Disseny de Jocs de Proves

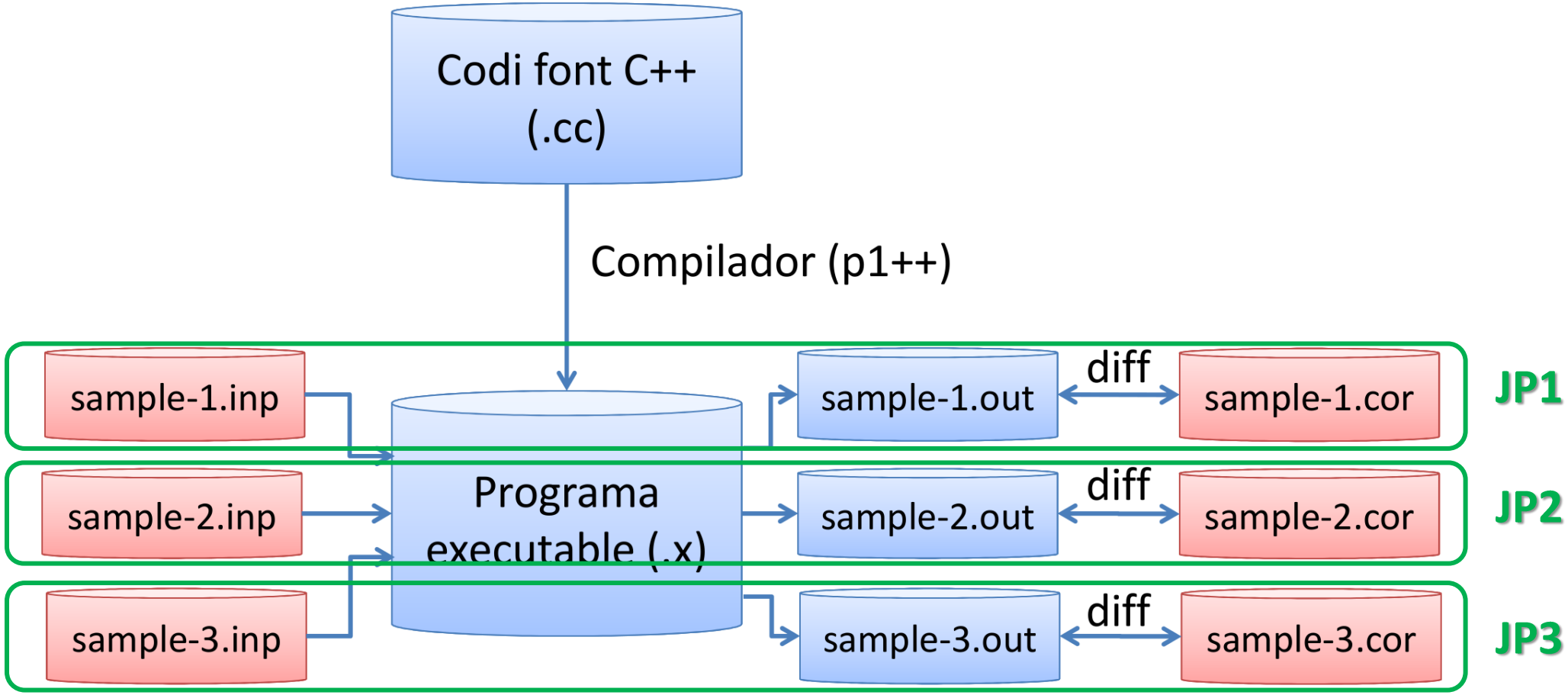
PRO1

Josep Carmona, Lluís Padró

Index

- **Breu descripció**
- Pas 1: Fer servir els JPs que ens facilita el jutge
- Pas 2: Dissenyar JPs propis
- Exercici

Els Jocs de Proves (JP)



Els Jocs de Proves (JP)

- Permeten tenir una certa confiança respecte la correctessa dels programes (però no un 100% de confiança!).
- Eviten enviaments al jutge.org. Recordeu que la normativa actual d'avaluació de **controls** de l'assignatura **penalitza** a partir del tercer enviament.
- Acostumar-se a dissenyar JPs és una molt bona pràctica, que us servirà en el futur (dintre i fora de la carrera), ja que no hi ha jutge a la vida real.

Index

- Breu descripció
- **Pas 1: Fer servir els JPs que ens facilita el jutge**
- Pas 2: Dissenyar JPs propis
- Exercici

Fer servir els JPs del jutge

1. Baixar els fitxers del problema a la teva zona personal:

Statement

Write a program to compute powers.

Input

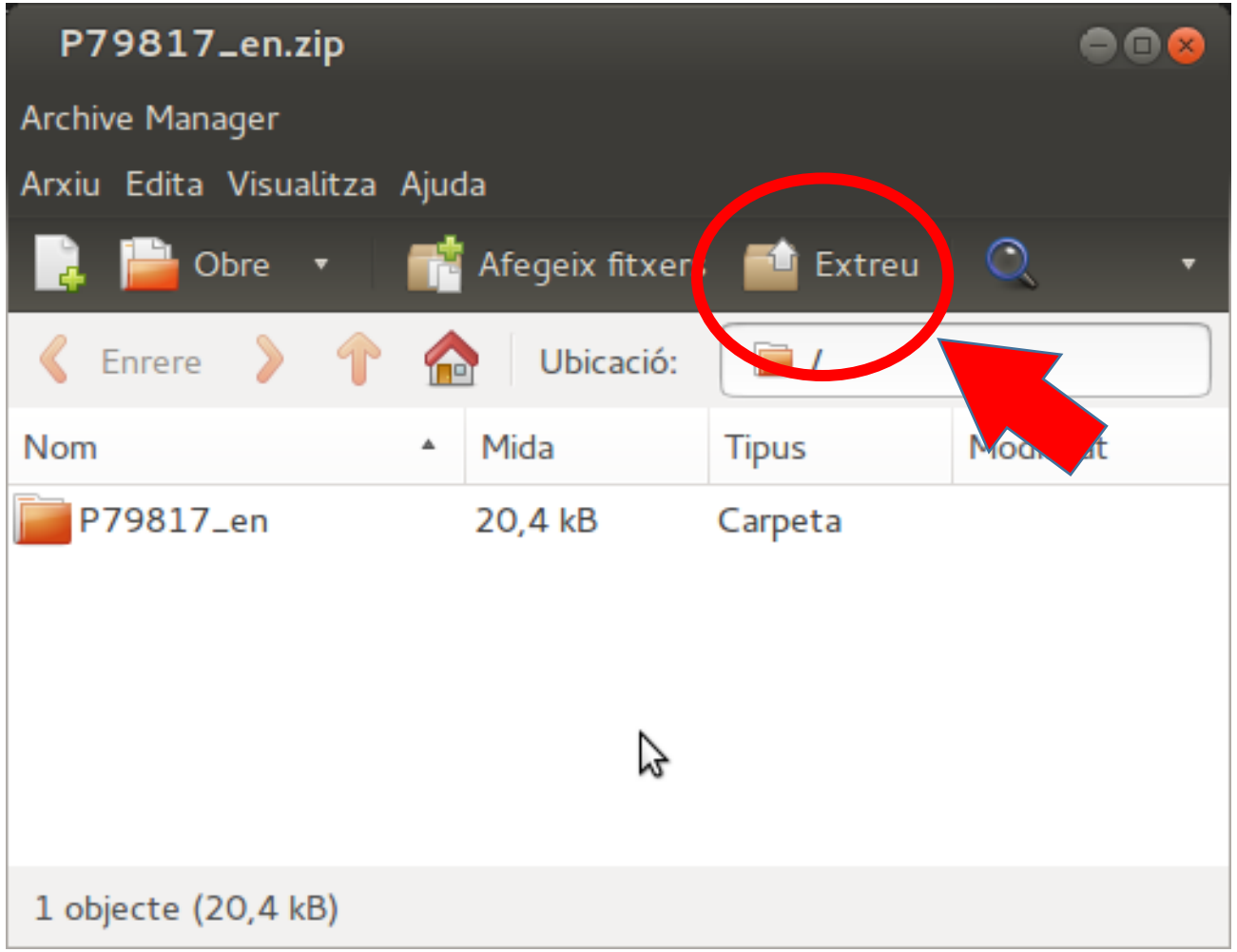
Input consists of several pairs of integer numbers *a* and *b*. Assume $b \geq 0$.

Output

For every pair *a*,*b*, print a^b. Supose, as usual, that $0^0 = 1$.

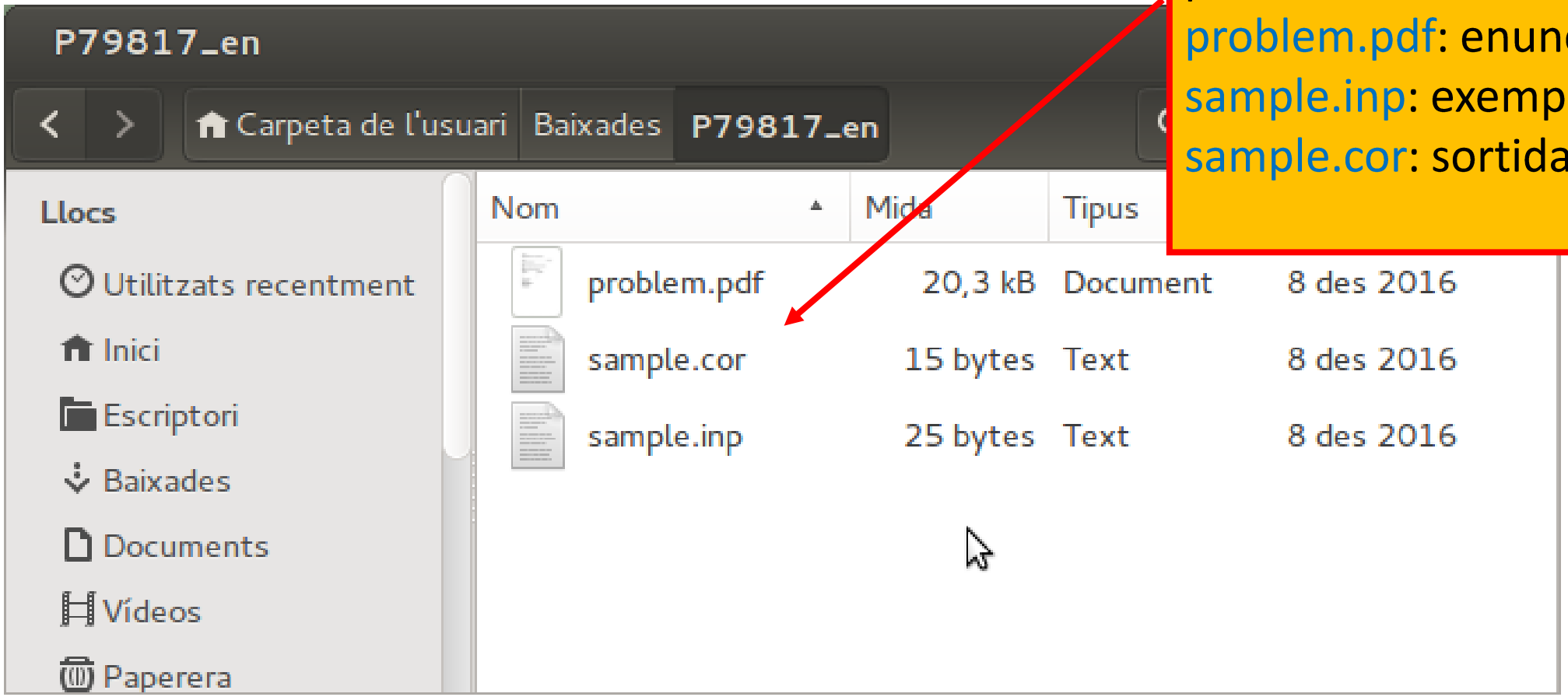
Fer servir els JPs del jutge

2. Descomprimir el fitxers del problema:



Fer servir els JPs del jutge

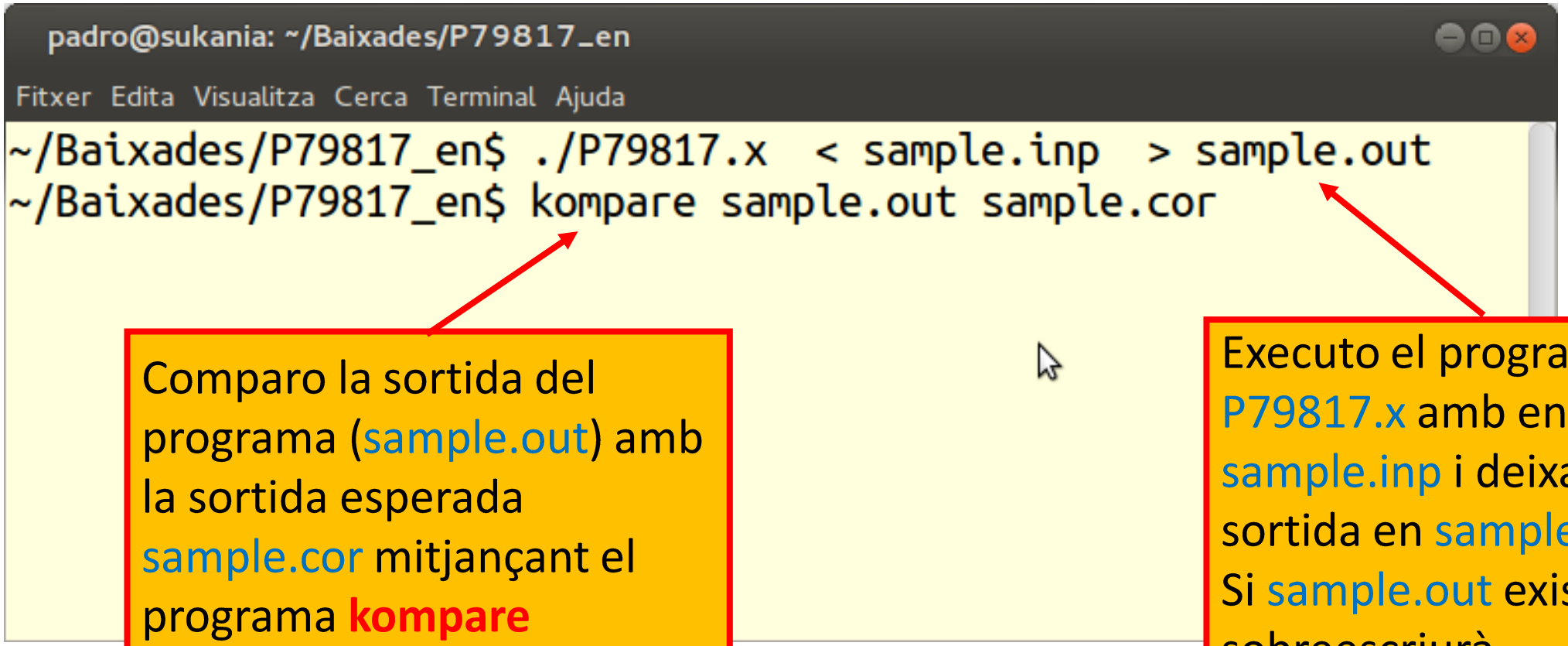
2. Descomprimir el fitxers del problema:



Un cop descomprimit, tenim una carpeta amb la informació del problema:
problem.pdf: enunciat
sample.inp: exemple d'entrada
sample.cor: sortida esperada

Fer servir els JPs del jutge

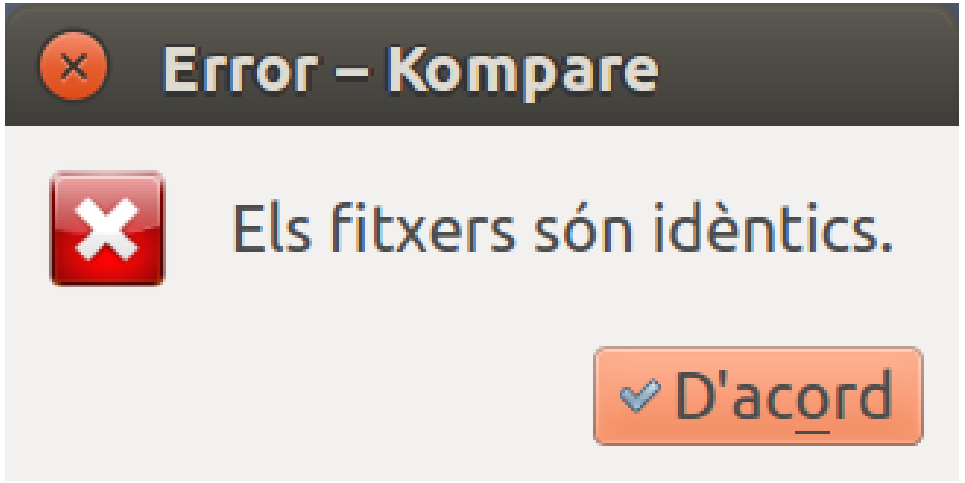
3. Executar el teu programa guardant la sortida en un arxiu.
4. Comparar la sortida del teu programa amb la sortida esperada.



Fer servir els JPs del jutge

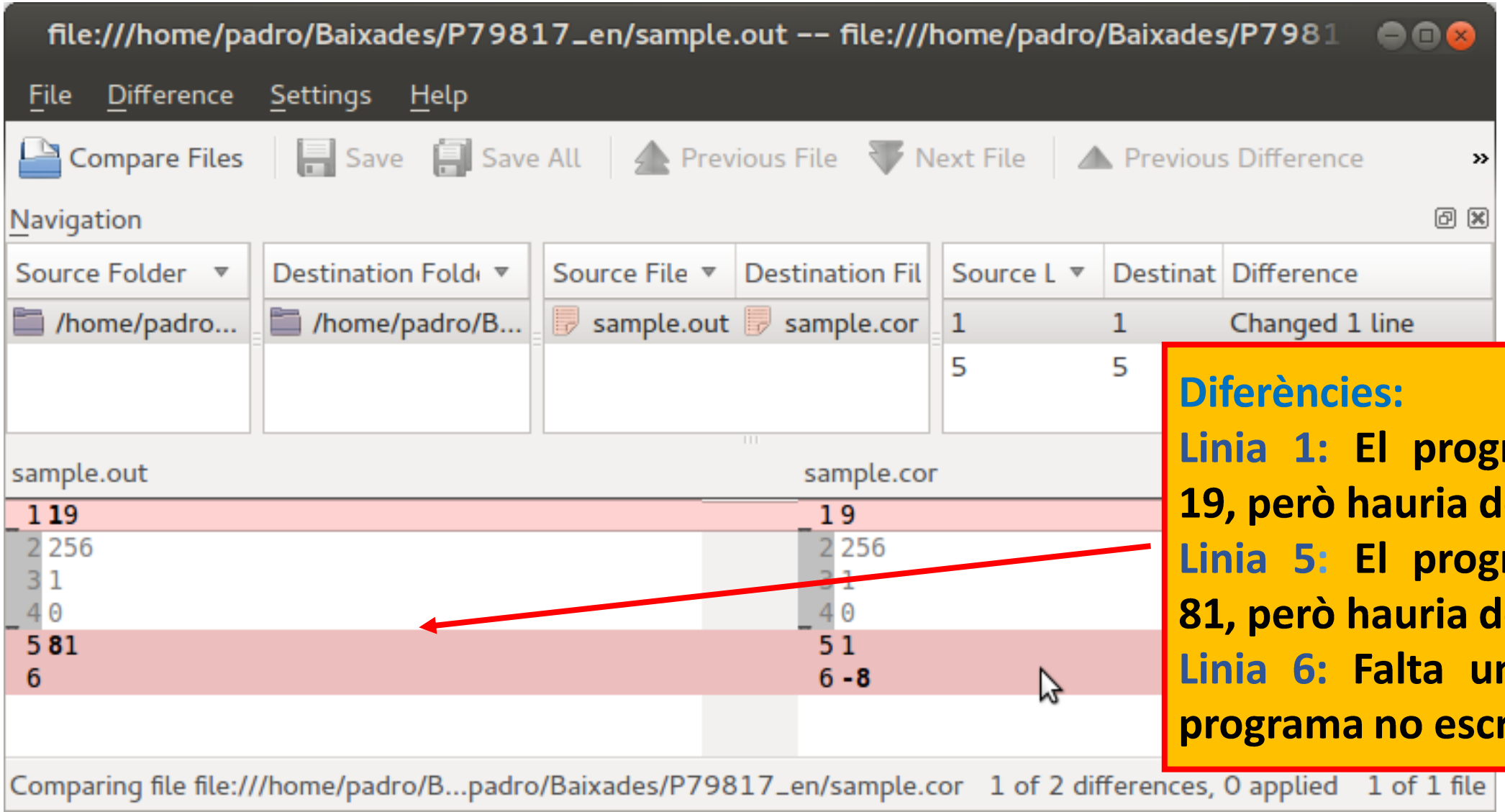
5. Sortida de **kompare.**

A) programes generen mateixa sortida



Fer servir els JPs del jutge

5. Sortida de **kompare**. B) Existeixen discrepàncies



Diferències:

Linia 1: El programa escriu 19, però hauria de ser un 9

Linia 5: El programa escriu 81, però hauria de ser un 1

Linia 6: Falta un -8 que el programa no escriu

Index

- Breu descripció
- Pas 1: Fer servir els JPs que ens facilita el jutge
- **Pas 2: Dissenyar JPs propis**
- Exercici

Dissenyar els JPs propis

- Com a regla general, un bon JP ha de cobrir tot el codi.
- Anem a veure-ho amb el codi per decidir si un nombre és primer.

Prime number

```
// Input:  read a natural number N>0
// Output: write "is prime" or "is not prime" depending on
//         the primality of the number

int main() {
    int N;
    cin >> N;

    int divisor = 2;
    bool is_prime = (N != 1);
    // 1 is not prime, 2 is prime, the rest enter the loop (assume prime)

    // is_prime is true while a divisor is not found
    // and becomes false as soon as the first divisor is found
    while (divisor < N) {
        if (N%divisor == 0) is_prime = false;
        divisor = divisor + 1;
    }

    if (is_prime) cout << "is prime" << endl;
    else cout << "is not prime" << endl;
}
```

EXECUTION 1: N is not a prime number

Prime number

```
// Input:  read a natural number N>0
// Output: write "is prime" or "is not prime" depending on
//         the primality of the number

int main() {
    int N;
    cin >> N;

    int divisor = 2;
    bool is_prime = (N != 1);
    // 1 is not prime, 2 is prime, the rest enter the loop (assume prime)

    // is_prime is true while a divisor is not found
    // and becomes false as soon as the first divisor is found
    while (divisor < N) {
        if (N%divisor == 0) is_prime = false;
        divisor = divisor + 1;
    }

    if (is_prime) cout << "is prime" << endl;
    else cout << "is not prime" << endl;
}
```

EXECUTION 2: N is a prime number

Dissenyar els JPs propis

- Cal llegir bé l'enunciat per determinar aquells casos extrems que compleixen l'enunciat però que potser requereixen un tractament especial.
- Anem a veure uns exemples d'aquestes situacions.

Dissenyar els JPs propis

Statement


Write a program that, given a number *n*, prints a “triangle of size *n*”.

Input

Input consists of a natural number *n*.

Output

Print *n* lines, in such a way that the *i*-th line contains *i* asterisks.

Que ha d'imprimir el programa si n=0 ?

Dissenyar els JPs propis

Statement


Write a program that reads two numbers *a* and *b*, and prints all numbers between *a* and *b*.

Input

Input consists of two natural numbers *a* and *b*.

Output

Print a line with *a*,*a*+1,…,*b*−1,*b*. Separate the numbers with commas.

Que ha d'imprimir el programa si:
1) a = b
2) a > b

Dissenyar els JPs propis

Statement


Write a program that reads a list of words and prints the number of times that the word “`hello`” appears in it.

Input

There input is a sequence of strings.

Output

Print the number of occurrences of the word “`hello`”.

Observation

Take into account that “`hola`” is different from “`Hola`”.

Que ha d'imprimir el programa si:

1) La seqüència d'entrada és buida
2) La seqüència d'entrada no conté la paraula “hello”
3) La seqüència d'entrada conté “Hello”

Index

- Breu descripció
- Pas 1: Fer servir els JPs que ens facilita el jutge
- Pas 2: Dissenyar JPs propis
- **Exercici**

Exercici

- Feu l'exercici X50286 ("*How many hello?*"), aplicant el que heu après en aquest tutorial.