


**Assignment 3 – Combination Strategies**

**\*\* Using the one slack period for this homework**

Question 1:

The first and last 10 test frames:

```
tsl > sort.tsl.tsl
1
2 Test Case 1      (Key = 1.1.1.1.1.1.1.)
3   help          : on
4   version       : on
5   order         : numeric
6   check         : yes
7   unique        : yes
8   file_contents : sorted_alphabetically
9   duplicates    : yes
10
11
12 Test Case 2      (Key = 1.1.1.1.1.1.2.)
13   help          : on
14   version       : on
15   order         : numeric
16   check         : yes
17   unique        : yes
18   file_contents : sorted_alphabetically
19   duplicates    : no
20
21
22 Test Case 3      (Key = 1.1.1.1.1.2.1.)
23   help          : on
24   version       : on
25   order         : numeric
26   check         : yes
27   unique        : yes
28   file_contents : sorted_numerically
29   duplicates    : yes
30
```

tsl >  sort.tsl.tsl

```
31
32 Test Case 4      (Key = 1.1.1.1.1.2.2.)
33   help          : on
34   version       : on
35   order         : numeric
36   check         : yes
37   unique        : yes
38   file_contents : sorted_numerically
39   duplicates    : no
40
41
42 Test Case 5      (Key = 1.1.1.1.1.3.1.)
43   help          : on
44   version       : on
45   order         : numeric
46   check         : yes
47   unique        : yes
48   file_contents : unsorted
49   duplicates    : yes
50
51
52 Test Case 6      (Key = 1.1.1.1.1.3.2.)
53   help          : on
54   version       : on
55   order         : numeric
56   check         : yes
57   unique        : yes
58   file_contents : unsorted
59   duplicates    : no
```

```
62  ∨ Test Case 7      (Key = 1.1.1.1.2.1.1.)
63      help          : on
64      version        : on
65      order           : numeric
66      check           : yes
67      unique          : no
68      file_contents   : sorted_alphabetically
69      duplicates      : yes
70
71
72  ∨ Test Case 8      (Key = 1.1.1.1.2.1.2.)
73      help          : on
74      version        : on
75      order           : numeric
76      check           : yes
77      unique          : no
78      file_contents   : sorted_alphabetically
79      duplicates      : no
80
81
82  ∨ Test Case 9      (Key = 1.1.1.1.2.2.1.)
83      help          : on
84      version        : on
85      order           : numeric
86      check           : yes
87      unique          : no
88      file_contents   : sorted_numerically
89      duplicates      : yes
```

```
91
92  ∨ Test Case 10     (Key = 1.1.1.1.2.2.2.)
93      help          : on
94      version        : on
95      order           : numeric
96      check           : yes
97      unique          : no
98      file_contents   : sorted_numerically
99      duplicates      : no
100
```

```
4692 Test Case 470      (Key = 2.2.5.2.1.1.2.)
4693     help           : off
4694     version          : off
4695     order             : human
4696     check             : no
4697     unique            : yes
4698     file_contents    : sorted_alphabetically
4699     duplicates        : no
4700
4701
4702 Test Case 471      (Key = 2.2.5.2.1.2.1.)
4703     help           : off
4704     version          : off
4705     order             : human
4706     check             : no
4707     unique            : yes
4708     file_contents    : sorted_numerically
4709     duplicates        : yes
4710
4711
4712 Test Case 472      (Key = 2.2.5.2.1.2.2.)
4713     help           : off
4714     version          : off
4715     order             : human
4716     check             : no
4717     unique            : yes
4718     file_contents    : sorted_numerically
4719     duplicates        : no
```

```
4722 Test Case 473      (Key = 2.2.5.2.1.3.1.)
4723     help           : off
4724     version         : off
4725     order            : human
4726     check            : no
4727     unique           : yes
4728     file_contents    : unsorted
4729     duplicates       : yes
4730
4731
4732 Test Case 474      (Key = 2.2.5.2.1.3.2.)
4733     help           : off
4734     version         : off
4735     order            : human
4736     check            : no
4737     unique           : yes
4738     file_contents    : unsorted
4739     duplicates       : no
4740
4741
4742 Test Case 475      (Key = 2.2.5.2.2.1.1.)
4743     help           : off
4744     version         : off
4745     order            : human
4746     check            : no
4747     unique           : no
4748     file_contents    : sorted_alphabetically
4749     duplicates       : yes
4750
```

```
4752  ∨ Test Case 476      (Key = 2.2.5.2.2.1.2.)
4753      help           :  off
4754      version          :  off
4755      order             :  human
4756      check              :  no
4757      unique             :  no
4758      file_contents      :  sorted_alphabetically
4759      duplicates         :  no
4760
4761
4762  ∨ Test Case 477      (Key = 2.2.5.2.2.2.1.)
4763      help           :  off
4764      version          :  off
4765      order             :  human
4766      check              :  no
4767      unique             :  no
4768      file_contents      :  sorted_numerically
4769      duplicates         :  yes
4770
4771
4772  ∨ Test Case 478      (Key = 2.2.5.2.2.2.2.)
4773      help           :  off
4774      version          :  off
4775      order             :  human
4776      check              :  no
4777      unique             :  no
4778      file_contents      :  sorted_numerically
4779      duplicates         :  no
4780
```

```
Test Case 479      (Key = 2.2.5.2.2.3.1.)
  help           :  off
  version          :  off
  order             :  human
  check              :  no
  unique             :  no
  file_contents      :  unsorted
  duplicates         :  yes
```

```
Test Case 480      (Key = 2.2.5.2.2.3.2.)
  help           :  off
  version          :  off
  order             :  human
  check              :  no
  unique             :  no
  file_contents      :  unsorted
  duplicates         :  no
```

**Qa) Pick one test frame and explain (on paper) what this frame means (i.e., what combinations of options you would test based on the frame – note you will want to choose one that does not have the help and/or the version turned on for this question). Use the existing input files to pick a file that would make sense to use for this test frame. Give this frame as a concrete test case and show a screen shot of the output (you will need to show the command line, so we know which input you used)**

➔ Test frame selected:

Test Case 479 (Key = 2.2.5.2.2.3.1.)

help : off  
version : off  
order : human  
check : no  
unique : no  
file\_contents : unsorted  
duplicates : yes

For this test case, the following combinations of options would be tested:

- ‘help’: The tool will not provide help information
- ‘version’: The tool will not display its version information
- ‘order’: The tool will sort the input in human-readable order
- ‘check’: The tool will not check for numerical consistency
- ‘unique’: The tool will not check for uniqueness of elements
- ‘file\_contents’: The input file is sorted
- ‘duplicates’: The input file contains duplicate elements

According to these options, using the “random numbers” input file would make sense for this test, because it contains duplicate elements and unsorted numbers. This combination of options will leave the input file unsorted, not check for uniqueness of elements, and check for duplicates.

**Qb) Now run the tsl on the file called “sort.refined.tsl”. How does sort.refined.tsl modify the tests cases that used help and version. Explain the differences and list how many frames it has.**

➔ Sort.refined.tsl has 130 test frames.

```
[kumra@pyrite-n1 tsl]$ ./tsl sort.refined.tsl  
  
130 test frames generated and written to sort.refined.tsl.tsl  
[kumra@pyrite-n1 tsl]$
```

Here, sort.refined.tsl test cases solely focus on the sorting algorithm, without considering any additional program features such as the “help” and “version” flags. These test cases specify the order in which the sorting algorithm should sort the input data, and whether it should check for

duplicates or unique values, and the contents of the sorted output file. They focused on the sorting behavior when presented with different inputs, such as sorting numbers numerically or sorting text alphabetically. Hence it modified the test cases that previously used “help” and “version” by removing them entirely and replacing them with new test cases solely focusing on the sorting algorithm.

Whereas, for the sort.tsl test cases, the “help” and “version” flags were used to indicate whether the program should display the information about how to use the program or the version of the program. These flags did not affect the sorting algorithm in any way.

## Q2a) How many combinations (test cases) are there for each of the above files?

➔ For the **cit-sort-input.txt file**, there were 15 combinations (test cases) in total, with a unique combination of parameter values.

CIT_Tool > cit_sample.out					
1	t	k	v	TCount	
2	2	5	14	75	
3					
4	5	2	2	3	2
5					
6	0	1	2	3	4
7	5	6			
8	7	8			
9	9	10	11		
10	12	13			
11					
12	15				
13	0	5	8	10	12
14	2	5	7	11	13
15	1	5	8	9	12
16	4	6	8	10	12
17	0	6	7	9	13
18	2	6	8	9	12
19	1	6	7	10	13
20	3	5	7	11	12
21	3	6	7	9	13
22	4	5	7	11	12
23	4	5	8	9	13
24	0	6	7	11	12
25	2	5	7	10	13
26	3	6	8	10	13
27	1	5	8	11	13
28					



```

[kumra@pyrite-n1 CIT_Tool]$ ./cit_generate cit-sort-input.txt -F
first part of exp table
 1 0.367879 0.135335 0.0497871 0.0183156 0.00673795 0.00247875 0.000911882 0.000
335463 0.00012341 4.53999e-05 1.67017e-05 6.14421e-06 2.26033e-06 8.31529e-07 3.
05902e-07 1.12535e-07 4.13994e-08 1.523e-08 5.6028e-09 2.06115e-09 7.58256e-10 2
.78947e-10 1.02619e-10 3.77513e-11 1.38879e-11 5.10909e-12 1.87953e-12 6.9144e-1
3 2.54367e-13 9.35762e-14 3.44248e-14 1.26642e-14 4.65889e-15 1.71391e-15 6.3051
2e-16 2.31952e-16 8.53305e-17 3.13913e-17 1.15482e-17
  kcombo
  9 12 12 11 12
Looking for cover:
t   k   v   TCount
2  5 14 75

  5 2 2 3 2

  0 1 2 3 4
  5 6
  7 8
  9 10 11
 12 13

upperbound = 10025

current tabsize 5516

current tabsize 2765

current tabsize 1389

current tabsize 701

current tabsize 357

current tabsize 185

current tabsize 99

current tabsize 56

current tabsize 35

current tabsize 24

current tabsize 19

current tabsize 16

current tabsize 15

```

For the cit-sort-input-3way.txt file, there were 30 combinations (test cases) in total with a unique combination of parameter values.

CIT_Tool > cit_sample.out				
1	t	k	v	TCount
2	3	5	14	194
3				
4	5	2	2	3
5				
6	0	1	2	3
7	5	6		
8	7	8		
9	9	10	11	
10	12	13		
11				
12	30			
13	4	5	8	11
14	0	5	7	11
15	2	6	8	10
16	3	5	8	9
17	2	5	8	11
18	1	6	7	10
19	4	5	7	9
20	2	5	7	10
21	0	5	8	9
22	3	5	8	11
23	1	6	7	11
24	3	6	7	9
25	4	6	8	9
26	3	6	8	10
27	1	6	8	9
28	1	5	7	9
29	4	5	8	10
30	2	5	8	9
31	3	5	7	10
32	1	5	8	11
33	0	6	8	10
34	3	6	7	11
35	0	5	7	10
36	0	6	8	11
37	4	6	7	10

37	4	6	7	10	13
38	2	6	7	9	12
39	1	5	8	10	12
40	2	6	7	11	13
41	4	6	7	11	12
42	0	6	7	9	13
43					

```
[kumra@pprite-nl CIT_Tool]$ ./cit_generate cit-sort-input-3way.txt -F
first part of exp table
1 0.367879 0.135335 0.0497871 0.0183156 0.00673795 0.00247875 0.000911882 0.000335463 0.00012341 4.53999e-05 1.67017e-05 6.14421e-06 2.26033e-06 8.31529e-07 3.05902e-07 1.12535e-07 4.13994e-08 1.523e-08 5.6028e-09 2.06115e-09 7.58256e-10 2.78947e-10 1.02619e-10 3.77513e-11 1.38879e-11 5.10909e-12 1.87953e-12 6.9144e-13 2.54367e-13 9.35762e-14 3.44248e-14 1.26642e-14 4.65889e-15 1.71391e-15 6.30512e-16 2.31952e-16 8.53305e-17 3.13913e-17 1.15482e-17
Kroabo
30 51 51 42 51
Looking for cover:
t k v TCount
3 5 14 194

5 2 2 3 2

0 1 2 3 4
5 6
7 8
9 10 11
12 13

upperbound = 10125
current tabsize 3830
current tabsize 1929
current tabsize 979
current tabsize 504
current tabsize 266
current tabsize 147
current tabsize 88
current tabsize 58
current tabsize 43
current tabsize 36
current tabsize 32
current tabsize 30
```

Battery saver

Battery saver is on  
Consider plugging in your device.

Q2b) Provide the output from the first one (cit-sort-input.txt).

CIT_Tool > cit_sample.out				
1	t	k	v	TCount
2	2	5	14	75
3				
4	5	2	2	3 2
5				
6	0	1	2	3 4
7	5	6		
8	7	8		
9	9	10	11	
10	12	13		
11				
12	15			
13	0	5	8	10 12
14	2	5	7	11 13
15	1	5	8	9 12
16	4	6	8	10 12
17	0	6	7	9 13
18	2	6	8	9 12
19	1	6	7	10 13
20	3	5	7	11 12
21	3	6	7	9 13
22	4	5	7	11 12
23	4	5	8	9 13
24	0	6	7	11 12
25	2	5	7	10 13
26	3	6	8	10 13
27	1	5	8	11 13
28				


**Q2c) Map the test cases so that they use the choices from the TSL.** For instance, you might have the header.


**Q3a) How many test frames do you have in this file?**

➔ There are 192 test frames in this file.

```
[kumra@pyrite-n4 tsl]$ ./tsl firefox.tsl  
  
192 test frames generated and written to firefox.tsl.tsl  
  
[kumra@pyrite-n4 tsl]$
```

**3b) Copy the contents of the firefox.tsl to the document. Copy the first and last 5 tests (10 in total of the firefox.tsl.tsl file to this document.**

```
tsl >  firefox.tsl  
1 #tsl for firefox  
2 Parameters:  
3     default_search_engine:  
4     google.  
5     bing.  
6     amazon.com  
7     duckduckgo.  
8     ebay.  
9     wikipedia.  
10 open_new_tabs_in_background:  
11     yes.  
12     no.  
13 show_tab_previews:  
14     yes.  
15     no.  
16 warn_on-closing_multiple_tabs:  
17     yes.  
18     no.  
19 update_options:  
20     automatic.  
21     manual.  
22 allow_search_updates:  
23     yes.  
24     no.  
25
```

tsl >  firefox.tsl.tsl

```
1
2 Test Case 1      (Key = 1.1.1.1.1.1.)
3   default_search_engine      : google
4   open_new_tabs_in_background : yes
5   show_tab_previews          : yes
6   warn_on-closing_multiple_tabs : yes
7   update_options              : automatic
8   allow_search_updates        : yes
9
10
11 Test Case 2      (Key = 1.1.1.1.1.2.)
12   default_search_engine      : google
13   open_new_tabs_in_background : yes
14   show_tab_previews          : yes
15   warn_on-closing_multiple_tabs : yes
16   update_options              : automatic
17   allow_search_updates        : no
18
19
20 Test Case 3      (Key = 1.1.1.1.2.1.)
21   default_search_engine      : google
22   open_new_tabs_in_background : yes
23   show_tab_previews          : yes
24   warn_on-closing_multiple_tabs : yes
25   update_options              : manual
26   allow_search_updates        : yes
27
28
29 Test Case 4      (Key = 1.1.1.1.2.2.)
30   default_search_engine      : google
31   open_new_tabs_in_background : yes
32   show_tab_previews          : yes
33   warn_on-closing_multiple_tabs : yes
34   update_options              : manual
35   allow_search_updates        : no
36
```

```
37
38 Test Case 5      (Key = 1.1.1.2.1.1.)
39   default_search_engine      : google
40   open_new_tabs_in_background : yes
41   show_tab_previews          : yes
42   warn_on-closing_multiple_tabs : no
43   update_options              : automatic
44   allow_search_updates        : yes
45
```

```
1685 Test Case 188      (Key = 6.2.2.1.2.2.)
1686     default_search_engine      : wikipedia
1687     open_new_tabs_in_background : no
1688     show_tab_previews           : no
1689     warn_on-closing_multiple_tabs : yes
1690     update_options              : manual
1691     allow_search_updates        : no
1692
1693
1694 Test Case 189      (Key = 6.2.2.2.1.1.)
1695     default_search_engine      : wikipedia
1696     open_new_tabs_in_background : no
1697     show_tab_previews           : no
1698     warn_on-closing_multiple_tabs : no
1699     update_options              : automatic
1700     allow_search_updates        : yes
1701
1702
1703 Test Case 190      (Key = 6.2.2.2.1.2.)
1704     default_search_engine      : wikipedia
1705     open_new_tabs_in_background : no
1706     show_tab_previews           : no
1707     warn_on-closing_multiple_tabs : no
1708     update_options              : automatic
1709     allow_search_updates        : no
1710
1711
1712 Test Case 191      (Key = 6.2.2.2.2.1.)
1713     default_search_engine      : wikipedia
1714     open_new_tabs_in_background : no
1715     show_tab_previews           : no
1716     warn_on-closing_multiple_tabs : no
1717     update_options              : manual
1718     allow_search_updates        : yes
1719
```

```
1720
1721 Test Case 192      (Key = 6.2.2.2.2.2.)
1722     default_search_engine      : wikipedia
1723     open_new_tabs_in_background : no
1724     show_tab_previews           : no
1725     warn_on-closing_multiple_tabs : no
1726     update_options              : manual
1727     allow_search_updates        : no
1728
```