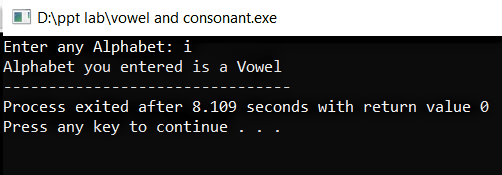
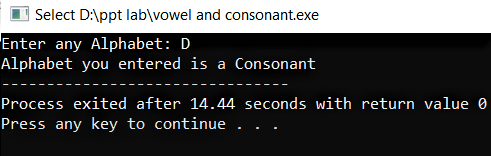
Step 3: Print “Alphabet you entered is a vowel” and

Stop the program.

Step 4: Print “Alphabet you entered is a Consonant” and

Stop the program.

C program---

Execution---

**Experiment 4.5**- “Find whether a given number is

even or odd.”

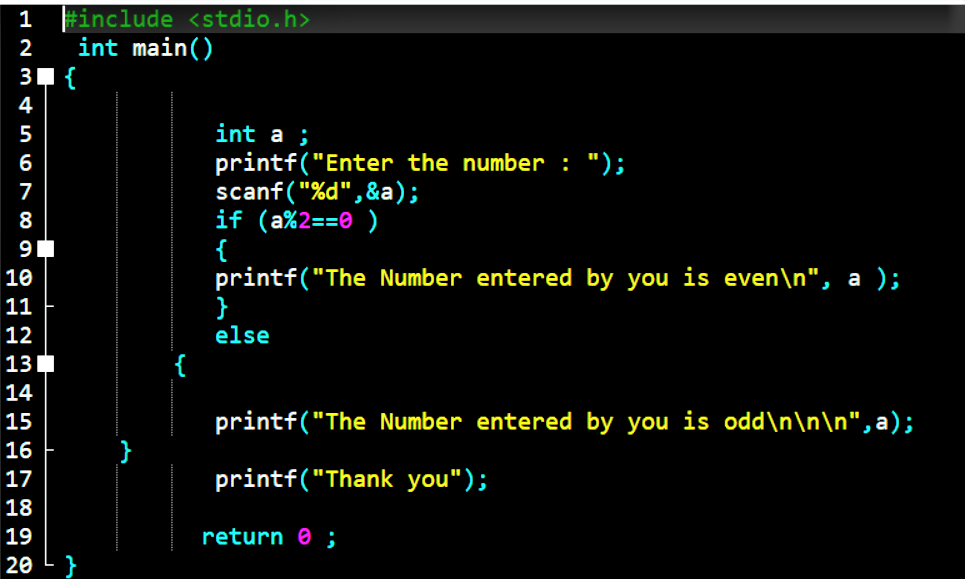
**ALOGRITHM**: ---

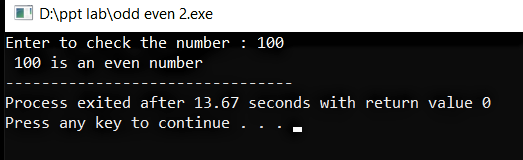
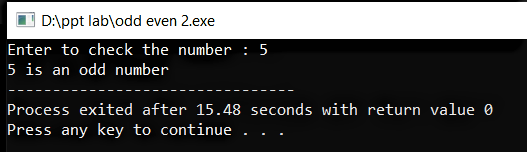
Step 1:Take the number and store it in a variable a

Step 2: If a%2==0 the go to Step 3, if not then Step 4

Step 3: Print “a is an Even number” and Stop the program

Step 4: Print “a is Odd number” and Stop the program

C-program---

Execution---

**Experiment 4.6**- “Find if the given number is positive,

negative or zero

**ALOGRITHM**: ---

Step 1:Take the number and put in variable ‘a’

Step 2: If a<0 then go to Step 3,

If a> 0 then go to Step 4,

If a== 0 then go to Step 5,

Step 3: Print “a is Positive number” and Stop the program

Step 4: Print “a is Negative number” and Stop the program

Step 5: Print “The number you entered is Zero” and Stop the

Program.

C-program---Text

Description automatically generated

Execution---Graphical user interface, text

Description automatically generatedText

Description automatically generated

**Experiment 4.7** – “Check if the given number is

divisible by 2 and 3 or not.”

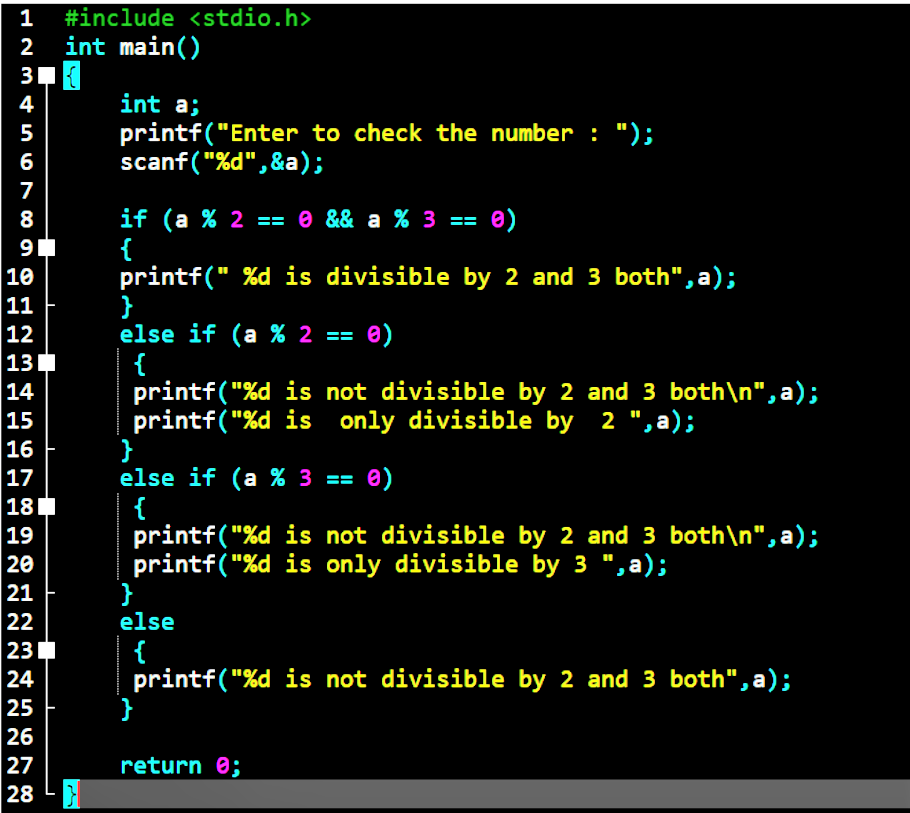
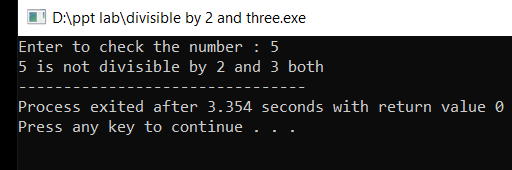
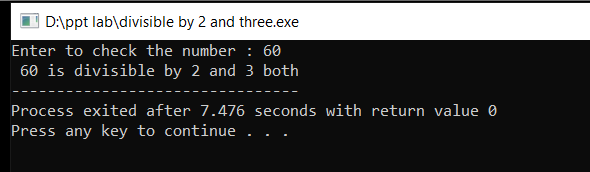
**ALOGRITHM**: ---

Step 1: take the number and put it in variable ‘a’

Step 2: If a%2==0 and a%3==0 then go to Step 3, otherwise go to Step 4

Step 3: Print “a is divisible by 2 and 3 ” and Stop the program

Step 4: Print “a is divisible by 2 and 3 or not” and Stop the program.

C-program---Execution---

**Experiment 4.8** – “Check whether a given Character is

an alphabet, digit or special character.”

**ALOGRITHM**: ---

Step 1: Take the character you entered and put it in variable ‘character’.

Step 2: If (character>='0' && character<='9') then go to

Step 4, otherwise go to Step 3

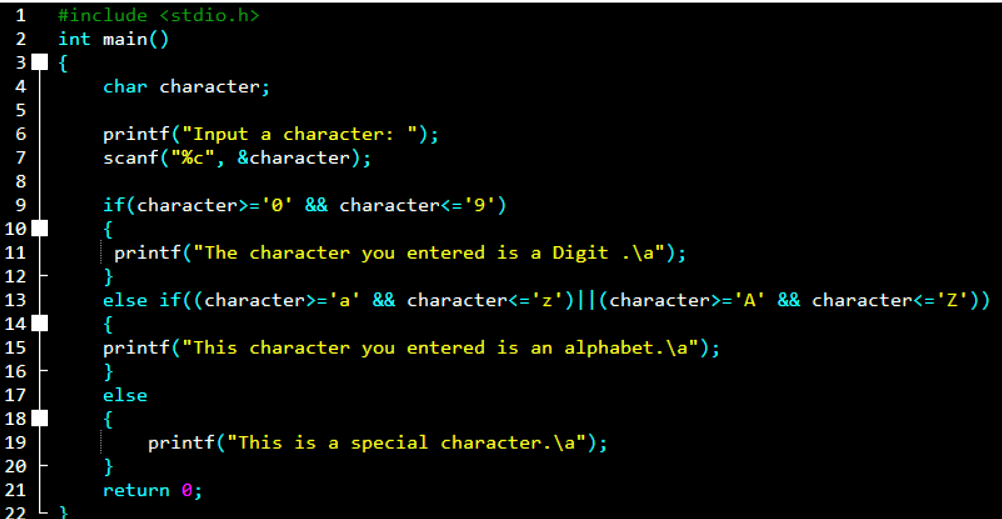
Step 3:If

((character>='a' && character<='z')||(character>='A' && character<='Z') then go to Step 5,otherwise go Step 6

Step 4: Print “The character you entered is a Digit” and Stop the program

Step 5: Print “This character you entered is an alphabet.” and Stop the program

Step 6: Print “This is a special character” and stop the program.

C-program---

Execution---