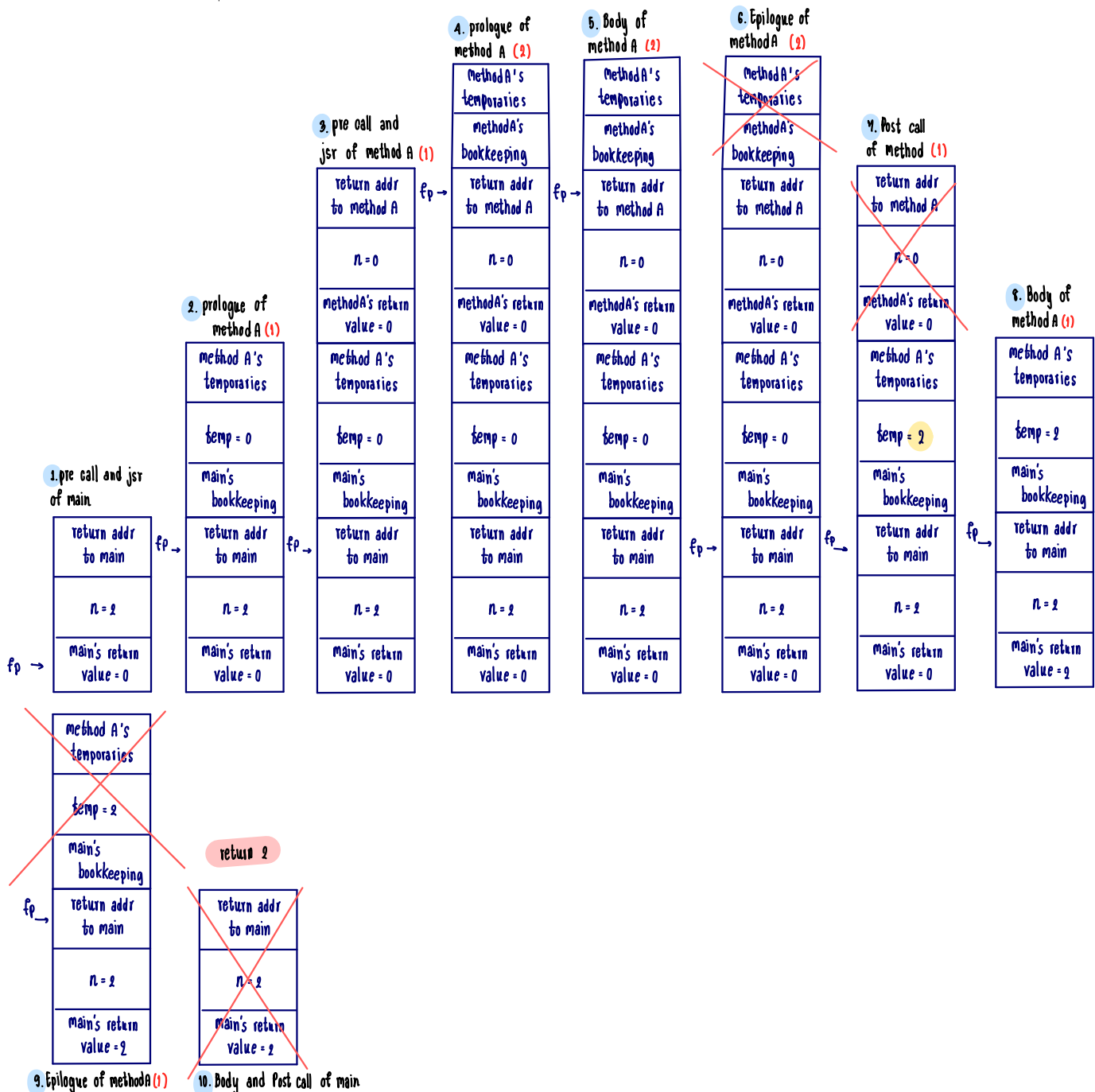


1. (10 คะแนน) มีโปรแกรมหนึ่งมีโค้ดดังนี้

```
int main(){
    return methodA(2);
}

int methodA(int n){
    if(n <= 0){
        return 0;
    }
    int temp;
    temp = n+methodA(n-2);
    return temp;
}
```

จงวาดสแตคในหน่วยความจำแสดงสิ่งที่เกิดจากการเรียกเมธอดแต่ละครั้ง จนถึงขณะที่เมธอดรัน methodA(0) เสร็จแล้วกำลังจะรีเทิร์น วาดรายละเอียดให้ได้มากที่สุดเท่าที่จะวาดได้



2. (10 คะแนน) มีโค้ดเมธอดของภาษาที่มี short-circuit Boolean evaluation ดังนี้

```
int f(int a, int b, int c, int d, int e, int f) {  
    int result = 0;  
    if( (a>b || c> d) && b != c) {  
        for(int i=1; i<=f; i++)  
            result = result + f;  
    }  
    return result;  
}
```

ถ้าภาษานี้ไม่มี short-circuit แต่เราต้องการให้การเช็คและรันเหมือนกับภาษาที่ใช้ short-circuit จงเขียนเมธอดนี้ใหม่

```
int f(int a, int b, int c, int d, int e, int f){  
    int result=0;  
    if (a>b){  
        if (b!=c){  
            for (int i=1; i<=f; i++){  
                result = result + f;  
            }  
        }  
    }else if (c>d){  
        if (b!=c){  
            for (int i=1; i<=f; i++){  
                result = result + f;  
            }  
        }  
    }  
    return result;  
}
```

3. (4 คะแนน) มีนิยามฟังก์ชันที่รับค่า `int` ที่ต้องเขียนดังนี้

$$\begin{aligned} f(i) &= i, \text{ if } 1 \leq i \leq 100 \\ &= 2i, \text{ if } 101 \leq i \leq 550 \\ &= 3i, \text{ if } 551 \leq i \leq 1000 \\ &= 0, \text{ otherwise} \end{aligned}$$

สมมติว่า `switch` สามารถเขียนโดยระบุ `range` ได้ เช่น 1 ถึง 10 ก็เขียนได้เลยว่า 1..10 ถ้าว่า นิสิตจะเลือกเขียนเมธอดนี้โดยใช้ `if else` หรือ `switch statement` จงบอกเหตุผลที่เลือก

ใช้ `if-else` เพราะ ช่วงของข้อมูลของฟังก์ชันขนาดใหญ่ การใช้ `switch statement` จะส่ว `jump table` จะสะดวกซึ่งไม่ต้องพิมพ์ซ้ำๆ ซากๆ

4. มีโค้ดของภาษาที่เมธอด nest กันได้ ดังนี้:

```
public class TheMohegan{
    public static void main(String[] args){
        int x =2;
        int y =2;
        int z = 3;
        public void method01(int x){
            public void method2(int y){
                int x = z+1;
                int m = x+y;
                method3(m);
            }

            public void method3(int m){
                z = z+x+m;
                System.out.println(x + "," + y + "," + z); //line1
            }

            y = y-x;
            x+=y;
            method2(x);
        }
        method1(x+1);
        System.out.println(x + "," + y + "," + z); //line2
    }
}
```

static

main  
 $x=2$   
 $y=2$   
 ~~$z=3$~~  11  
 method1(int  $x=2$ )  
 $y=y-x+1$   
 $x=2$   
 method2(int  $y=2$ )  
 $x=4$   
 $m=4+2=6$   
 method3(int  $m=6$ )  
 $z=3+2+6$   
 $=11$   
 2, -1, 11

dynamic

main  
 $x=2$   
 $y=2$   
 $z=3$   
 method1(int  $x=2$ )  
 $y=y-x+1$   
 $x=2$   
 method2(int  $y=2$ )  
 $x=4$   
 $m=4+2=6$   
 method3(int  $m=6$ )  
 $z=3+4+6$   
 $=13$   
 4, 2, 11

a. (3 คะแนน) ถ้าใช้ static scope, line1 กับ line 2 จะพิมพ์อะไรออกมา

line 1 : 2, -1, 11  
 line 2 : 2, -1, 11

b. (3 คะแนน) ถ้าใช้ dynamic scope, line1 กับ line 2 จะพิมพ์อะไรออกมา

line 1 : 4, 2, 13  
 line 2 : 2, -1, 13