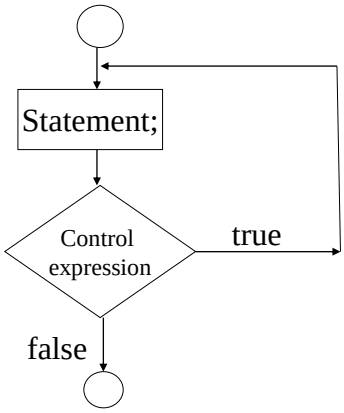
Iterative Control Structures

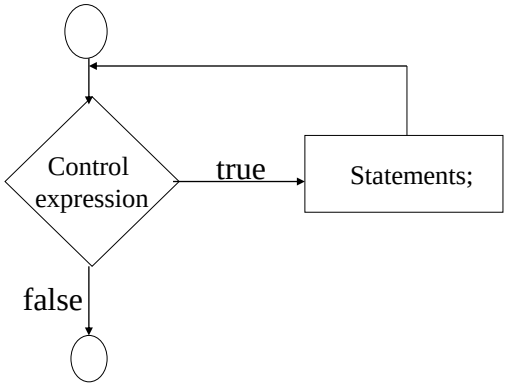
do – while loop

- ***while*** statement - test for continuation of the loop is carried out at the beginning of each pass through the loop
  - Pre-test
- ***do-while*** statement - the test for continuation is at the end of each pass
  - Post-test
  - statement will always be executed ***at least*** once

do/while Flowchart



while Flowchart



do/while loop

1. Execute the statements in the loop
2. Evaluate the control expression
3. While the control expression is true go back to 1
4. When the control expression is false exit the loop and execute the next statement after the loop

do-while loop

Syntax or General Form

```
do{
   statements;
}while (expression);
```

- Executes the statement or statements while the expression is true
- Code will always be executed at least once
- Curly brackets not necessary – but good programming

while vs do-while

- A ***do-while*** will execute at *least* once
- A ***while*** may *never* execute

- A ***while*** executes zero or more times
- A ***do-while*** executes one or more times

- A ***while*** is pre-test
- A ***do-while*** is post-test

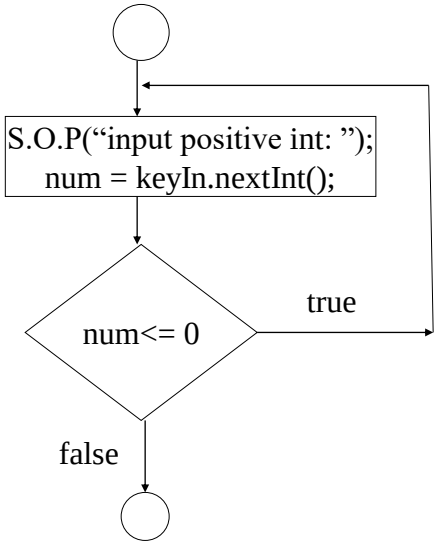
Sample Code 1

- Read in a positive integer

```
do{
   System.out.print("Input a positive integer: ");
   num = keyIn.nextInt( );
}while (num <= 0);   //while num is negative
```

- As long as a *nonpositive* integer is entered, the user will be told to enter a positive integer.
- Loop will terminate once a positive number is entered.

do/while Flowchart



Sample Code 2
If negative – display error message

```
do{
   System.out.print("Input a positive integer:");
   num = keyIn.nextInt( );
   if(num<=0)
   {
     System.out.println("Invalid number entered");
   }
}while (num <= 0);
```

Sample Code 2– using ***while***

```
int num ;

System.out.print("Enter a positive integer value:  " );
num = keyIn .nextInt();

while(num <= 0)
{
       System.out.println("Invalid number entered");
   System.out.print("Enter a positive integer value: ");
   num = keyIn .nextInt();

} //end while

System.out.println("Number entered is " +num);
```

Consider the question…

Write a program that will repeatedly display the following menu on the screen until the user enters the number 3:-

Main menu
1. Option 1
2. Option 2
3. Quit
Please enter choice:-

If the user enters **1** then the program should display "*Option 1 chosen*….". It should then redisplay the menu

If the user enters **2** then the program should display "*Option 2 chosen*….". It should then redisplay the menu

If the user enters **3** then the program should end.

If the user enters **any other option** then the program should display "*Invalid Option chosen – please enter 1, 2, or* 3…". It should then redisplay the menu

Pseudocode

```
DO
    DISPLAY menu

    GET option

    IF option is 1
            DISPLAY option 1 chosen
    ELSE IF option is 2
            DISPLAY option 2 chosen
    ELSE IF option is 3
            quit
    ELSE
            DISPLAY invalid option chosen

WHILE option is not 3
```

Solution

```
int option;

do{
   //display menu
   System.out.println("\nMain Menu\n");
   System.out.println("1. Option 1\n2. Option 2\n3. Quit\n");

   //get user option
   System.out.print("Please enter option: ");
   option = keyIn.nextInt();
   switch(option){
        case 1:  System.out.println("Option 1 chosen...");
                 break;
        case 2:  System.out.println("Option 2 chosen...");
                 break;
        case 3:  System.out.println("You have chosen quit...");
                 break;
        default: System.out.println("Invalid option - please"
                        +"enter 1, 2 or 3...");
   } //end switch
}while(option != 3);
```

Sample Code 4 – using char

```
char response;
int num;
do {
  System.out.print("Enter number:  " );
  num = keyIn.nextInt();
  System.out.print("Do you want to change number?");
  response = keyIn.next ().charAt(0);
     //while response is yes
}while(response == 'y' ||  response == 'Y' );

System.out.println("final number  "  +num  );
```

Sample Code 4 – using String

```
String response;
int num;
do {
  System.out.print("Enter number:  " );
  num = keyIn.nextInt();
  System.out.print("Do you want to change number?");
  response = keyIn.next ();
}while(response.equalsIgnoreCase("yes")); //while yes

System.out.println("final number  "  +num  );
```