```
/Users/calving/Desktop/venv/bin/python /Users/calving/Desktop/feature_selection/main.py
Welcome to Calvin Ng 's Feature Selection Algorithm
Type in the name of the file to test : cs_170_small43.txt

Type the number of the algorithm you want to run.

1)Forward Selection
2)Backward Elimination
            2
This dataset has 10 features (not including the class attribute), with 100 instances.

Please wait while I normalize the data… Done!

Running nearest neighbor with all 10 features, using “leaving-one-out” evaluation, I get an accuracy of 73.0%

Beginning Search.

     Using feature(s)[2, 3, 4, 5, 6, 7, 8, 9, 10] accuracy is 71.0
     Using feature(s)[1, 3, 4, 5, 6, 7, 8, 9, 10] accuracy is 76.0
     Using feature(s)[1, 2, 4, 5, 6, 7, 8, 9, 10] accuracy is 73.0
     Using feature(s)[1, 2, 3, 5, 6, 7, 8, 9, 10] accuracy is 73.0
     Using feature(s)[1, 2, 3, 4, 6, 7, 8, 9, 10] accuracy is 74.0
     Using feature(s)[1, 2, 3, 4, 5, 7, 8, 9, 10] accuracy is 69.0
     Using feature(s)[1, 2, 3, 4, 5, 6, 8, 9, 10] accuracy is 73.0
     Using feature(s)[1, 2, 3, 4, 5, 6, 7, 9, 10] accuracy is 80.0
     Using feature(s)[1, 2, 3, 4, 5, 6, 7, 8, 10] accuracy is 69.0
     Using feature(s)[1, 2, 3, 4, 5, 6, 7, 8, 9] accuracy is 72.0

Feature(s) set[1, 2, 3, 4, 5, 6, 7, 9, 10] was best, accuracy is 80.0

     Using feature(s)[2, 3, 4, 5, 6, 7, 9, 10] accuracy is 81.0
     Using feature(s)[1, 3, 4, 5, 6, 7, 9, 10] accuracy is 80.0
     Using feature(s)[1, 2, 4, 5, 6, 7, 9, 10] accuracy is 77.0
```

```
34      Using feature(s)[1, 2, 3, 5, 6, 7, 9, 10] accuracy is
   75.0
35      Using feature(s)[1, 2, 3, 4, 6, 7, 9, 10] accuracy is
   79.0
36      Using feature(s)[1, 2, 3, 4, 5, 7, 9, 10] accuracy is
   72.0
37      Using feature(s)[1, 2, 3, 4, 5, 6, 9, 10] accuracy is
   79.0
38      Using feature(s)[1, 2, 3, 4, 5, 6, 7, 10] accuracy is
   71.0
39      Using feature(s)[1, 2, 3, 4, 5, 6, 7, 9] accuracy is
   79.0
40
41 Feature(s) set[2, 3, 4, 5, 6, 7, 9, 10] was best, accuracy
   is 81.0
42
43      Using feature(s)[3, 4, 5, 6, 7, 9, 10] accuracy is 83.
   0
44      Using feature(s)[2, 4, 5, 6, 7, 9, 10] accuracy is 76.
   0
45      Using feature(s)[2, 3, 5, 6, 7, 9, 10] accuracy is 76.
   0
46      Using feature(s)[2, 3, 4, 6, 7, 9, 10] accuracy is 77.
   0
47      Using feature(s)[2, 3, 4, 5, 7, 9, 10] accuracy is 71.
   0
48      Using feature(s)[2, 3, 4, 5, 6, 9, 10] accuracy is 78.
   0
49      Using feature(s)[2, 3, 4, 5, 6, 7, 10] accuracy is 70.
   0
50      Using feature(s)[2, 3, 4, 5, 6, 7, 9] accuracy is 82.0
51
52 Feature(s) set[3, 4, 5, 6, 7, 9, 10] was best, accuracy is
   83.0
53
54      Using feature(s)[4, 5, 6, 7, 9, 10] accuracy is 82.0
55      Using feature(s)[3, 5, 6, 7, 9, 10] accuracy is 82.0
56      Using feature(s)[3, 4, 6, 7, 9, 10] accuracy is 85.0
57      Using feature(s)[3, 4, 5, 7, 9, 10] accuracy is 74.0
58      Using feature(s)[3, 4, 5, 6, 9, 10] accuracy is 79.0
59      Using feature(s)[3, 4, 5, 6, 7, 10] accuracy is 70.0
60      Using feature(s)[3, 4, 5, 6, 7, 9] accuracy is 85.0
61
62 Feature(s) set[3, 4, 6, 7, 9, 10] was best, accuracy is 85.
   0
63
64      Using feature(s)[4, 6, 7, 9, 10] accuracy is 78.0
65      Using feature(s)[3, 6, 7, 9, 10] accuracy is 87.0
66      Using feature(s)[3, 4, 7, 9, 10] accuracy is 77.0
67      Using feature(s)[3, 4, 6, 9, 10] accuracy is 82.0
```

```
     Using feature(s)[3, 4, 6, 7, 10] accuracy is 70.0
     Using feature(s)[3, 4, 6, 7, 9] accuracy is 84.0

Feature(s) set[3, 6, 7, 9, 10] was best, accuracy is 87.0

     Using feature(s)[6, 7, 9, 10] accuracy is 78.0
     Using feature(s)[3, 7, 9, 10] accuracy is 79.0
     Using feature(s)[3, 6, 9, 10] accuracy is 81.0
     Using feature(s)[3, 6, 7, 10] accuracy is 66.0
     Using feature(s)[3, 6, 7, 9] accuracy is 83.0

(Warning, Accuracy has decreased! Continuing search in
case of local maxima)

Feature(s) set[3, 6, 7, 9] was best, accuracy is 87.0

     Using feature(s)[6, 7, 9] accuracy is 84.0
     Using feature(s)[3, 7, 9] accuracy is 82.0
     Using feature(s)[3, 6, 9] accuracy is 84.0
     Using feature(s)[3, 6, 7] accuracy is 69.0

Addition of features is not improving the model

Finished Search!! The best feature subset is [3, 6, 7, 9
], which has an accuracy of 87.0

Process finished with exit code 0

```