Experiment 7 :

Illustrate the following controls statements in R:

- if and else
- ifelse
- switch

Aim: To understand the working of different control statements in R.

Experiment 7: Control statements in R

- Control statements allow us to control the flow of our programming and cause different things to happen, depending on the values of tests.

- Tests result in a logical, TRUE or FALSE.

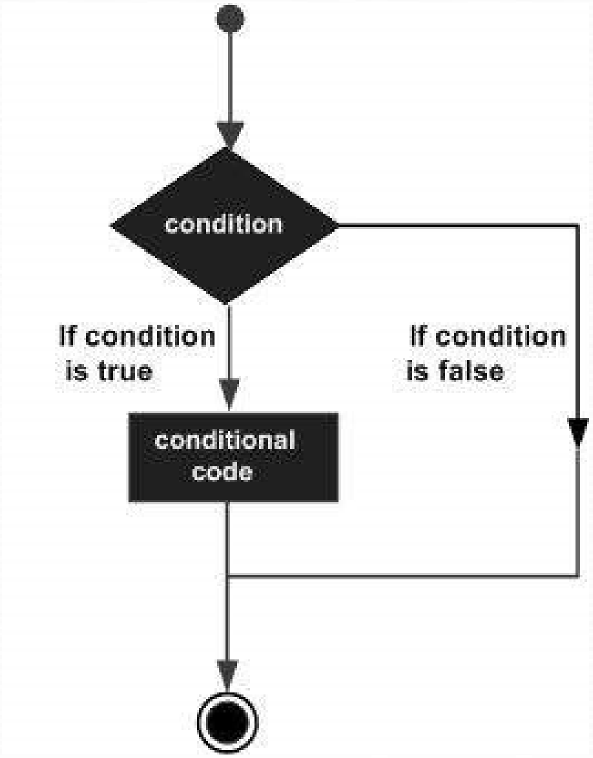
Experiment 7: Control statements in R

- **The main control statements are :**
  - if
  - ifelse
  - switch

Exp. 7: Control statements in R: IF-ELSE

- We can have a control structure with just an if block.
- if (Boolean Expression){

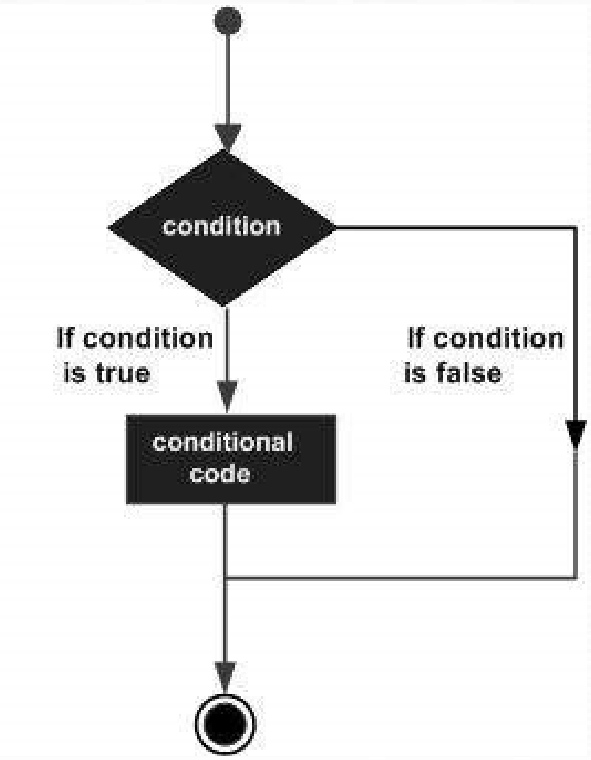
```
        <Code Block>

        }
```



Exp. 7: Control statements in R: IF-ELSE

Example:

```
if (n>0){
    print(“True”)
    }
```



Exp. 7: Control statements in R: IF-ELSE

- A control structure with if-else can have an **"if"** block followed by multiple optional **"else if"** blocks and followed by an optional **"else"** block.

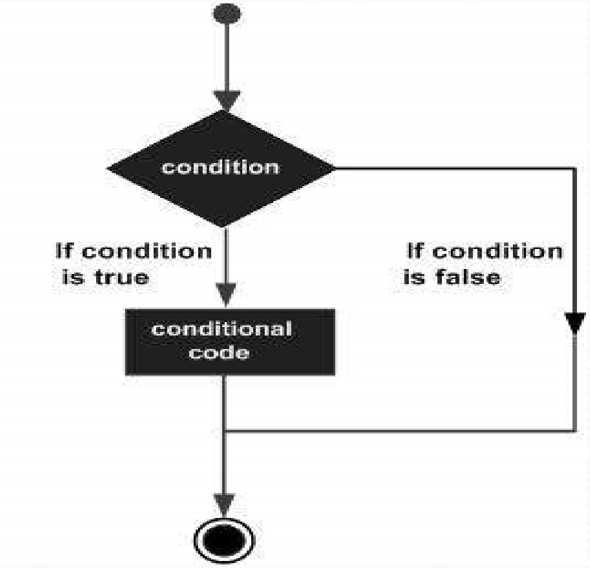
- if (Boolean Expression){

  <Code Block1>

  }else {

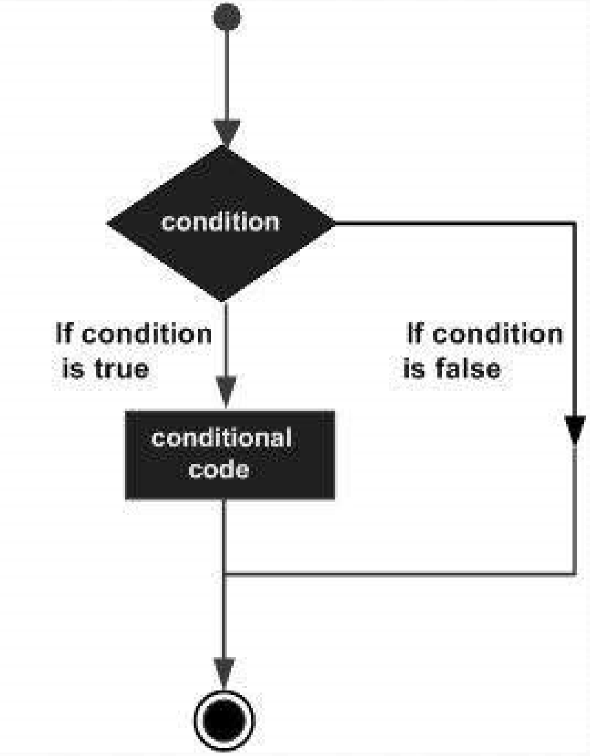
  <Code Block 2>

  }



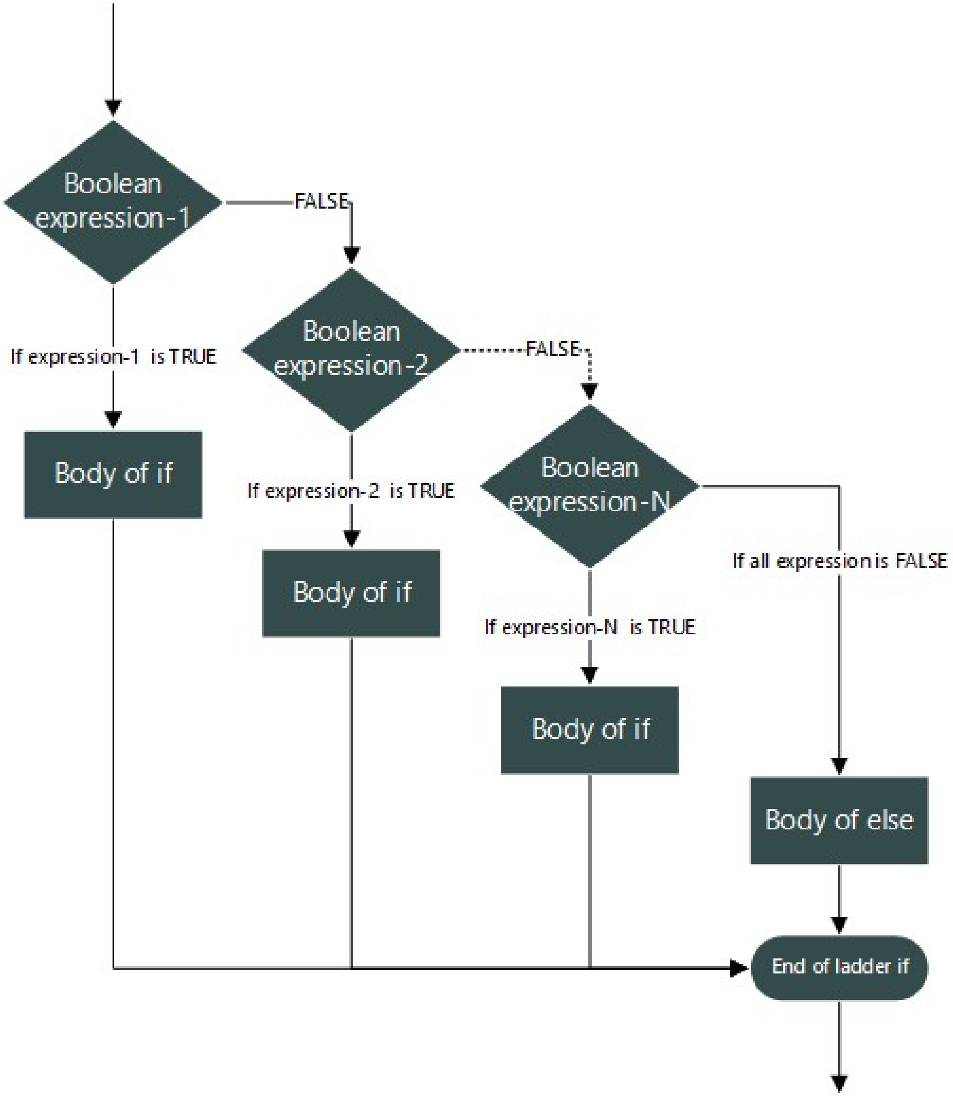
Exp. 7: Control statements in R: IF-ELSE

```
if (n%%2==0){
    print(“Even”)
    }else{
    print(“Odd”)
    }
```



Exp. 7: Control statements in R: IF-ELSE

We can have a control structure with an if followed by else if and else blocks.



Exp. 7: Control statements in R: IF-ELSE

```
if (Boolean Expression){
        <Code Block 1>
} else if(Boolean Expression){
        <Code Block 2>
        }
        .
        .
}else if (Boolean Expression){
        <Code Block n-1>
        }else{
        <Code Block n>
        }
```

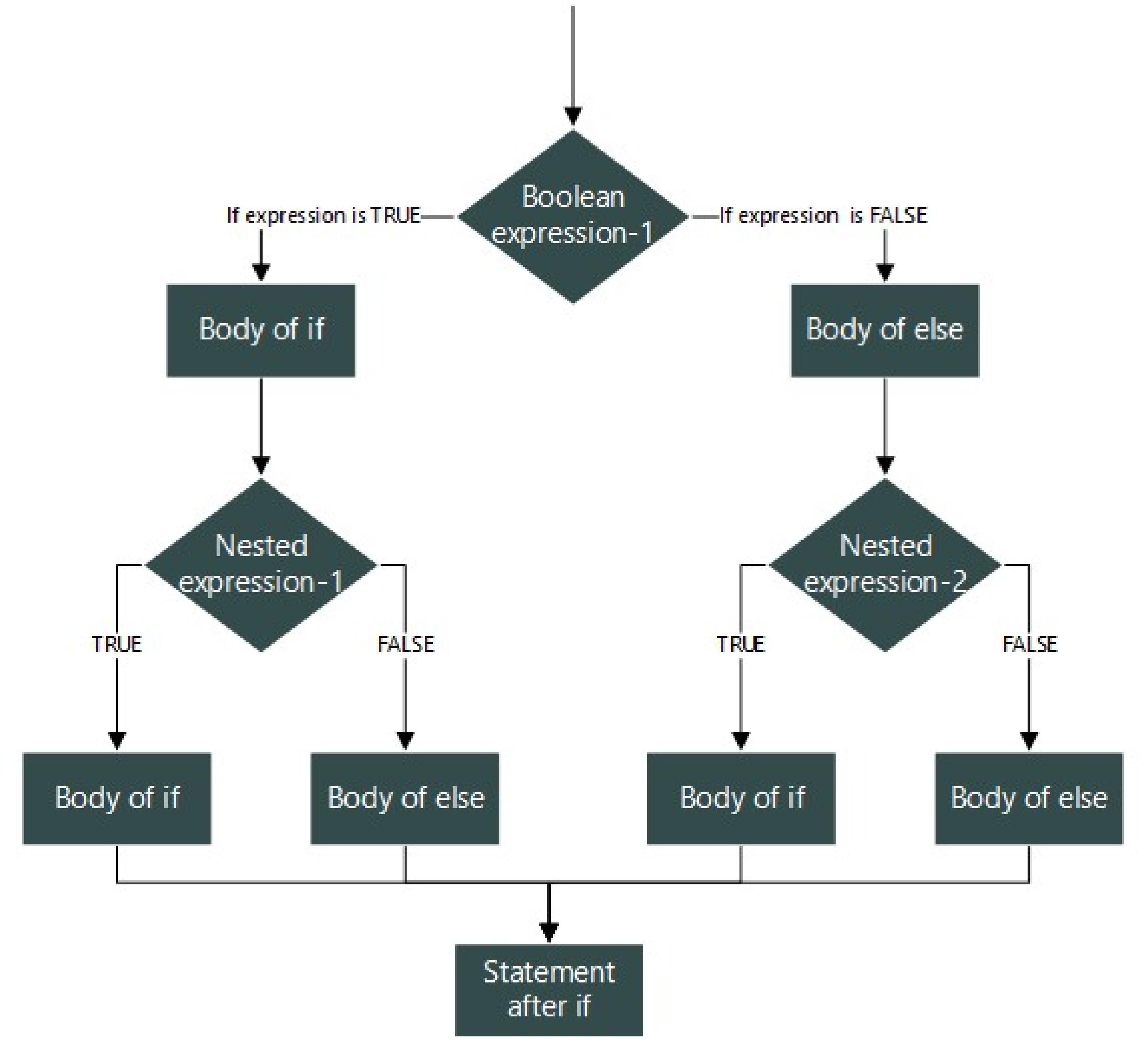
Exp. 7: Control statements in R: IF-ELSE

Example: age.R

```
age=readline(“Enter age:”)
age=as.integer(age)
if (age<=12){
print("Child")
}else if(age<=19){
print("Teenager")
}else if (age<=30){
print("Adult")
}else if(age<=45)
{ print("Middle Aged") }
else {
print("Old")
 }
```

Exp. 7: Control statements in R: IF-ELSE

We can create nested if else control structures.



Exp. 7: Control statements in R: IF-ELSE

- ifelse is a ternary statement that is a online equivalent of a simple if-else structure.

- **Syntax:**

  ifelse(BooleanExpression,<Exp1>,<Exp2>)

Exp. 7: Control statements in R: IF-ELSE

- ifelse(BooleanExpression,<Exp1>,<Exp2>)

- Equivalent if-else Structure:

```
if (Boolean Expression){
        <Exp1>
        }else{
        <Exp2>
        }
```

Exp. 7: Control statements in R: IF-ELSE

Examples:

```
a=c(1:10)
b=ifelse(a %% 2 == 0,"even","odd")
print(b)

a=6
b=ifelse(a%%2==0,"even","odd")
print(b)
```

Exp. 7: Control statements in R: IF-ELSE

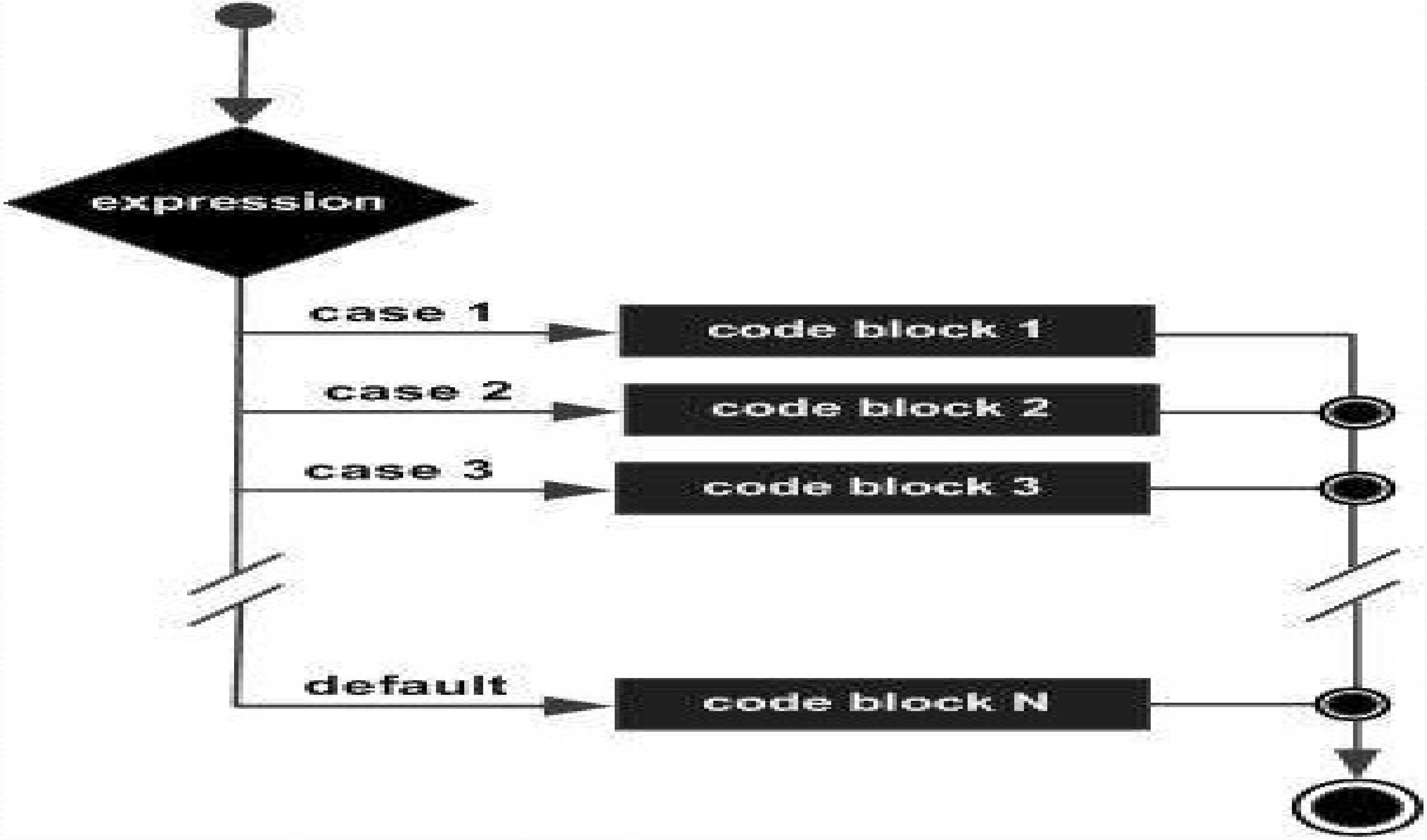
- If we have multiple cases to check, writing else if repeatedly can be cumbersome and inefficient.

- This is where switch is most useful.

Exp. 7: Control statements in R: IF-ELSE

Syntax:

switch(expression, case1, case2, case3....)



Exp. 7: Control statements in R: IF-ELSE

- There are basically **two ways** in which one of the cases is selected:
  - **Based on Index:**
    - If the cases are values like a character vector, and the expression is evaluated to a number than the expression's result is used as an index to select the case.

Exp. 7: Control statements in R: IF-ELSE

Example:

```
s=switch(4,"Ramya","Surya","Rama","Sitha")
print(s)
```

Exp. 7: Control statements in R: IF-ELSE

- **Based on Matching Value:**
  - ❖ When the cases have both case value and output value like ["case_1"="value1"], then the expression value is matched against case values. If there is a match with the case, the corresponding value is the output.

Exp. 7: Control statements in R: IF-ELSE

Example:

```
a=10
b=2
cat("Enter Your Choice:\n1 for add \n2 for sub \n3 for Div \n4 for mul")
y=readline()
switch(y,
        "1"=cat("Addition=",a+b),
         "2"=cat("Subtraction=",a-b),
        "3"=cat("Division= ",a/b),
        "4"=cat("multiplication =",a*b)
 )
```