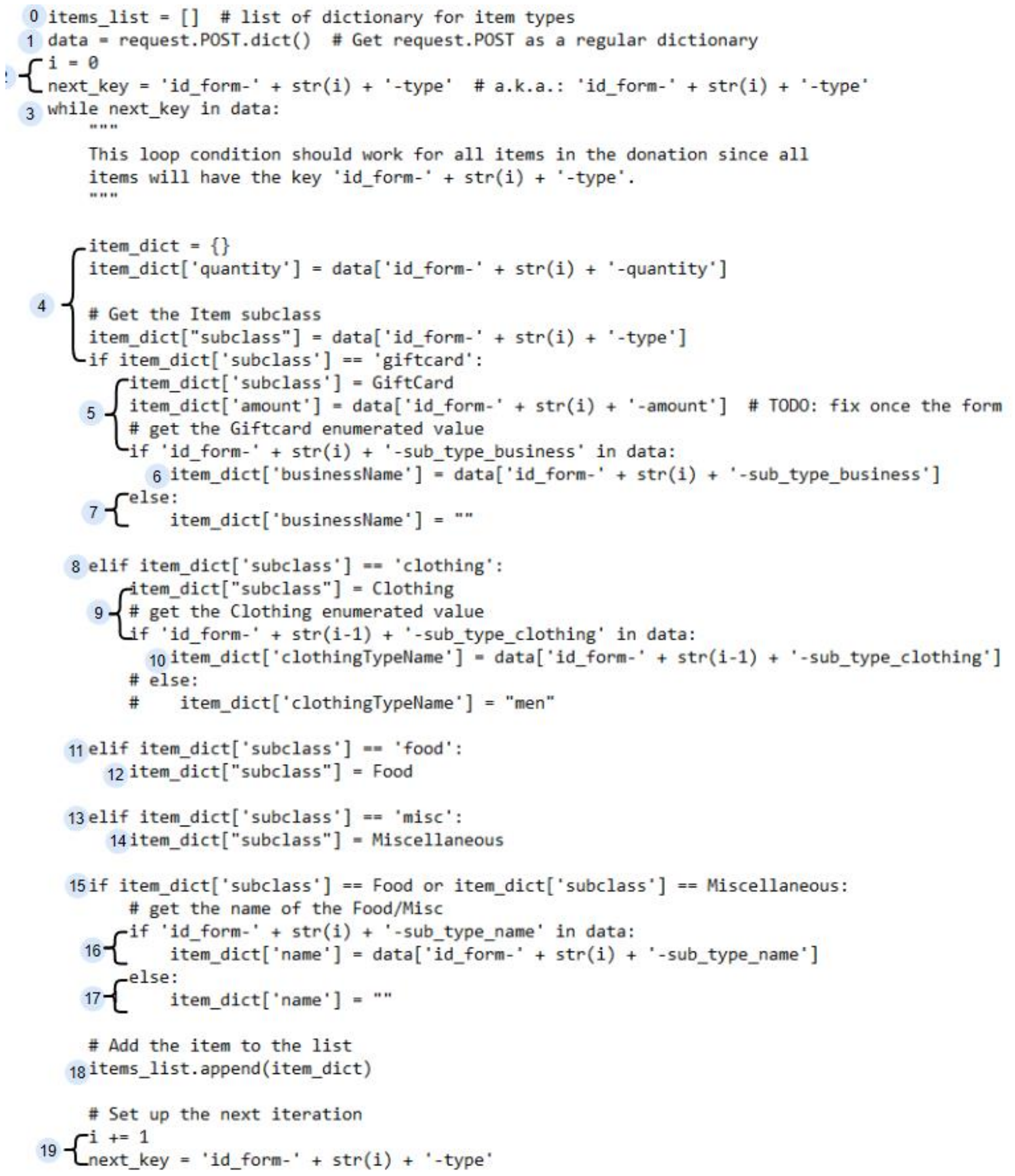
More Software Testing Fun & Security Eng.

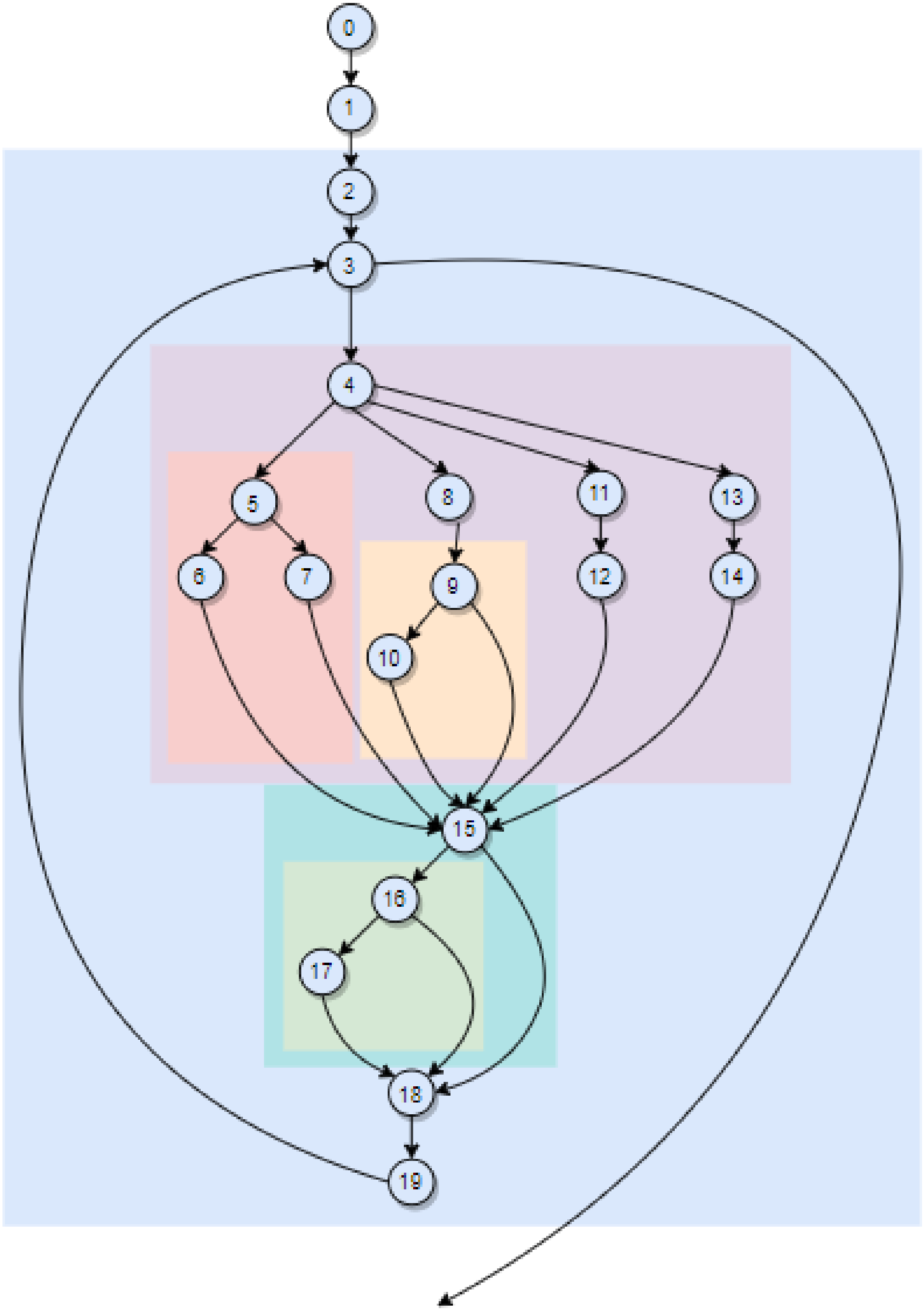
Jamal Madni

CECS 445

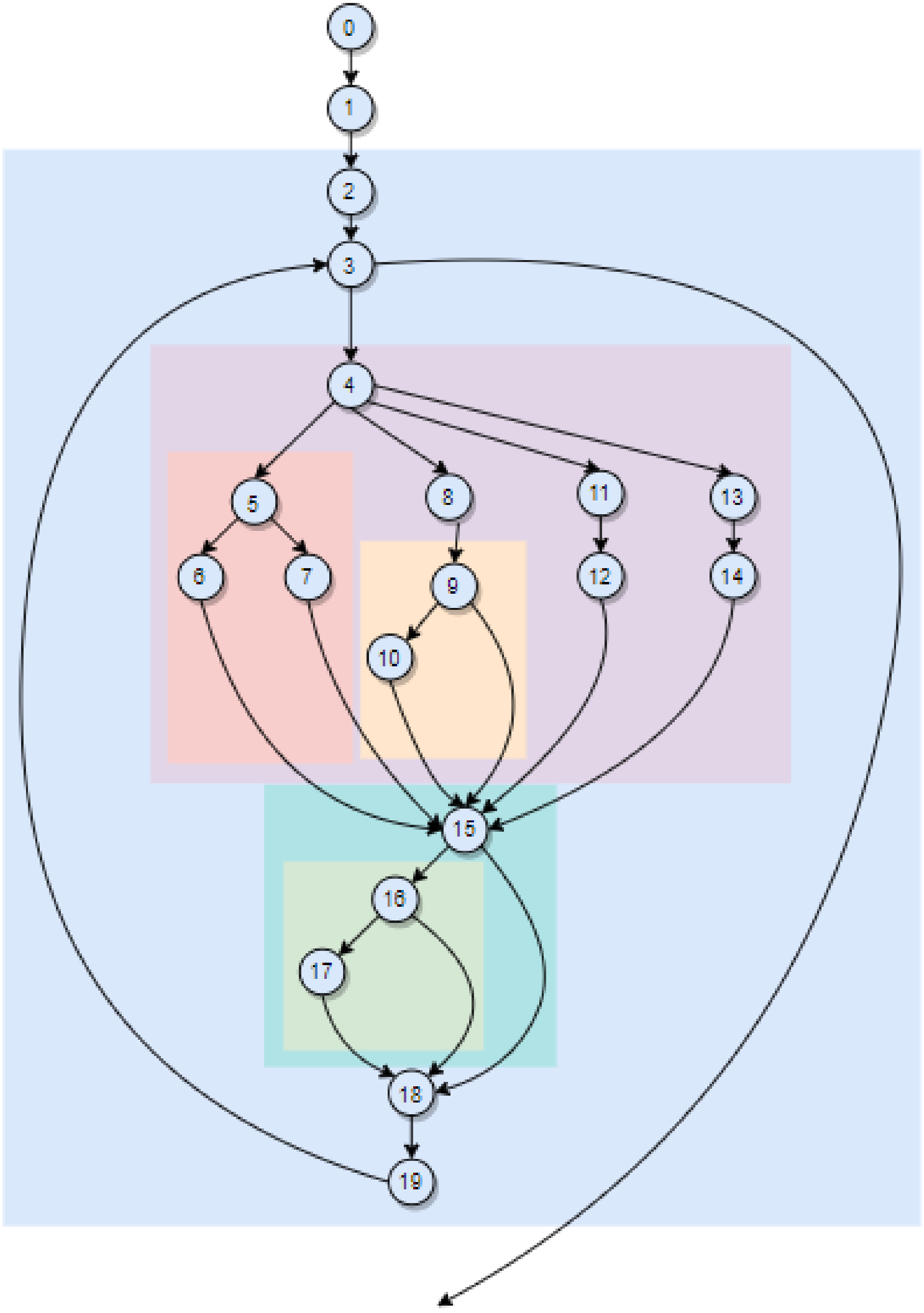
Lecture 13: March 23rd, 2021



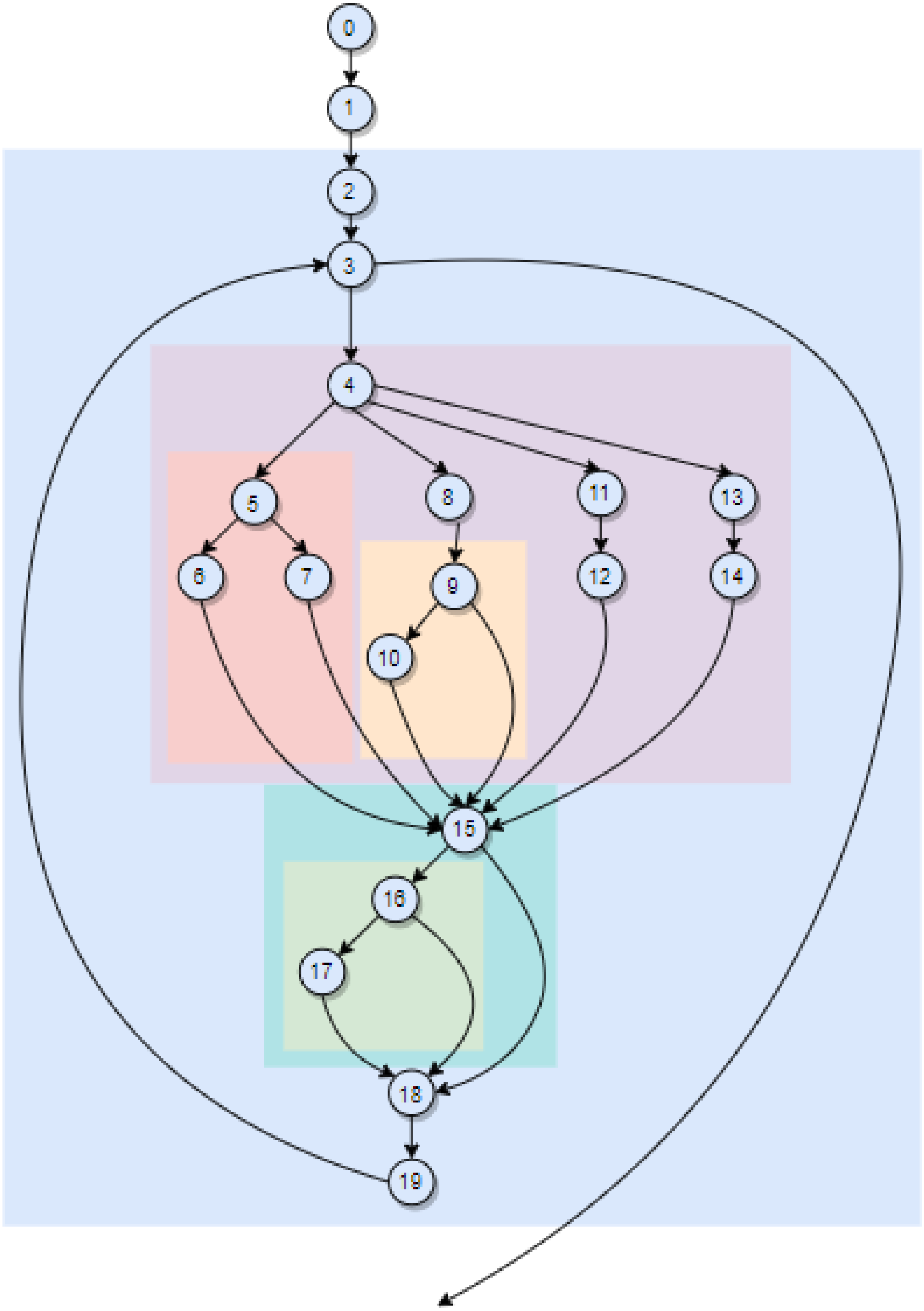
```
0  items_list = []  # list of dictionary for item types
1  data = request.POST.dict()  # Get request.POST as a regular dictionary
2  i = 0
   next_key = 'id_form-' + str(i) + '-type'  # a.k.a.: 'id_form-' + str(i) + '-type'
3  while next_key in data:
       """
       This loop condition should work for all items in the donation since all
       items will have the key 'id_form-' + str(i) + '-type'.
       """

4      item_dict = {}
       item_dict['quantity'] = data['id_form-' + str(i) + '-quantity']

       # Get the Item subclass
       item_dict["subclass"] = data['id_form-' + str(i) + '-type']
       if item_dict['subclass'] == 'giftcard':
5          item_dict['subclass'] = GiftCard
           item_dict['amount'] = data['id_form-' + str(i) + '-amount']  # TODO: fix once the form
           # get the Giftcard enumerated value
           if 'id_form-' + str(i) + '-sub_type_business' in data:
6              item_dict['businessName'] = data['id_form-' + str(i) + '-sub_type_business']
7          else:
               item_dict['businessName'] = ""

8      elif item_dict['subclass'] == 'clothing':
9          item_dict["subclass"] = Clothing
           # get the Clothing enumerated value
           if 'id_form-' + str(i-1) + '-sub_type_clothing' in data:
10             item_dict['clothingTypeName'] = data['id_form-' + str(i-1) + '-sub_type_clothing']
           # else:
           #    item_dict['clothingTypeName'] = "men"

11     elif item_dict['subclass'] == 'food':
12         item_dict["subclass"] = Food

13     elif item_dict['subclass'] == 'misc':
14         item_dict["subclass"] = Miscellaneous

15     if item_dict['subclass'] == Food or item_dict['subclass'] == Miscellaneous:
           # get the name of the Food/Misc
16         if 'id_form-' + str(i) + '-sub_type_name' in data:
               item_dict['name'] = data['id_form-' + str(i) + '-sub_type_name']
17         else:
               item_dict['name'] = ""

       # Add the item to the list
18     items_list.append(item_dict)

       # Set up the next iteration
19     i += 1
       next_key = 'id_form-' + str(i) + '-type'
```



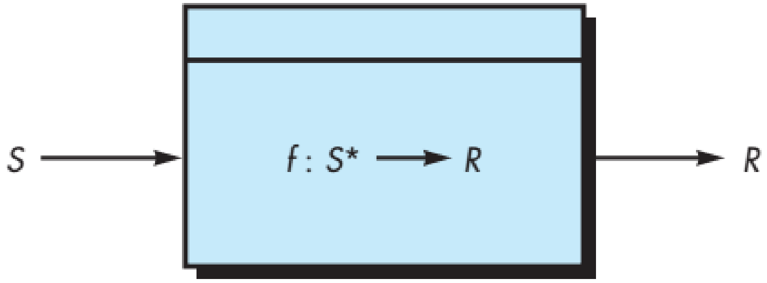
- V(G) = E – N + 2 = 28 - 19 + 2 = 11
- Linearly Independent Paths = {F} + {M} + {E}
- Test-Cases = *Each of 11 LI Paths tested with any other piece*
- Front Paths (F):
    - 0-1-2-3-20
    - 0-1-2-3-4
- Middle Paths (M):
    - 4-5-6-15
    - 4-5-7-15
    - 4-8-9-10-15
    - 4-8-9-15
    - 4-11-12-15
    - 4-13-14-15
- End Paths (E):
    - 15-16-17-18-19-3
    - 15-16-18-19-3
    - 15-18-19-3



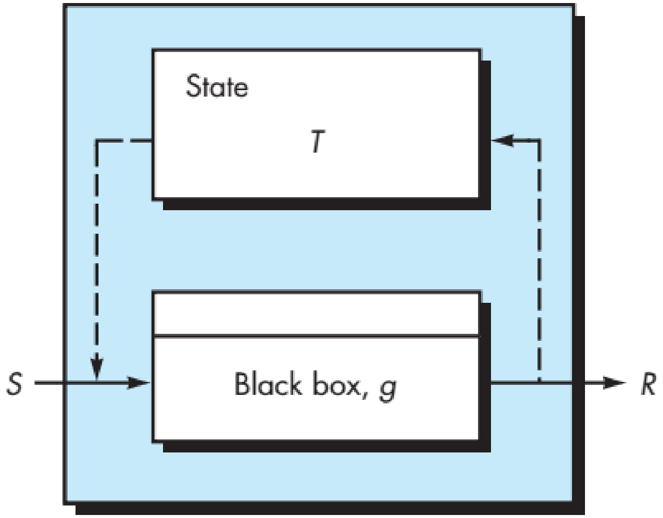
Test Case	Parameter Values
F1	next_key = X X not in data
F2	…
M1	
M2	
M3	
M4	
M5	
M6	
E1	
E2	
E3	



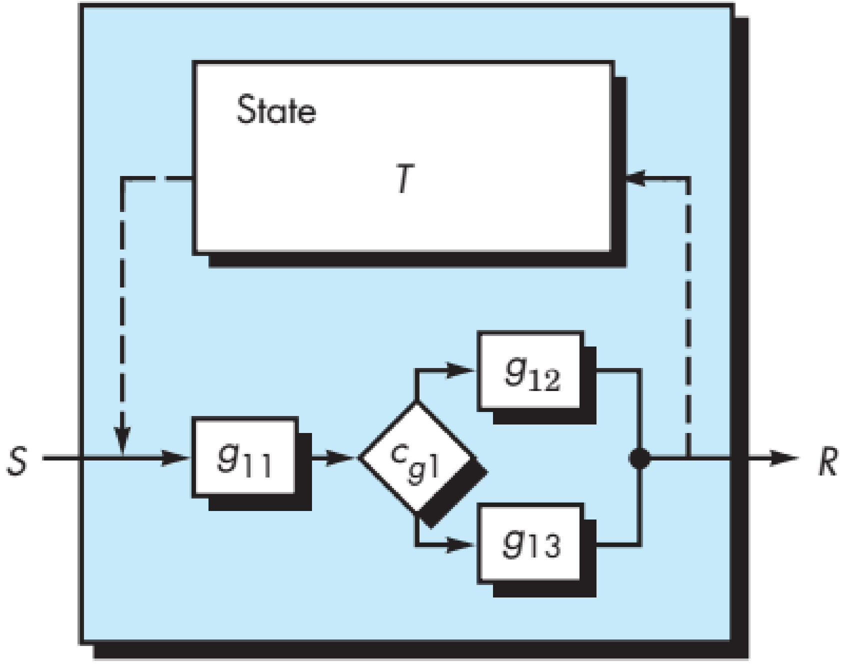
Black Box vs. State Box vs. Clear Box



Black Box

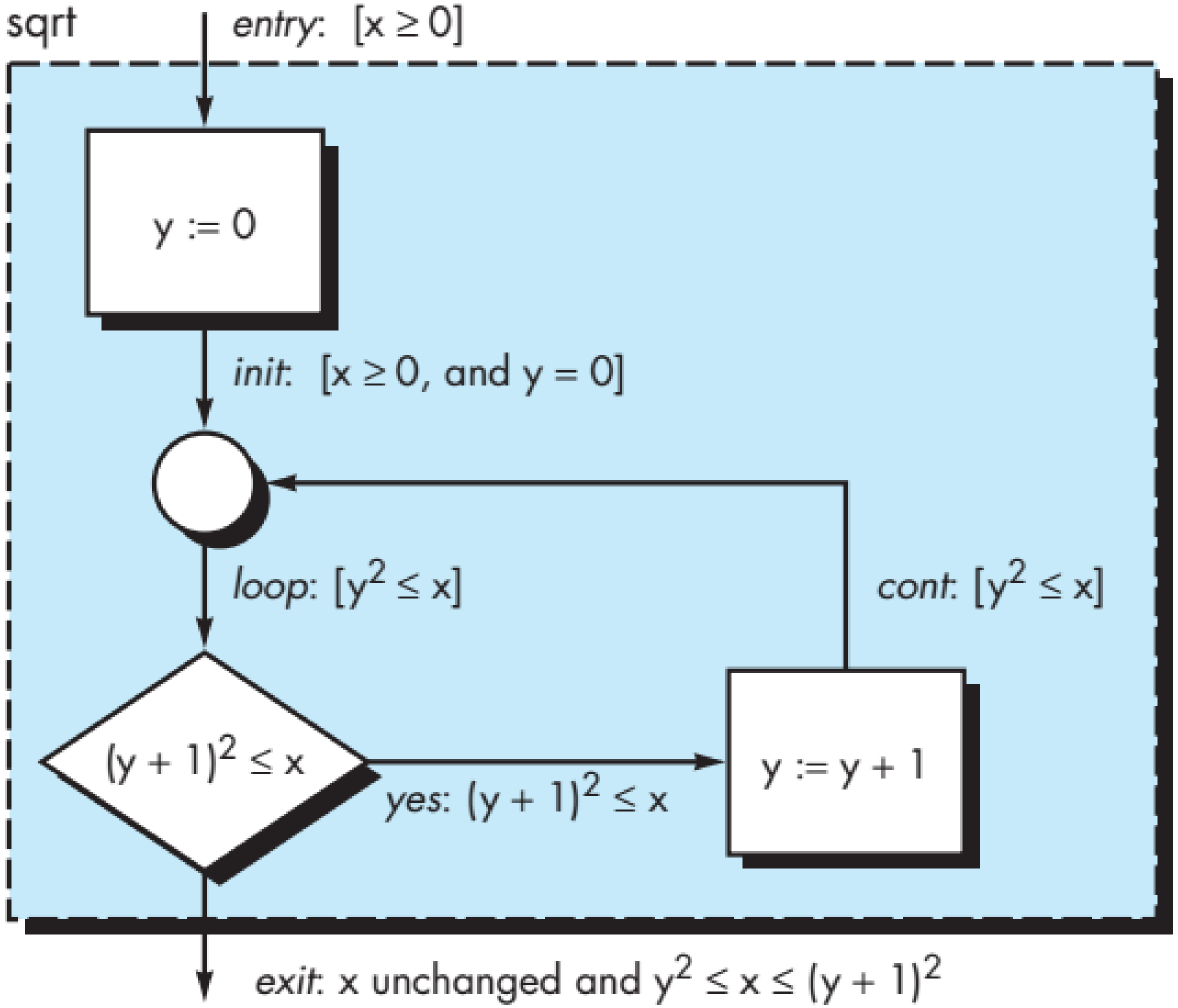


State Box



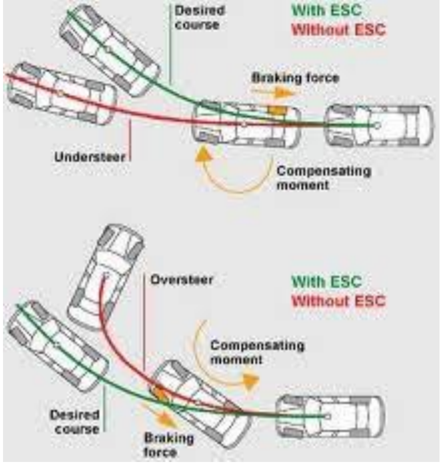
Clear Box

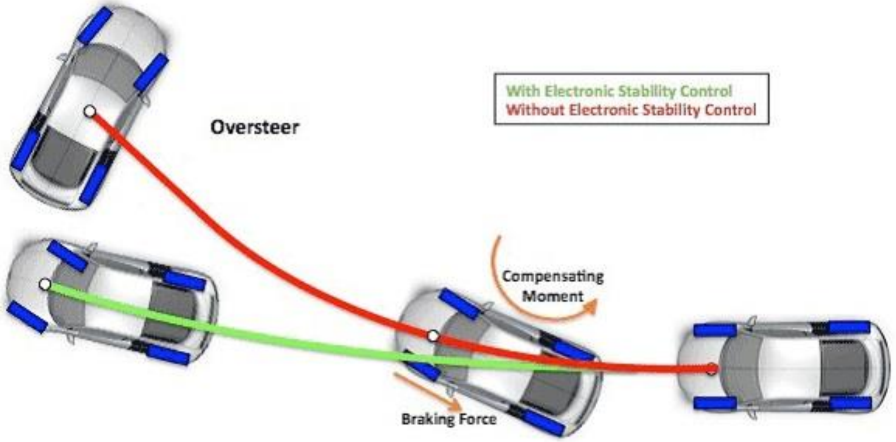
Black Box vs. State Box vs. Clear Box Example

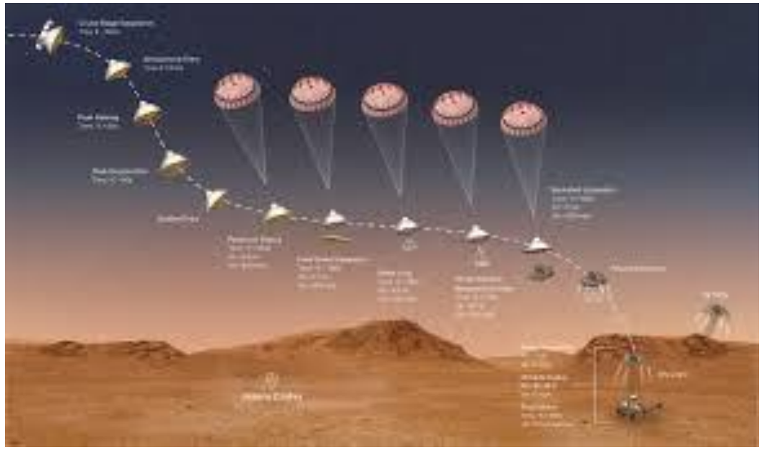

sqrt
entry: $[x \geq 0]$
$y := 0$
init: $[x \geq 0$, and $y = 0]$
loop: $[y^2 \leq x]$
$(y + 1)^2 \leq x$
yes: $(y + 1)^2 \leq x$
$y := y + 1$
cont: $[y^2 \leq x]$
exit: x unchanged and $y^2 \leq x \leq (y + 1)^2$

Security Engineering

- Dependability (operates under hostile conditions)

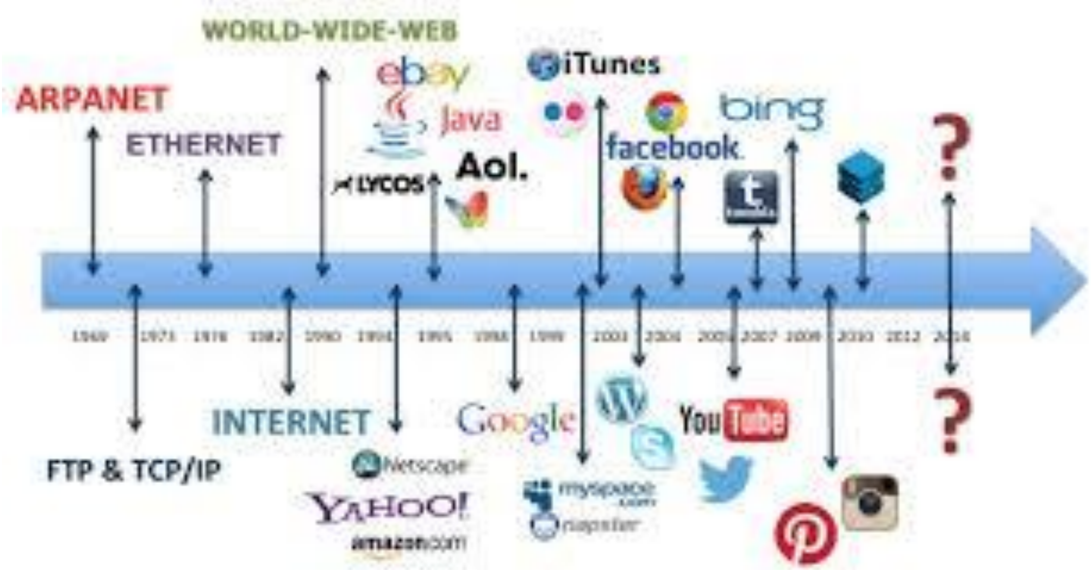



With Electronic Stability Control
Without Electronic Stability Control
Oversteer
Compensating Moment
Braking Force



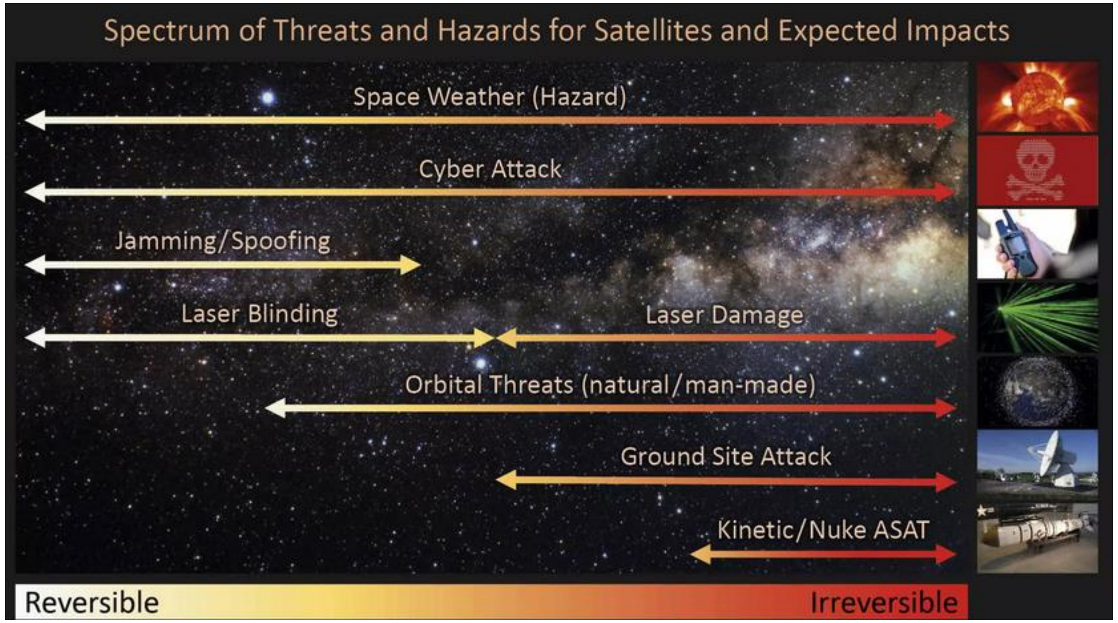
Security Engineering

- Trustworthiness (system will not behave in malicious manner)



Security Engineering

- Survivability (continues to operate when compromised)


Spectrum of Threats and Hazards for Satellites and Expected Impacts
Space Weather (Hazard)
Cyber Attack
Jamming/Spoofing
Laser Blinding
Laser Damage
Orbital Threats (natural/man-made)
Ground Site Attack
Kinetic/Nuke ASAT
Reversible
Irreversible