

**ECE 322**  
**SOFTWARE TESTING AND MAINTENANCE**  
**Fall 2021**

---

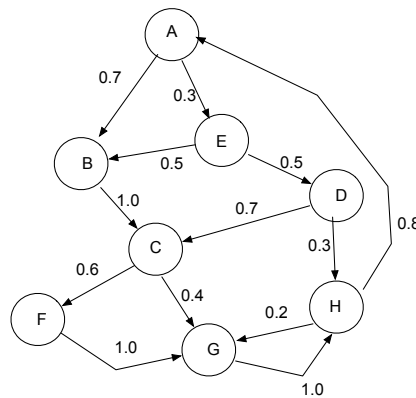
**Assignment #4**

**Due date: Monday, November 1, 2021 by 11:00 PM**

Total: 40 points

*10 points*

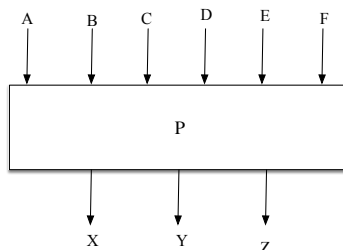
1. The behavior of a communication software system can be represented in the form of the finite state machine shown below. The figure shows the corresponding transition probabilities between the states A, B,....



Determine an order in which you would like to test individual states.

*10 points*

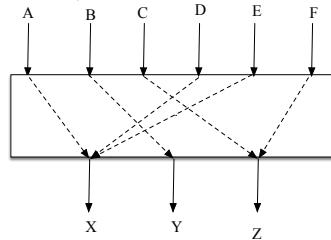
2. Consider the program P with 6 inputs A, B, C, D, E, F and 3 outputs X, Y, and Z. The input variables assume some values shown in the table below.



input	values
A	0, 1, 2, 3,4

B	A, B, C, D, E
C	100, 200, 300
D	7, 8, 9, 10, 11
E	Y, N
F	$\alpha, \beta, \gamma$

Consider the input-output testing strategy. Consider the situation when (i) you do not know functional dependencies among the input and output variables, and (ii) where these dependencies are known (as shown in the figure below).



What is the number of test cases required when using these two strategies. For (ii) propose a collection of test cases. In this strategy, how could you minimize the number of test cases; show an example.

*10 points*

3. For the configuration problem shown below, develop test cases using the strategy of pairwise testing. Compare the number of test cases obtained here with the number of tests required to consider all combinations of input values.

*Hint:* you can use the python package discussed in the lecture.

Parameter Name	Values	# Values
HARDKEYBOARDHIDDEN	NO, UNDEFINED, YES	3
KEYBOARDHIDDEN	NO, UNDEFINED, YES	3
KEYBOARD	12KEY, NOKEYS, QWERTY, UNDEFINED	4
NAVIGATIONHIDDEN	NO, UNDEFINED, YES	3
NAVIGATION	DPAD, NONAV, TRACKBALL, UNDEFINED, WHEEL	5
ORIENTATION	LANDSCAPE, PORTRAIT, SQUARE, UNDEFINED	4
SCREENLAYOUT_LONG	MASK, NO, UNDEFINED, YES	4
SCREENLAYOUT_SIZE	LARGE, MASK, NORMAL, SMALL, UNDEFINED	5
TOUCHSCREEN	FINGER, NOTOUCH, STYLUS, UNDEFINED	4

*10 points*

4. Propose test cases to carry out syntax-based testing for the following grammar

```

<expr> ::= <id> | <num> | <expr> <op> <expr>
<id> ::= <letter> | <letter> <id>
<num> ::= <digit> | <digit> <num>
<op> ::= * | / | +
  
```

$\langle \text{letter} \rangle ::= a|b|c|\dots|z|A|B|C|\dots|Z$   
 $\langle \text{digit} \rangle ::= 0|1|2|\dots|9$

so that the following coverage criteria are satisfied: (i) terminal symbol coverage, (ii) production coverage, and (iii) derivation coverage

How many test cases are required to meet the above coverage?