Pengfei Li (pl189)

Homework 1, ECE 590 & CS320 Software Reliability.

1. Please use your own words to describe the different definitions of **fault**, **error** and **failure**

Fault: some conditions may cause error in the project or program which might cause finally lead to failures.

Error: what we got from the program or project is different from what we want to get in theory. Error is a state of a system, may cause the failure.

Failure: a program cannot satisfy the requirement of what we want, it is a failure.

Fault may cause error, error may or not cause failure

2. Take any program that you have written (paste the piece of codes) and identify at least one fault (bug), the errors it will cause, the inputs that will trigger the fault and failure that results. How can you avoid them next time?

```java
class Solution {
    public boolean isValid(String s) {
        Stack st = new Stack();
        char ch = ' ';
        for(char c:s.toCharArray()){
            if(c=='('||c=='{'||c=='[')
                st.push(c);
            else{
                if(st.size()==0) return false;
                ch = (char)st.pop();
                if((c==')'&&ch!='(')||(c=='}'&&ch!='{')||(c==']'&&ch!='[')) return false;
            }
        }
        return true;
    }
}
```

Here is the code that I write to determine the Valid Parentheses, for example, "[()]{}" is valid, "{" or "{}]" are not valid.
The fault is after the for loop, I did not check the size of the stack, if the input string has more left parentheses than right parentheses, it will cause the error, and finally cause the failure.

For example, when the input is "{", it would push it in to the stack in line 7, and then end the for loop and go to line 14 without checking the stack size, this fault would cause the error. Obviously, it is an invalid string, but our program gives "true" as the result, it is a failure.

The correct code is:

```java
class Solution {
    public boolean isValid(String s) {
        Stack st = new Stack();
        char ch = ' ';
        for(char c:s.toCharArray()){
            if(c=='('||c=='{'||c=='[')
                st.push(c);
            else{
                if(st.size()==0) return false;
                ch = (char)st.pop();
                if((c==')'&&ch!='(')||(c=='}'&&ch!='{')||(c==']'&&ch!='[')) return false;
            }
        }
        if(st.size()>0) return false;
        return true;
    }
}
```

To avoid this problem, I should consider some corner cases as examples input of my program, such as input size is 0 or 1. And run it line by line to check the code. And always pay attention to for loop or while loop, sometimes I will miss some status check statement after the loop.

3. Find a well-publicized incident, read about it and determine, the fault(s), errors and failure of this accident. And find out how the engineers react after the incident happens?

Facebook service outage (10/04/2021)

Original text: (Original text got from Facebook announcement):

"Our engineering teams have learned that configuration changes on the backbone routers that coordinate network traffic between our data centers caused issues that interrupted this communication. This disruption to network traffic had a cascading effect on the way our data centers communicate, bringing our services to a halt."

Fault, errors, and failure:

Our engineering teams have learned that configuration changes on the backbone routers that coordinate network traffic between our data centers (fault) caused issues that interrupted this communication (error). This disruption to network traffic had a cascading effect on the way our data centers communicate (error), bringing our services to a halt (failure).

Engineers react: (partially cite from Facebook announcement)

Engineers confirmed that the outage was caused by a Domain Name System (DNS) problem and ruled out the possibility that the outage originated from the cyber-attack. And Facebook sent engineers onsite to the data centers to have them debug the issue and restart the systems.

4.   Try to describe the classification of errors.

(Cite from Basic Concepts and Taxonomy of Dependable and Secure Computing)

A convenient classification of errors is to describe them in terms of the elementary service failures that they cause, **content versus timing errors, detected versus latent errors, consistent versus inconsistent errors, minor versus catastrophic errors**. In the field of error control codes, content errors are further classified according to the damage pattern: single, double, triple, byte, burst, erasure, arithmetic, track, etc., errors.
Some faults (e.g., a burst of electromagnetic radiation) can simultaneously cause errors in more than one component. Such errors are called multiple related errors. Single errors are errors that affect one component only.