**Problema A**

Carlos Eduardo González Álvarez - 201511169

Javier Antonio Troconis Guia – 201424518

**Algoritmo de solución:**

El problema se resolvió aprovechando la siguiente propiedad del mcd:

Ya que el mcd se puede escribir como combinación lineal de los valores, podemos conseguir un par de números con y obtener los valores de la combinación lineal para resolver el problema. El métodoA revisa de manera ordenada los subcojuntos de tamaño 2; se empieza en el primero y se compara con todos los elementos siguientes, este proceso se repite con todos los otros elementos de la lista.

Métodos Usados:

* metodoA:
* mcdExtendido:

Para la lectura de los datos se tiene que pasar a int los String de la entrada y a al imprimir se tiene que pasar por todo el arreglo de nuevo para mostrar en salida. Se supone que la conversión de String a entero se realiza en

**Análisis de Complejidad**

Para el ciclo de métodoA se comparan todos los subconjuntos de tamaño 2. Esto quiere decir que se realiza:

Dentro del ciclo se ejecuta el método mcdExtendido. Esta complejidad depende de la cantidad de veces que se ejecute la recursión, es decir la cantidad de divisiones que se realicen. El teorema de Lamé nos dice que se hacen a lo sumo [1] esto quiere decir que es lineal con respecto al número de dígitos del número con menos. Si tomamos k como el numero con mayor cantidad de dígitos de la lista de valores de entrada, entonces el peor caso seria . Si nos limitamos a los casos de prueba sabemos que los valores de k están entre entonces el mayor número de dígitos está acotado por l. En conclusión,

En términos del espacio, se crea el arreglo donde se retorna la respuesta exclusivamente. Entonces,

**Comentarios finales**

Utilizando los casos de prueba suministrados en el documento y otros casos planteados por nosotros, nuestra solución demora tiempos satisfactorios en resolver el problema. La decisión más importante que se tomo fue la de cual algoritmo usar para calcular el mcd de los pares de números. Se escogió el algoritmo de Euclides, utilizando divisiones en vez de restas, ya que permitía calcular al tiempo los valores que formaban la combinación lineal. No se logró encontrar una manera para reducir la cantidad de conjuntos que se debían revisar, excepto los eliminados por la propiedad conmutativa del mcd. Efectivamente lo que pesa en el costo temporal es la cantidad de subconjuntos a revisar y si existe una posibilidad de optimización debe estar en esa parte del algoritmo.

**Referencias**

[1] <https://en.wikipedia.org/wiki/Euclidean_algorithm>