

Piedras preciosas

En la novela "Kim de la India", cuando Kim visita al "Curandero de perlas", éste le propone jugar junto a su otro discípulo al "Juego de las joyas". Sobre una bandeja se encuentran una serie de joyas que cada uno de los jugadores debe recordar, después de mirarlas durante un corto espacio de tiempo. Ganará aquel que consiga describir con mayor exactitud las piedras preciosas. La primera vez que juega, Kim no consigue recordar todas ellas y es vencido por el otro chico. Sin embargo, aprende pronto y las siguientes veces consigue hacer la relación exacta del contenido de la bandeja.



Dado que ahora los dos chicos son capaces de recordar las piedras de la bandeja con toda exactitud, el sahib Lurgan ha decidido modificar un poco el juego para hacerlo más complicado. Coloca las piedras preciosas formando una línea y les pregunta cuantas veces aparece una secuencia de longitud 7 que tenga al menos 3 zafiros y 2 rubís. Viendo que el juego capta su interés sigue realizando este tipo de preguntas, ¿Cuántas veces aparece una secuencia de longitud 5 con al menos 3 diamantes y 1 rubí?, o ¿Cuántas veces aparece una secuencia de tamaño 4 con al menos 2 esmeraldas y 2 jades?

Para no tener que comprobar visualmente que discípulo ha respondido de forma correcta, el sahib desarrolla un programa que dada la lista con las piedras calcula cuantas veces aparece la secuencia deseada. De esta forma no tiene miedo de equivocarse al dar el premio.

Requisitos de implementación.

Para representar las piedras preciosas se utilizará un tipo enumerado

```
enum piedrasPreciosas {diamante, rubi, esmeralda, zafiro, jade};
```

Para leer los valores se sobrecarga el operador extractor para el tipo enumerado anterior:

```
std::istream& operator>> (std::istream& entrada, piedrasPreciosas& p) {
    char num;
    entrada >> num;
    switch (num) {
        case 'd': p = diamante; break;
        case 'r': p = rubi; break;
        case 'e': p = esmeralda; break;
        case 'z': p = zafiro; break;
        case 'j': p = jade; break;
    }
    return entrada;
}
```

Cuando se lee un valor del tipo enumerado se debe utilizar el operador de extracción.

```
int numpiedras; int numtipo1, numtipo2; piedrasPreciosas tipo1, tipo2;
std::cin >> numpiedras >> tipo1 >> numtipo1 >> tipo2 >> numtipo2;
```

El vector debe guardar valores del tipo `piedrasPreciosas`. Y la lectura de los valores del vector se hará con el `for` basado en iteradores.

```
std::vector<piedrasPreciosas> v(numElem);
for (piedrasPreciosas& i : v) std::cin >> i;
```

Entrada

La entrada comienza con el número de casos de prueba. Cada caso tiene dos líneas. En la primera se indica el número de piedras, el tamaño de la secuencia que se busca, el primer tipo de piedra y el número de veces que debe aparecer y el segundo tipo de piedra y el número de veces que debe aparecer. En la segunda línea se indica la lista de piedras preciosas representadas por su inicial en minúsculas.

Las piedras preciosas que se consideran son diamante, rubí, esmeralda, zafiro y jade. Cada una se identifica por el primer carácter de su nombre. Se garantiza que el tamaño de la secuencia es menor o igual que el número de piedras y que la suma del número de veces que debe aparecer la primera y la segunda piedra es menor o igual que la longitud de la secuencia.

Salida

Para cada caso de prueba se escribe en una línea el número de secuencias que cumplen la propiedad pedida.

Entrada de ejemplo

```
4
6 3 d 1 z 1
r d z e d z
7 3 e 2 j 1
e j e r e e j
5 2 r 1 z 0
d z j r e
7 4 z 2 r 1
z r d z z r e
```

Salida de ejemplo

```
4
2
2
4
```

Autor: Isabel Pita.