Testing the tool on Standard Library Test Files:

```
repeat until testNumber = 10
   if testNumber mod 2 = 0
      output testNumber + " is even"
   else
      output testNumber + " is odd"
   end
   testNumber = testNumber + 1
end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:2
Action identified is:Not identified
```

```
   action Main
      integer i = 1
      boolean firstCase = true
      repeat 4 times
         if firstCase
            output square(square(i))
         else
            output cube(cube(i))
         end
         i = i + 1
      end
   end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:2

Action identified is:Not identified
```

```
class Main
   action Main
      integer i = 1
      boolean firstCase = false
      repeat 4 times
         if firstCase
            output square(square(i))
         else
            output cube(cube(i))
            end
            i = i + 1
        end
    end

    action square(integer i) returns integer
        return i * i
    end

    action cube(integer i) returns integer
        return i * i * i
    end
end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:2

Action identified is:Not identified
```

```
class Main
    action Main
        Array<integer> array
        Random random
        random:SetSeed(12.0)
        repeat 9 times
            rand = random:RandomInteger(10000)
            array:Add(rand)
        end

        array:Sort()
        Iterator<integer> after = array:GetIterator()
        boolean isSorted = true
        integer first = -1
        result = ""
        repeat while after:HasNext()
            out = after:Next()
            if out >= first
                first = out
            else
                isSorted = false
            end
            result = result + out + ","
        end

        if isSorted
            output "sorted"
        else
            output "not sorted"
        end
    end
```

F1 is:1
F2 is:0
F3 is:0
F4 is:2
F5 is:0
F6 is:0
F7 is:1
F8 is:1

Action identified is:**Max/min**

```
class Main
    action Main
        Array<number> array
        Random random
        random:SetSeed(12.0)
        repeat 9 times
            rand = random:RandomNumber()
            array:Add(rand)
        end

        array:Sort()
        Iterator<number> after = array:GetIterator()
        boolean isSorted = true
        number first = -1
        result = ""
        repeat while after:HasNext()
            out = after:Next()
            if out >= first
                first = out
            else
                isSorted = false
            end
            result = result + out + ","
        end

        if isSorted
            output "sorted"
        else
            output "not sorted"
        end
    end
end
```

F1 is:1
F2 is:0
F3 is:0
F4 is:2
F5 is:0
F6 is:0
F7 is:1
F8 is:1

Action identified is:**Max/min**

```
class Main
    action Main
        Array<text> array
        array:Add("andy")
        array:Add("melissa")
        array:Add("jeff")
        array:Add("bill nye the science guy")

        array:Sort()
        Iterator<text> after = array:GetIterator()
        boolean isSorted = true
        text first = "a"
        result = ""
        repeat while after:HasNext()
            out = after:Next()
            CompareResult res = out:Compare(first)
            if res:result = res:LARGER or res:result = res:EQUAL
                first = out
            else
                isSorted = false
            end
            result = result + out + ","
        end

        if isSorted
            output "sorted"
        else
            output "not sorted"
        end
    end
end
```

F1 is:1
F2 is:0
F3 is:0
F4 is:2
F5 is:0
F6 is:0
F7 is:0
F8 is:2

Action identified is:Not identified

```
class Main
    action Main
        integer k = 0
        repeat 4 times
            if k > 2
                check
                    number f = cast(number, "nope")
                detect e
                    output "detect"
            end
         end
         k = k + 1
      end
   end
end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:1

Action identified is:Not identified
```

```
class Main
   action Main
      integer k = 0
      repeat until k = 4
         if k > 2
            check
               number f = cast(number, "nope")
            detect e
               output "detect"
            end
         end
         k = k + 1
      end
   end
end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:1

Action identified is:Not identified
```

```
class Main
   action Main
      integer k = 0
      repeat while k < 4
         if k > 2
            check
               number f = cast(number, "nope")
            detect e
               output "detect"
```

```
                end
            end
            k = k + 1
        end
    end
end
```

```
F1 is:0
F2 is:0
F3 is:0
F4 is:0
F5 is:0
F6 is:0
F7 is:0
F8 is:1

Action identified is:Not identified
```

```
class Main
    action Main
        integer a = 1
        repeat a times
            check
                integer i = 0
                repeat while i < 100
                    if i = 5
                        alert("i is 5")
                    end
                    i = i + 1
                end
            detect e
                if a = 1
                    check
                        integer z = 0
                        repeat until z = 100
                            integer t = 0
                            repeat while t < 3
                                if t = 1
                                    alert("t is 1")
                                end
                                t = t + 1
                            end
                        end
                    detect f
                        output "ok 1"
                        check
                            integer t = 0
                            repeat 9 times
                                if t = 2
                                    alert("t = 2")
                                end
                                t = t + 1
                            end
                        detect g
                            output "ok 2"
```

```
                  always
                     output "ok 3"
                  end
               always
                  output "ok 4"
               end
            end
         always
            output "ok 5"
         end
      end
   end
end
```

Error

```
class Main
   action Main
      integer a = 2
      repeat a times
         check
            integer i = 0
            repeat while i < 100
               if i = 5
                  alert("i is 5")
               end
               i = i + 1
            end
         detect e
            if a = 1
               // none of this code will be executed
               check
                  integer z = 0
                  repeat until z = 100
                     integer t = 0
                     repeat while t < 3
                        if t = 1
                           alert("t is 1")
                        end
                        t = t + 1
                     end
                  end
               detect f
                  output "bad"
                  check
                     integer t = 0
                     repeat 9 times
                        if t = 2
                           alert("t = 2")
                        end
                        t = t + 1
                     end
                  detect g
                     output "bad"
                  always
                     output "bad"
                  end
               always
                  output "bad"
               end
            elseif a = 4
               // none of this code will be executed
               check
                  integer z = 0
                  repeat until z = 100
                     integer t = 0
                     repeat while t < 3
                        if t = 1
                           alert("t is 1")
                        end
                        t = t + 1
                     end
                  end
               detect f
                  output "bad"
                  check
                     integer t = 0
                     repeat 9 times
                        if t = 2
                           alert("t = 2")
                        end
                        t = t + 1
                     end
                  detect g
                     output "bad"
                  always
                     output "bad"
                  end
               always
                  output "bad"
               end
            else
               check
                  integer z = 0
                  repeat until z = 100
                     integer t = 0
                     repeat while t < 3
                        if t = 1
                           alert("t is 1")
                        end
                        t = t + 1
                     end
                  end
               detect f
                  output "ok 1"
                  check
                     integer t = 0
                     repeat 9 times
                        if t = 2
                           alert("t = 2")
                        end
                        t = t + 1
```

```
                     end
                  detect g
                     output "ok 2"
                  end
               always
                  output "ok 3"
               end
            end
         end
      end
   end
end
```

```
Error
```

```
      repeat while after:HasNext()
         out = after:Next()
         if out >= first
            first = out
         else
            isSorted = false
         end
         result = result + out + ","
      end
```

```
F1 is:1
F2 is:0
F3 is:0
F4 is:2
F5 is:0
F6 is:0
F7 is:1
F8 is:1
Action identified is:Max/min
```

Report:

Out of identified 13 loop-ifs, actions were identified for **3** loop-ifs- all three were **MAX/MIN.**

For rest of the 10 loop-ifs, no actions were identified.