Basis Path Testing

Notes:

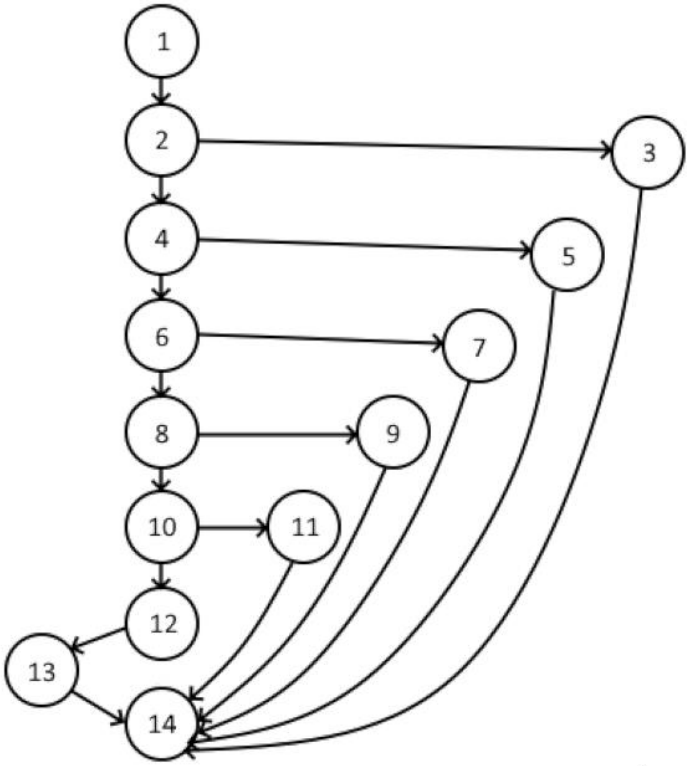
Our website currently does not contain any numerical calculations. Therefore, we have decided to complete our basis path testing using the flow of switch cases used for our password verification.

1: Algorithm and Flow Graph:

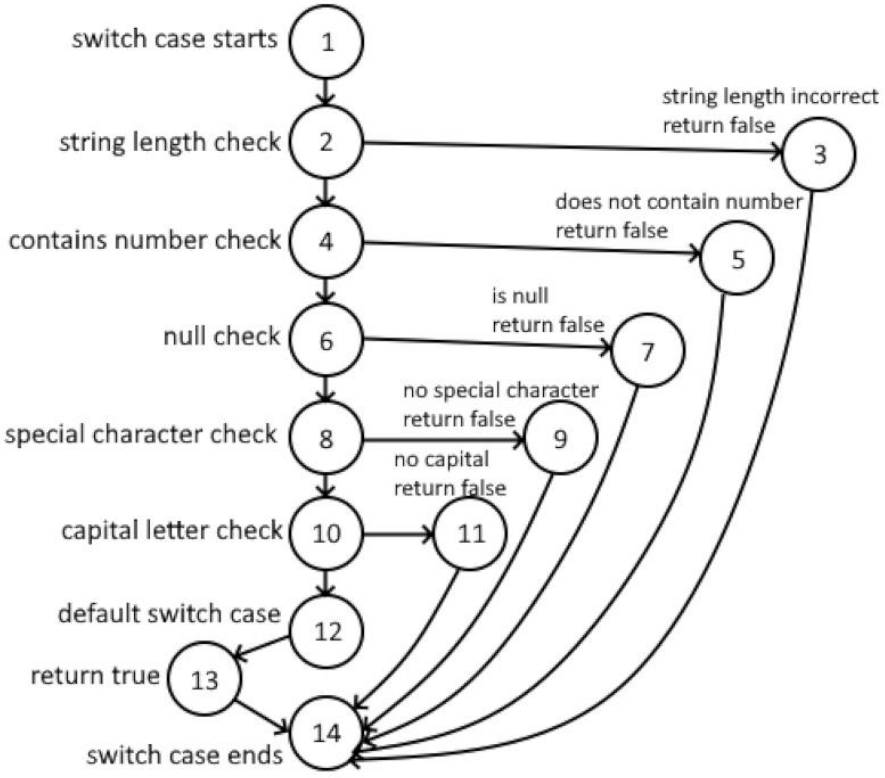
1.1: Algorithm:

```
1:     switch($password) {

2:           case (strlen($password) < 5 || strlen($password) > 30):

3:                 echo nl2br("Password must be at least 5-30 characters long." .
           "\n");

                   echo nl2br("\n");

                   return false;

4:           case (!preg_match($numReg, $password)):

5:                 echo nl2br("Password must contain at least 1 number.". "\n");

                   echo nl2br("\n");

                   return false;

6:           case (null):

7:                 echo nl2br("Password cannot be null." . "\n");

                   echo nl2br("\n");

                   return false;

8:           case (!preg_match($symbolReg, $password)):

9:                 echo nl2br("Password must contain at least 1 special
           character.". "\n");

                   echo nl2br("\n");

                   return false;

10:          case (!preg_match($capitalReg, $password)):

11:                echo nl2br("Password must contain at least 1 capital
           character." . "\n");

                   echo nl2br("\n");

                   return false;

12:          default:

13:          return true;

14:          }
```

1.2: Simple Flow Graph:



1.3: Detailed Flow Graph:



2: Cyclomatic Complexity:

We can use a calculation to discover the number of individual paths that can be taken from the above graph. We will discover this using the calculation below:

Vertex = V.
Graph = G.
Edge = E.
Node = N.

V(G) = E – N + 2
= 18 – 14 + 2
= 6

There are 6 independent paths that can be taken through the above graph.

3: Independent Paths:

1: 1-2-4-6-8-10-12-13-14
2: 1-2-4-6-8-10-11-14
3: 1-2-4-6-8-9-14
4: 1-2-4-6-7-14
5: 1-2-4-5-14
6: 1-2-3-14

4: Test Cases:

Path:	Input:	Expected Outcome:	Outcome:
1	Password123!	true	true
2	password123!	false	false
3	Password123	false	false
4		false	false
5	Password!	false	false
6	Pw1!	false	false