

Act 5.2 - Actividad Integral sobre el uso de códigos hash

La idea de la técnica de hashing es dispersar los datos sobre el arreglo, realizando una función matemática llamada función de hashing a la llave del elemento para localizar la posición que le coarrendaría. Lo ideal es que esa posición siempre esté disponible para él.

La técnica del hashing debe de preocuparse por dos cosas principalmente:

1. Diseñar una buena función de hashing para distribuir los elementos lo mas aleatoriamente posible en el arreglo y que se puedan ejecutar de forma rápida.
2. Dado que no es posible asegurar una función de hashing que garantice un valor distinto para cada elemento, también se debe de preocupar por el manejo de colisiones.

Como en cualquier empresa, en la compañía naviera "International Seas, LTD." cada empleado tiene una descripción de su puesto de trabajo. Una descripción contiene unos pocos párrafos que describe la responsabilidad del puesto. Esta, junto con otros factores, como la antigüedad, se utiliza para determinar su salario.

En la empresa hemos desarrollado el sistema "Unlimited Seas". Este sistema libera al departamento de Recursos Humanos de tener que hacer un juicio inteligente sobre el valor del empleado; la descripción del puesto simplemente se escanea en busca de palabras y frases que indiquen responsabilidad. En particular, las descripciones de puestos que indican el control de un gran presupuesto o la gestión de un gran número de personas producen puntuaciones altas en "Unlimited Seas".

Lo que haremos en este activad, será implementar un sistema "Unlimited Seas" simplificado. Recibirás como entrada un diccionario y una serie de descripciones de puestos. Para cada descripción, deberás calcular el salario asociado con el puerto, de acuerdo con el sistema.

Entrada

La primera línea de entrada contiene 2 números enteros positivos: m 1,000, el número de palabras en diccionario "Unlimited Seas", y n 100, el número de descripciones de puestos. Las siguientes m líneas; cada una contiene una palabra y un valor en dólares (un número entero entre 0 y 1,000,000). A continuación, se encuentran las n descripciones de puestos. Cada descripción de puesto de trabajo consta de uno o más líneas de texto; para mayor facilidad, el texto se ha convertido a minúsculas y no tiene

más caracteres que letras, números y espacios. Cada descripción de puesto de trabajo con una línea que contiene un punto.

Salida

Para cada descripción de puesto de trabajo, deberás calcular el salario correspondiente calculado como la suma de los valores de "Unlimited Seas" para todas las palabras que aparecen en la descripción. Las palabras que no aparecen en el diccionario tienen un valor de 0.

Solución

Para este código se realizó una solución que pueda servir en escenarios reales, se utilizó correctamente el método de hashing para poder mapear los datos que se pedían.

Test cases

```
7 4
administer 100000
spending 200000
manage 50000
responsibility 25000
expertise 100
skill 50
money 75000
the incumbent will administer the spending of kindergarden milk money
and exercise responsibility for making change he or she will share
responsibility for the task of managing the money with the assistant
whose skill and expertise shall ensure the successful spending exercise
.
this individual must have the skill to perform a heart transplant and
expertise in rocket science
.
skill
individual
skill
.
.
```

En estos test cases se pueden mostrar distintas pruebas, dos normales con datos regulares, una que inicia y termina con una palabra que se encuentra en el diccionario, y la ultima que está vacía.

Output

```
700150
150
100
0
```

En este caso, los outputs resultaron ser los esperados, por lo que el programa está correctamente funcionando.

Complejidades:

getSalario: Este algoritmo presenta una complejidad de $O(n)$, siendo n el número de palabras que presenta el elemento.

readData: Este algoritmo lee los datos del archivo txt, y presenta una complejidad de $O(n+c)$, siendo n el número de datos para el diccionario y c los test cases.

quadratic: la complejidad que presenta este algoritmo es de $O(n * l)$ donde n son los datos y l la longitud de la tabla.

getElemento: Este algoritmo tiene una complejidad de $O(n * l)$

Reflexión

En este problema se busca encontrar el valor de una oración a través de el valor de las palabras que la conforman, por ello el método de hashing para relacionar llaves con valores sale muy conveniente, ya que se puede iterar por toda la oración y si encuentras una palabra que se encuentra en el hash solo sumas su valor.

Este método es muy conveniente cuando se buscan relacionar llaves con sus valores, es muy utilizado, por ejemplo, para encontrar características en algún objeto, passwords, validación de datos etc.

Oportunidades de mejora

- Métodos de hashing más seguros.
- Modelar tamaño del hash desde un interfaz.
- Exception handling