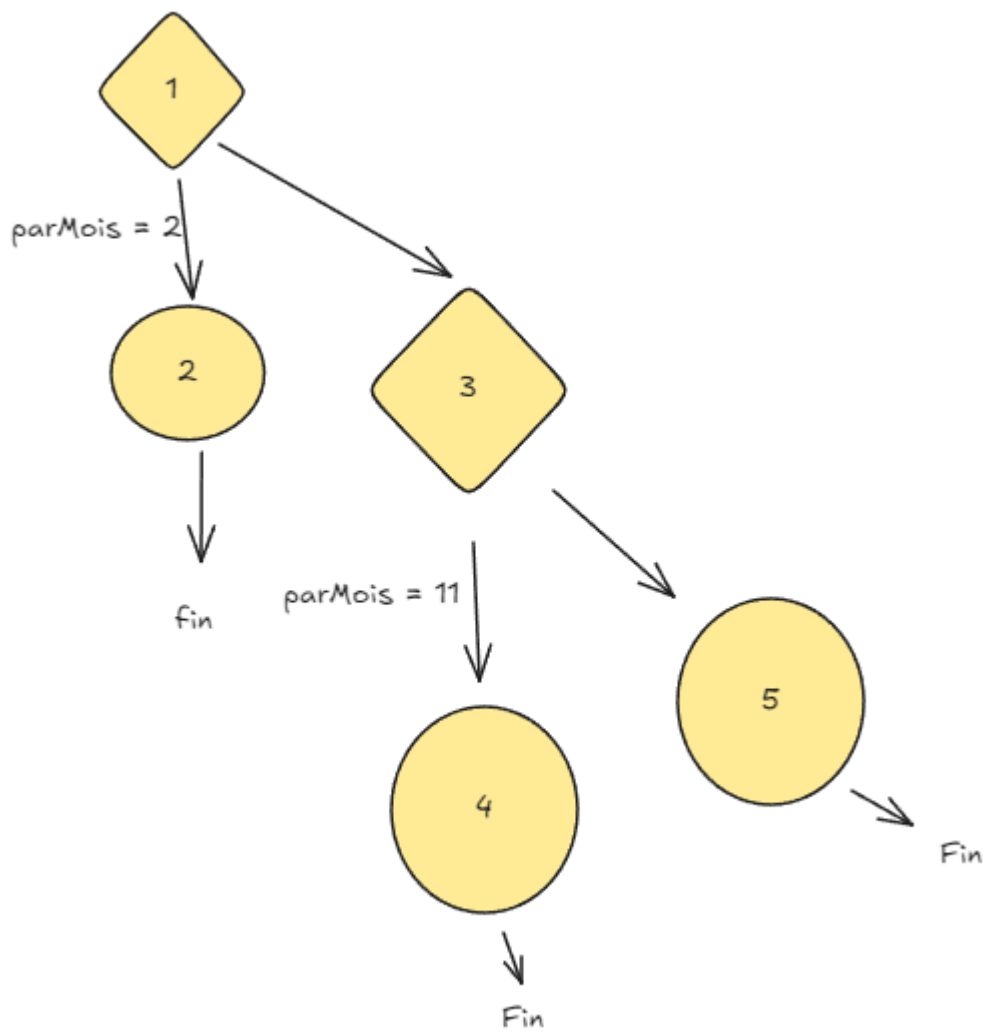


méthode : dernierJourDuMois

Test	MOIS	ANNEE	Résultats attendus
P1	mois	annee	nbr jour du mois
P2	mois	annee	nbr jour du mois
P3	mois	annee	nbr jour du mois
P4	mois	annee	nbr jour du mois

Test	MOIS	ANNEE	Résultats attendus
P1	2	2023	28
P2	2	2024	29
P3	3	2023	31
P4	3	2024	31

```
protected static int dernierJourDuMois(int parMois, int parAnnee) { 3 usages
    switch (parMois) {
        case 2: —————> ①
            return estBissextile(parAnnee) ? 29 : 28; } ②
        case 4:
        case 6:
        case 9:
        case 11: —————> ③
            return 30; } ④
        default:
            return 31; } ⑤
    }
}
```



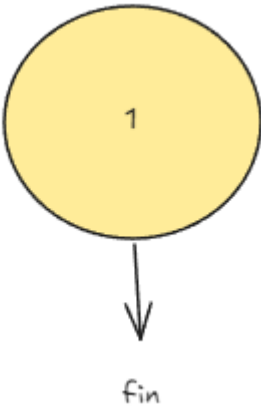
méthode : EstBissextile

Test	ANNEE	Résultats attendus
P1	pasbissextile	false
P2	pasbissextile	false
P3	bissextile	true

P4	pasbissextile	false
P5	bissextile	true

Test	ANNEE	Résultats attendus
P1	2013	false
P2	1581	false
P3	2024	true
P4	1900	false
P5	2000	true

```
private static boolean estBissextile(int parAnnee) { 1 usage
    return (parAnnee % 4 == 0 && parAnnee % 100 != 0) || parAnnee % 400 == 0;
}
```



méthode : dateDuLendemain

Test	ANNEE	MOIS	JOUR	Résultats attendus
P1	annee	mois	jour	anne, mois, jour +1

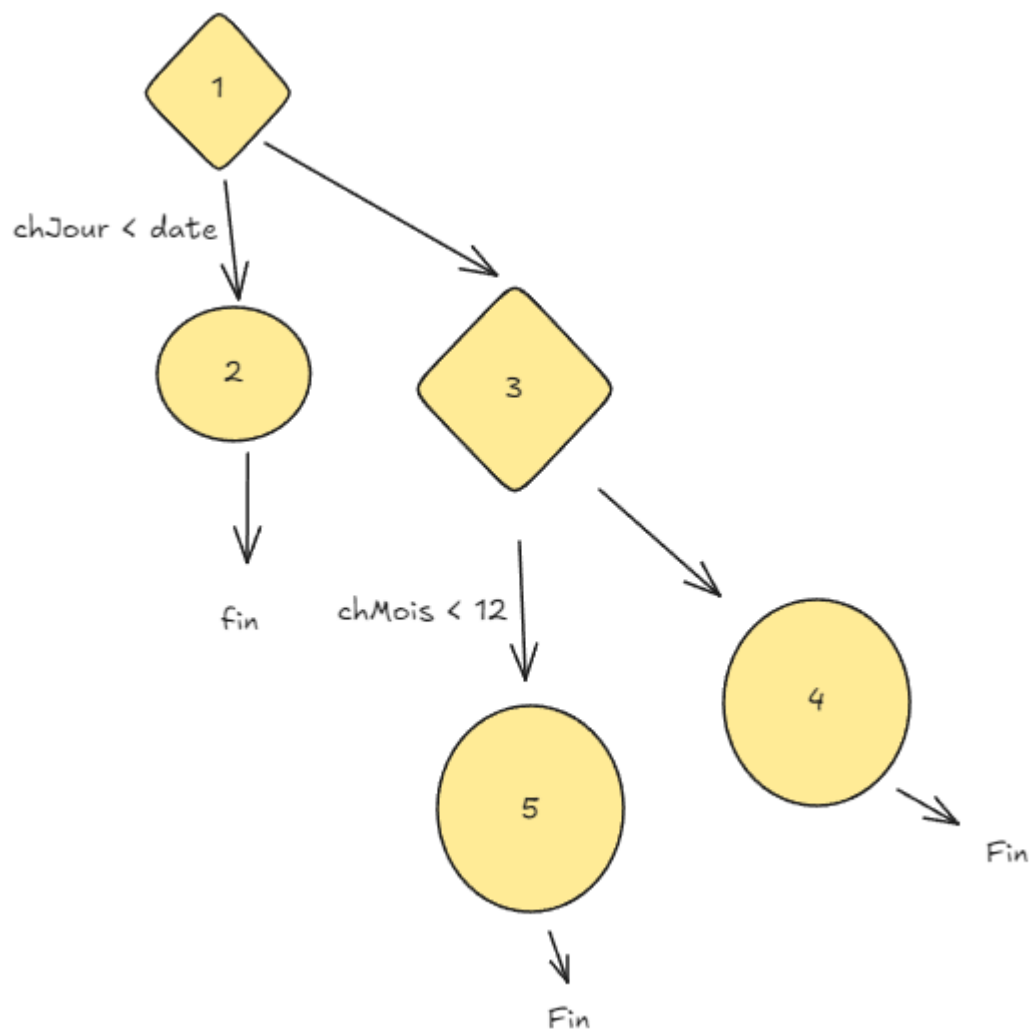
Test	ANNEE	MOIS	JOUR	Résultats attendus
------	-------	------	------	--------------------

P1	2024	2	20	2024-02-21
----	------	---	----	------------

```

public Date dateDuLendemain() { 2 usages 1 override
    if (chJour < Date.dernierJourDuMois(chMois, chAnnee))
        return new Date( parJour: chJour + 1, chMois, chAnnee);
    if (chMois < 12)
        return new Date( parJour: 1, parMois: chMois + 1, chAnnee);
    return new Date( parJour: 1, parMois: 1, parAnnee: chAnnee + 1);
}

```

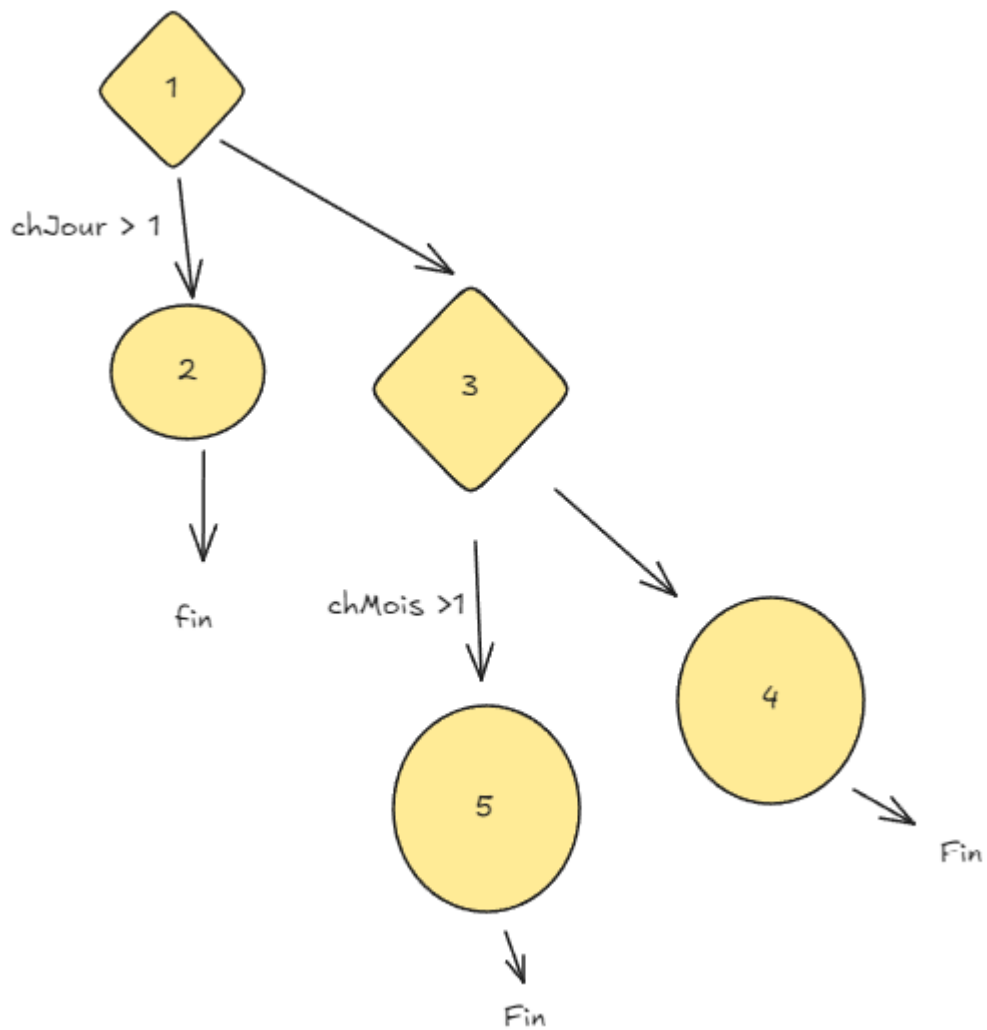


méthode : dateDeLaVeille

Test	ANNEE	MOIS	JOUR	Résultats attendus
P1	annee	mois	jour	2024-02-20

Test	ANNEE	MOIS	JOUR	Résultats attendus
P1	2024	2	21	2024-02-20

```
public Date dateDeLaVeille() { 2 usages 1 override
    if (chJour > 1) Ⓚ①
        return new Date( parJour: chJour - 1, chMois, chAnnee); Ⓜ②
    if (chMois > 1) Ⓚ③
        return new Date(Date.dernierJourDuMois( parMois: chMois - 1, chAnnee), parMois: chMois - 1, chAnnee); Ⓜ④
    return new Date( parJour: 31, parMois: 12, parAnnee: chAnnee - 1); Ⓜ⑤
}
```



Méthode : EstValide

Test	ANNEE	MOIS	JOUR	Résultats attendus
P1	valide	valide	valide	true
P2	valide	valide	invalide	false
P3	valide	invalide	valide	false
P4	invalide	valide	valide	false

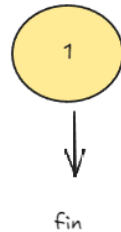
Test	ANNEE	MOIS	JOUR	Résultats attendus
------	-------	------	------	--------------------

P1	2020	2	21	true
P2	2020	2	32	false
P3	2020	13	20	false
P4	-10	12	20	false

```

public boolean estValide() { 1 usage
    return chAnnee > 1582 &&
        chMois >= 1 && chMois <= 12 &&
        chJour >= 1 && chJour <= dernierJourDuMois(chMois, chAnnee);
}

```



Méthode : CompareTo

Test	Date 1 (ANNEE, MOIS, JOUR)	Date 2 (ANNEE, MOIS, JOUR)	Résultat attendu
P1	annee -1, mois, jour	annee, mois, jour	-100
P2	annee, mois, jour	annee -1, mois, jour	100
P3	annee , mois-1, jour	annee, mois, jour	-12
P4	annee, mois, jour	annee, mois-1, jour	12
P5	annee , mois, jour-1	annee, mois, jour	-7
P6	annee, mois, jour	annee , mois, jour-1	7
P7	annee, mois, jour	annee, mois, jour	0

Test	Date 1 (ANNEE, MOIS, JOUR)	Date 2 (ANNEE, MOIS, JOUR)	Résultat attendu
P1	2023, 5, 10	2024, 5, 10	-100
P2	2024, 5, 10	2023, 5, 10	100
P3	2024, 4, 10	2024, 5, 10	-12
P4	2024, 6, 10	2024, 5, 10	12
P5	2024, 5, 9	2024, 5, 10	-7
P6	2024, 5, 11	2024, 5, 10	7

P7	2024, 5, 10	2024, 5, 10	0
----	-------------	-------------	---

```

public int compareTo(Date parDate) {
    if (chAnnee < parDate.chAnnee) ———— ①
        return -100; ②
    if (chAnnee > parDate.chAnnee) ———— ③
        return 100; ④
    if (chMois < parDate.chMois) ———— ⑤
        return -12; ⑥
    if (chMois > parDate.chMois) ———— ⑦
        return 12; ⑧
    if (chJour < parDate.chJour) ———— ⑨
        return -7; ⑩
    if (chJour > parDate.chJour) ———— ⑪
        return 7; ⑫
    return 0; ⑬
}

```

