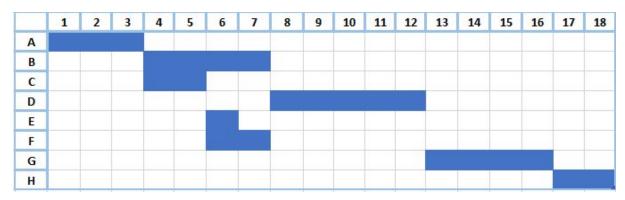
# **Question No 2 - Slide 26**

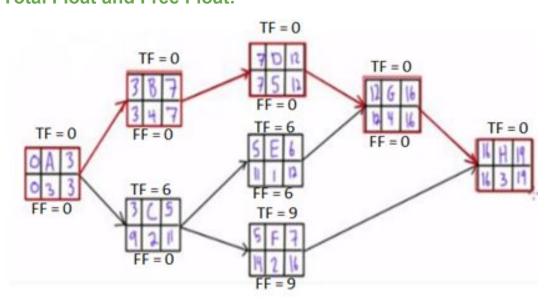
## **Activity Table:**

| Activity | Predecessor | Duration |
|----------|-------------|----------|
| Α        |             | 3        |
| В        | Α           | 4        |
| С        | Α           | 2        |
| D        | В           | 5        |
| E        | С           | 1        |
| F        | С           | 2        |
| G        | D, E        | 4        |
| н        | F, G        | 2        |

### **Gantt Chart:**



### **Total Float and Free Float:**

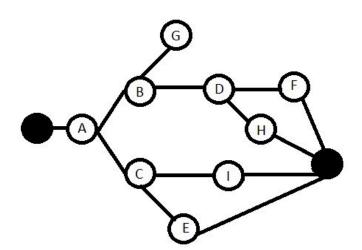


# **Question 2 - Slide 24**

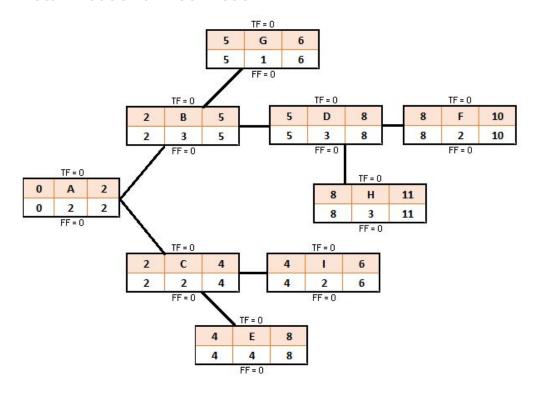
# **Activity Table:**

| Activity | Predecessor | Duration |
|----------|-------------|----------|
| A        |             | 2        |
| В        | A           | 3        |
| С        | A           | 2        |
| D        | В           | 3        |
| E        | С           | 4        |
| F        | D           | 2        |
| G        | В           | 1        |
| Н        | D           | 3        |
| ı        | С           | 2        |

## **Activity Network:**



### **Total Float and Free Float:**



### **Critical Path:**

$$\mathsf{A}\to\mathsf{B}\to\mathsf{D}\to\mathsf{F}$$

$$\mathsf{A} \to \mathsf{B} \to \mathsf{D} \to \mathsf{H}$$

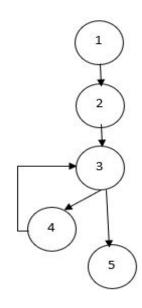
# **Question 1**

## **Find Factorial of a Number:**

### Begin:

- 1. int x=1, fact=1, n;
- 2. cin >> n;
- 3. while(i < = n){
- 4. fact=fact\*i; }
- 5. print result.

End.

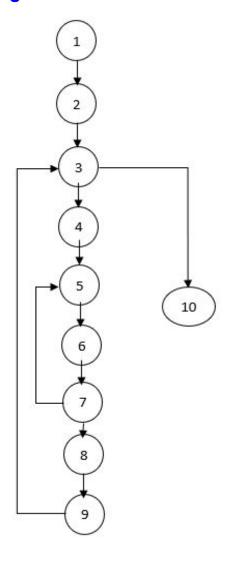


## Prime number between two integers:

#### Begin:

- 1. Int a,b, x, check;
- 2. Cin >> a >> b;
- 3. While(a<b){
- 4. check=0;
- 5. for (i = 2; i < a/2; i++)
- 6. if(a % 1 = = 0){
- 7.  $check = 1;}$
- 8. if( check == 0 ){
- 9. a++;
- 10. print result.

End.

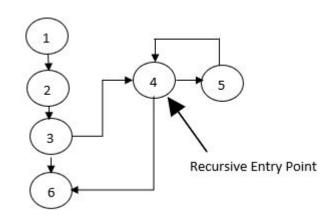


## **Sum of N natural numbers using recursion:**

#### Begin:

- 1. int n;
- 2. Cin >> n;
- 3. Cout << "Sum = " << addFunction(n);
- 4. If (n != 0)
- 5. Return n + addFunction(n-1);
- 6. return 0;

End.

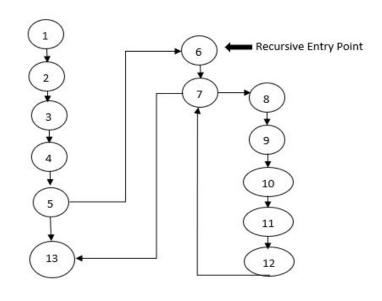


## **Print Fibonnaci series of N numbers using recursion:**

### Begin:

- 1. int n;
- 2. cout << "Enter number of elements;
- 3. cin >> n;
- 4. cout << "0" << "1";
- 5. FibonacciFunctionCall(n-2);
- 6. int n1=0,n2=1,n3;
- 7. If (n > 0)
- 8. n3 = n1+n2;
- 9. n1=n2;
- 10. n2=n3;
- 11. cout << n3;
- 12. FibonacciFunctionCall(n-1);}}
- 13. return 0;

End.



## **Print Pyramid of N numbers:**

#### Begin:

- 1. int i, space, rows, k=;
- 2. cin >> rows;
- 3. for(i=1; i<rows; i++){
- 4. for(space = 1; space <= (rows-1); space++){
- 5. cout << " ";}
- 6. while(k != (2\*i-1)){
- 7. cout << " ";
- 8. k++;}
- 9. k=0;
- 10. cout << "\n";}
- 11. getch();

End.

