Assignment #2

Due date: 10/10/2023 at 23.55

Question #1

Given is the following statement, while version control systems have advanced in the last couple of years, fully automated merging remains a challenge and is typically not recommended.

Is the above statement correct – briefly justify your answer (max 50 words)

Answer to Question #1: Yes, the statement is correct. While automated merging can handle syntactical correctness, it often struggles with semantic correctness and can override changes in conflicting code regions when multiple users modify code simultaneously.

Question #2

In a recent department meeting your new boss made the following statement. I just read a research paper which included the following statements. “The key objectives of Continious Integration is (1) to reduce the frequency of builds and therefore (2) making it easier to identify and fix programming and integration errors”

Clearly indicate, for each of the two statements, if you agree / disagree with it and correct them if necessary. Briefly justify your answer (1-2 sentences).

Answer to Question #2: Disagree. While Git supports concurrent development, it doesn't entirely eliminate the need for merging. Git defers merging until later stages, requiring developers to eventually reconcile their code with the main branch.

Question #3

You are attending a workshop and one of the presenters makes the following statements. 1.) Dependency management systems such as Maven will help managing build dependencies and the automation of the build process. 2.) Such build management system will also remove technical debt, by eliminating all built errors.

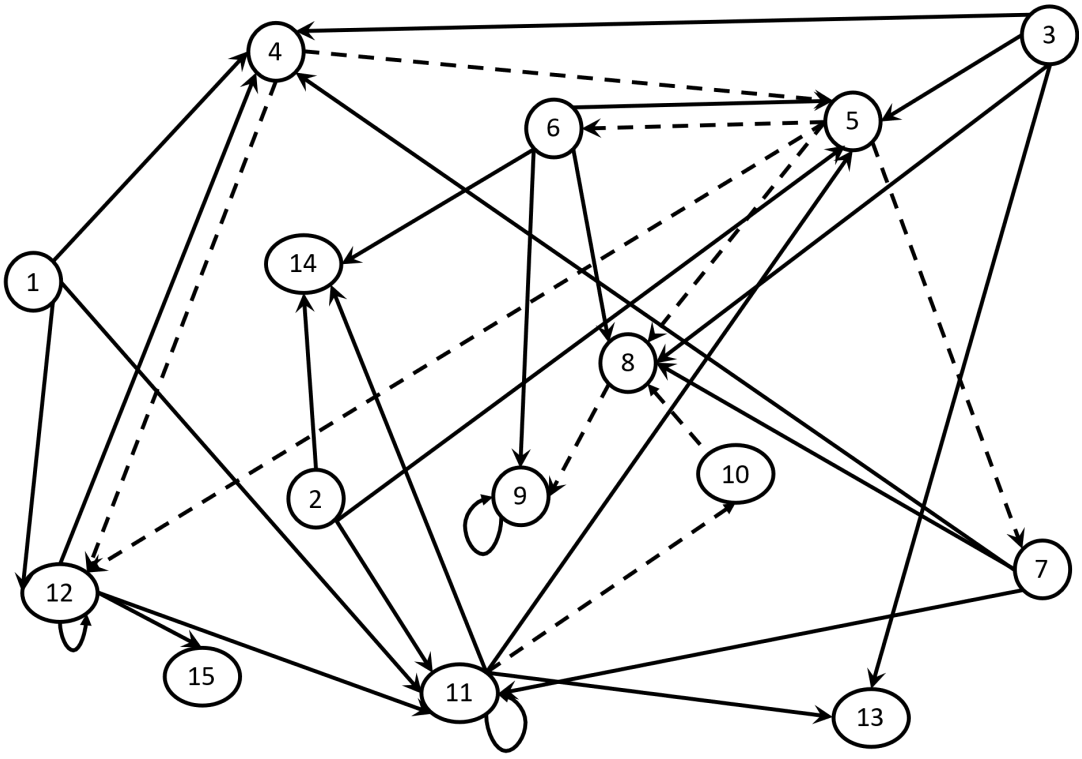
Clearly indicate, if you agree / disagree with the above statement. Briefly justify your answer (1-2 sentences).

Disagree. While build management systems can help reduce technical debt by catching build errors early, they cannot eliminate all built errors, as technical debt may involve various aspects beyond just build issues, such as architectural or design problems

Question #4

Given is the following program and PDG

```
VOID MAIN ()
{
1.  CIN>>X;
2.  CIN>>Y;
3.  CIN>>Z;
4.  IF (Y<Z)
    {
5.    WHILE (Z>X)
      {
6.      Y = 100;
7.      CIN >>Z;
8.      IF (Z<Y)
        {
9.        Y = Y+Z;
        }
10.     ELSE
        {
11.       Z=Z+X+Y;
        }
      }
12. X++;
    }
13. COUT<<Z;
14. COUT <<Y;
15. COUT <<X;
}
```



Verify the above static PDG for correctness – use the tables on the next page to mark the incorrect/missing dependencies.

PLEASE COMPLETE THE PDG TABLES BELOW (IT MATCHES THE PDG FROM THE PREVIOUS PAGE)

PLEASE COORECT THE PDG BY ADDING MISSING DEPENDENCIES, REMOVING WRONG DEPENDENCIES IN THE TABLE BELOW

WRONG DEPENDENCIES – FILL THE BOX ■ ; ADDING A MISSING DEPENDENCIES USE A CIRCLE ● SEE EXAMPLE BELOW

ORIGINAL CORRECTED

1. Y =10
2. IF (Y)
3. Y=Y+1;

	Is **data** dependent on			
Node		1	2	3
	1			
	2	X		
	3			X

	Is **data** dependent on			
Node		1	2	3
	1			
	2	X		
	3			

●

Node		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1															
	2															
	3															
	4	x	●	x				x					x			
	5	●	x	x			x	●				x				
	6															
	7															
	8			x			x	x								
	9						x	●		x						
	10															
	11	x	x				●	x				x	x			
	12	x											x			
	13			x				●				x				
	14		x				x			●		x				
	15	●											x			

Data Dependencies
Control Dependencies

Node		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1															
	2															
	3															
	4															
	5				x											
	6					x										
	7					x										
	8					x					x					
	9								x							
	10								●			x				
	11										●					
	12				x	x										
	13															
	14															
	15															