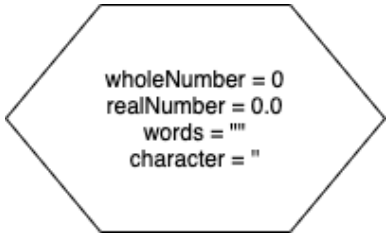
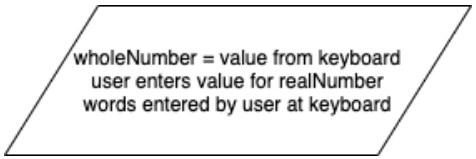
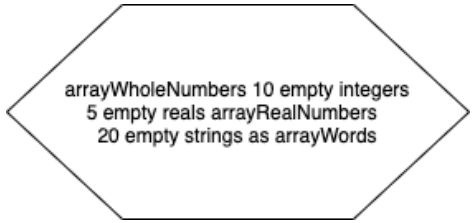
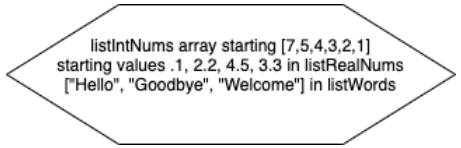
Flow Chart	Pseducode	Java
Variable declarations		
	WholeNumber integer = 0 realNumber = 0.0 [a real number] words as a String = “” char as character containing a	`int wholeNumber = 0;` `double realNumber = 0.0;` `String words = "";` `char character = 'a';`
Keyboard Input		
	User enters value for wholeNumber from keyboard Ask user to enter realNumber from keyboard Get value for words from user using keyboard	`wholeNumber = Keyboard.getInt("Question to display goes here");` `realNumber = Keyboard.getReal("Question to display goes here");` `words = Keyboard.getText("Question to display goes here");`
Declaring Arrays [Empty]		
	An empty array of 10 integers called arrayWholeNumbers An array called arrayRealNumbers containing 5 empty doubles An array of 20 empty Strings called arrayWords	`int [] arrayWholeNumbers = new int [10];` `double [] arrayRealNumbers = new double [5];` `String [] arrayWords = new String [20];`

Declaring Arrays [Pre-Defined]



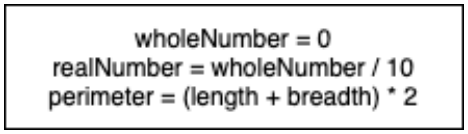
An array of int called listIntNums starting with 7,5,4,3,2,1

An array called listRealNums containing 0.1, 2.2, 4.5, 3.3

The values Hello, Goodbye, Welcome stored in an array of Strings called listWords

```
int [] listIntNums = {7, 5, 4, 3, 2, 1};
double [] listRealNums = { 0.1, 2.2, 4.5, 3.3};
String [] listWords = {"Hello","Goodbye","Welcome"};
```

Assignments and Calculations



wholeNumber = 0

realNumber equals wholeNumber divided by 10

Perimeter = (length + breadth) * 2

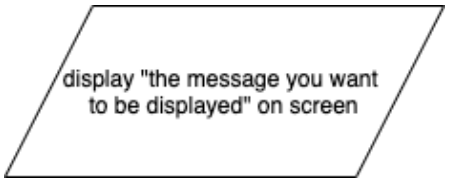
```
wholeNumber = 0;

realNumber = wholeNumber / 10;
```

In java you only need to declare the type (int, double etc) of the variable once

```
perimeter = 2*(length+breadth);
```

Data Output

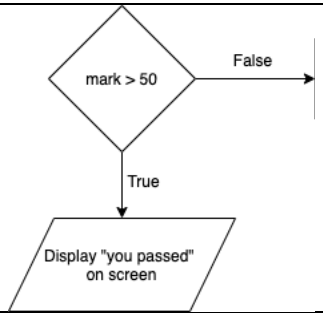


Display “the message you want to be displayed” for the user on the screen

```
Screen.display("The message you want displayed", "Task Heading");
```

Data Output (multiple)		
		`String output = "";` `String heading = "Heading of message";` `output = output + "Next part of message" + "\n";` `Screen.display(output, heading);` The “\n” adds a new line to the output The output = output + “new text” concatenates the new text on to the ned of output
Logic Operators and Comparisons		
	wholeNumber MORE THAN realNumber wholeNumber LESS THAN realNumber wholeNumber LESS THAN OR EQUAL realNumber wholeNumber MORE THAN OR EQUAL realNumber wholeNumber EQUAL realNumber wholeNumber NOT EQUAL realNumber words EQUALS “Hello” words NOT EQUALS “Hello” words EQUALS “Hello” AND wholeNumber MORE THAN OR EQUAL realNumber words EQUALS “Hello” OR wholeNumber MORE THAN OR EQUAL realNumber	`(wholeNumber > realNumber)` `(wholeNumber < realNumber)` `(wholeNumber <= realNumber)` `(wholeNumber >= realNumber)` `(wholeNumber == realNumber)` `(wholeNumber != realNumber)` `(words.equals("Hello"))` `(!(words.equals("Hello")))` `((words.equals("Hello")) && (wholeNumber >= realNumber))` `((words.equals("Hello")) || (wholeNumber >= realNumber))` AND is && NOT is ! OR is ||

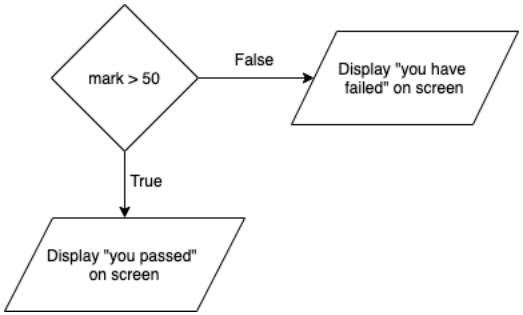
Selection (simple)



1. If mark is greater than 50 then
2. display you passed on screen
3. End if

```
heading = "Test Calculator";
if (mark > 50) {
    output = "Well done you passed";
}
Screen.display(output, heading);
```

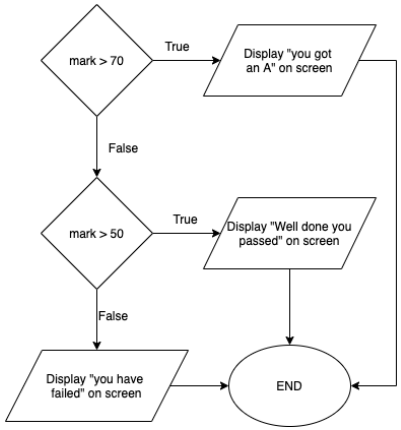
Selection (with else)



1. If mark is greater than 50 then
2. display you passed on screen
3. else
4. display you have failed
5. End if

```
heading = "Test Calculator";
if (mark > 50) {
    output = "Well done you passed";
} else {
    output = "You have failed";
}
Screen.display(output, heading);
```

Selection (Nested efficiently)

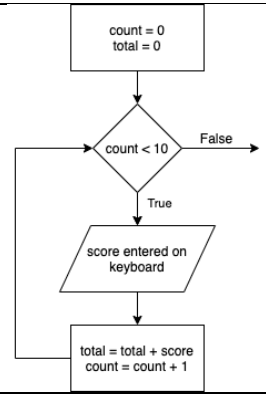


1. If mark is greater than 70 then
2. display you got an A on screen
3. else If mark is greater than 50 then
4. display you passed the test on screen
5. else
6. display you failed the test on screen
7. End if

```
heading = "Test Calculator";
if (mark > 70) {
    output = "You got an A in the test";
} else if (mark > 50){
    output = "Well done you passed";
} else {
    output = "You have failed";
}
Screen.display(output, heading);
```

s

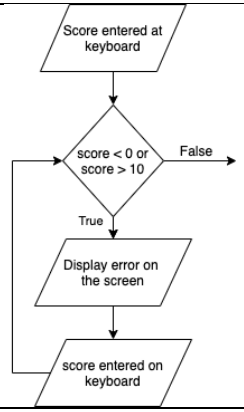
Fixed Loop (Bounded Repetition)



1. set integer total to 0
2. repeat 10 times
3. get integer score from user using keyboard
4. add score to total
5. end loop

```
int total = 0;
for (int index = 0; index < 10; index ++) {
   int score = Keyboard.getInt("Please enter the score");
   total = total + score;
}
```

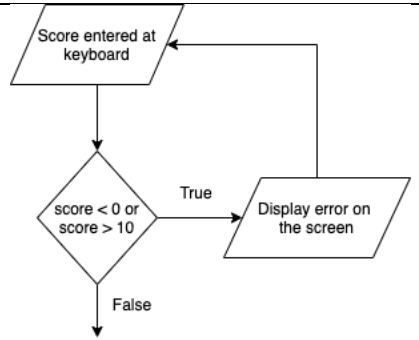
Conditional Loop (pre conditional) - check before loop starts



1. get real value from keyboard as score
2. while score < 0 or score > 10 loop
3. display error must be between 0 and 10
4. get real value from keyboard as score
5. end loop

```
double score = Keyboard.getReal("Please enter the score between 0 and 10");
while (score < 0 || score > 10) {
   Screen.display("Please enter a value between 0 and 10","Error");
   score = Keyboard.getReal("Please enter the score between 0 and 10");
}
```

Conditional Loop (post conditional) - check at end of loop



1. repeat
2. get real value from keyboard as score
3. if score < 0 or score > 10 then
4. display error must be between 0 and 10
5. end if
6. until score >= 0 and score <= 10

Or

6. while score < 0 or score > 10

```
do{
   score = Keyboard.getReal("Please enter the score between 0 and 10");
   if  (score < 0 || score > 10){
      Screen.display("Please enter a value between 0 and 10","Error");
   }
} while (score < 0 || score > 10);
```

Random Numbers		
	Set int randomNumer to a random value between 0 and 10	`int random = N5.randomInt(10);` smallest number 0 and max number 10
Rounding Numbers		
	Round averageAge to two decimal places Round averageAge to an integer	`averageAge = N5.roundDp(averageAge, 2);` `int averageAgeInt = N5.roundToInt(averageAge);` used to get rid of the .0 after a number if required
Length of String		
	1. If the length of name is more than 10 characters 2. Display wow that is a logn name 3. end if	`if (name.length() > 10) {` `  Screen.display("That is a long name","WOW");` `}`
Working with arrays (keyboard input) - traverse 1d array standard algorithm		
	1. Create an empty array of numberOfPeople strings called names 2. Repeat numberOfPeople times 3. get value for current index of array names from keyboard 4. end loop	`String [] names = new String [numberOfPeople];` `for (int index = 0; index < numberOfPeople; index++) {` `  names[index] = Keyboard.getText("please enter the persons name");` `}`

Working with arrays (output values) – traverse 1d array standard algorithm		
	1. Repeat numberOfPeople times 2. display name[index] on screen each on new line 3. end loop	``` String output = ""; String heading = "Array example"; for (int index = 0; index < numberOfPeople; index++) { output = output + names[index] + "\n"; } Screen.display(output, heading); ```