

Lab #5

Contents

1	While loops	1
1.1	Infinite loops	2
1.2	break statement	2
1.3	continue statement	4
2	Task 9	5
3	Task 10	5

1 While loops

While loop repeats code while the condition is true.

The basic syntax is:

```
1 while condition
2     code
3 end
```

For example:

```
1 clear;
2
3 n = 5;
4
5 while n > 1
6     n = n-1;
7     disp(n);
8 end
```

1.1 Infinite loops

Be careful, an infinite loop (it is a loop which never ends on its own) is possible with `while`, for example

```
1 % This loop will be running forever
2 clear;
3
4 n = 5;
5
6 while n > 1 % initially n > 1
7     n = n+1; % and we increasing n each iteration,
8     disp(n); % hence the loop will never end
9 end
```

1.2 break statement

`break` statement stops the loop immediately. No further iterations will be done. This statement works with both `while` and `for` loops.

In the following example we use `break` to exit the infinite cycle.

```
1 % This loop will be running forever
2 clear;
3
4 n = 5;
5 while n > 1
6     n = n+1;
7
8     if n > 100 % when n > 100
9         break % we stop the loop
10    end
11    disp(n);
12 end
```

The following code will stop printing after 3 because the loop is terminated when `a == 4`.

```
1 clear;
2
3 for a=1:5
4     if a == 4
```

```

5         break
6     end
7
8     disp(a);
9 end

```

In the following example the loop stops when the user chooses 0 as the input.

```

1 clear;
2 secret = 3;
3 guess = 0;
4
5 while guess ~= secret
6     guess = input('Guess my secret number between 1
7                 and 10 (to exit enter 0 ):');
8
9     if guess == 0
10         disp('You chose to exit. ');
11         break
12     end
13
14     if guess == secret
15         disp('Correct!');
16     else
17         disp('Try again >>');
18     end
19 end

```

Also this program can be implemented using infinite loops:

```

1 clear;
2 secret = 3;
3 guess = 0;
4
5 while 1 == 1 %force the loop to be infinite
6     guess = input('Guess my secret number between 1
7                 and 10 (to exit enter 0 ):');
8
9     if guess == 0
10         disp('You chose to exit. ');
11         break

```

```

11     end
12
13     if guess == secret
14         disp('Correct!');
15         break
16     else
17         disp('Try again >>');
18     end
19 end

```

1.3 continue statement

`continue` statement allows to skip the rest part of the code in current iteration and to go to the next iteration of the loop. This statement works with both `while` and `for` loops.

In the following example "Hello, world!" text will not be printed, because `continue` is the first statement in the `for` loop.

```

1 clear;
2
3 for a=1:5
4     continue
5
6     disp('Hello, world!');
7 end

```

And the following code will print "Hello, world!" 5 times and "Good bye, world!" only 3 times because of the `continue` command before the second `disp` command.

```

1 clear;
2
3 for a=1:5
4     disp('Hello, world!');
5
6     if a > 3
7         continue
8     end
9
10    disp('Good bye, world!');
11 end

```

2 Task 9

Write a program which prints text "Hello, World!" 5 times using `while` loop.

3 Task 10

Implement Euclidean GCD algorithm from lecture slides on Matlab (https://docs.google.com/presentation/d/1JmBCX4HxZN_ew7aiV_9u22gqHRt9NbIBw4Ag9ke1k7o/edit#slide=id.p19)