UBA – Facultad de Ciencias Exactas y Naturales – Departamento de Computación

Algoritmos y Estructura de Datos I

Segundo cuatrimestre de 2017

28 de Agosto de 2017

TPE - "La sala de reuniones"

1. INTRODUCCIÓN

En la empresa AlgoSRL son muchos empleados y se realizan reuniones periódicamente, mantener el orden y hacer una minuta de la reunión suele ser una tarea difícil. La empresa nos encargó hacer un sistema que permita a través de las voces grabadas de las personas detectar ciertas características de las mismas y también poder escuchar lo que se habló a diferentes velocidades. Para que luego ellos en base a esta información puedan armar la minuta, detectar cuales son las personas que gritan, quienes nunca participan y quienes interrumpen.

2. DETALLES TÉCNICOS

Los problemas que vamos a resolver estan relacionados con el procesamiento del habla. En particular, problemas relacionados a cómo procesar información proveniente de micrófonos en una sala de reuniones. Un micrófono, es un dispositivo electrónico que captura fluctuaciones de presión en el aire. Luego, estos datos se almacenan (de forma analógica o digital) para poder procesarlos.

Una **señal analógica** es una señal que varía de forma continua a lo largo del tiempo. Una **señal digital** es aquella que presenta una variación discontinua con el tiempo y que sólo puede tomar ciertos valores discretos.

Para este trabajo, supondremos que registramos de manera digital las señales captadas por micrófonos.

Estas señales digitales serán almacenadas en forma de secuencia de números enteros (**AMPLITUDES**) que varían en un cierto umbral positivo y negativo dependiendo de la **PROFUNDIDAD** (medido en bits). La cantidad de bits, determina la cantidad de niveles (números posibles) que puede registrarse. Los valores posibles estarán definidos en el rango $[-2^{(P-1)}$, $2^{(P-1)} - 1\ ]$ donde P se refiere a la profundidad.

Ejemplos:

Profundidad: 8 bits

- **Niveles**: 256
- **Amplitud** (P = 8): $[-2^7$, $2^7 - 1\ ]$ = [-128, 127]

Profundidad: 16 bits

- **Niveles**: 65.536
- **Amplitud** (P = 16): $[-2^{15}$, $2^{15} - 1\ ]$ = [-32768, 32767]

Comúnmente, la profundidad suele ser de 8 bits. Ejemplo visualización señal tomada a 4 bits:
Profundidad: 4 bits

- **Niveles**: 8
- **Amplitud** (P = 4): $[-2^3 , 2^3 - 1]$ = [-8, 7]

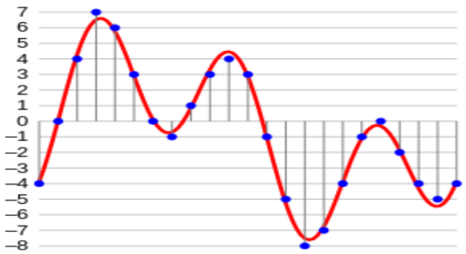

Figura 1: ejemplo señal de 8 bits

Por otra parte, la cantidad de muestras por segundo que registramos se conoce como FRECUENCIA DE MUESTREO y se mide en Hertz (muestras por segundo). Comúnmente la frecuencia de muestreo varía entre 8.000 Hz (teléfonos por ejemplo) y hasta 96.000 Hz (calidad DVDs).

3. REPRESENTACIÓN

Para este trabajo, consideraremos el caso en el que un número fijo de micrófonos se encuentran en una sala de reuniones y comienzan a grabar en simultáneo por un tiempo dado y luego se apagan en el mismo momento. Cada micrófono registra sonidos que serán almacenados en secuencias de números enteros.

Por lo tanto, contaremos con una lista de secuencias $M : seq\langle seq\langle \mathbb{Z} \rangle\rangle$ en donde cada secuencia será la que contenga la información de cada uno de los micrófonos. En este trabajo supondremos que cada micrófono sólo capta el sonido de la persona que lo utiliza y no el sonido de otras personas en la sala (caso skype con auriculares por ejemplo).

4. EJERCICIOS

4.1. Dados los renombres de tipos:

type amplitud = $\mathbb{Z}$

type señal = $seq\langle amplitud \rangle$

type tiempo = $\mathbb{R}$ (medido en segundos)

type intervalo = (tiempo x tiempo)

4.2. Especificar los siguientes problemas:

1. **proc grabaciónVálida(in s: *señal*, in prof : $\mathbb{Z}$, in freq: $\mathbb{Z}$, out result :** Bool):
   - Que dada una señal, una frecuencia de muestreo y una profundidad compruebe que:
     a) Los números están en rango según la profundidad p.
     b) La duración de la señal es mayor a 10 segundos.
     c) El micrófono funciona: No existe una cadena de 0s de longitud mayor a 1 segundo.
     d) La profundidad es 16 o 32 bits.
     e) La frecuencia de muestreo es 44100 ó 48000 Hz
2. **proc elAcaparador(in m:** $seq\langle$***señal***$\rangle$**, in freq: $\mathbb{Z}$, in prof : $\mathbb{Z}$, out persona: $\mathbb{Z}$)**: Que determina quién fue la única persona que gritó más que todo el resto teniendo en cuenta la intensidad media de cada hablante en toda su señal. La intensidad media de un intervalo se calcula como el promedio del valor absoluto de las amplitudes de la señal en los puntos contenidos en ese intervalo. Suponer que las personas están numeradas de 1 a len(m) en el orden en que se recibe en m.

3. **proc reacomodar(inout m: $seq\langle$*señal*$\rangle$, in freq: $\mathbb{Z}$, in prof : $\mathbb{Z}$)**: Que reordena las señales según su intensidad media (de menor a mayor).

4. **proc calmateJosé!(in s: *señal*, in umbralHulk: amplitud, in prof : $\mathbb{Z}$, in freq: $\mathbb{Z}$, out result:** Bool**)**: que indica si hay una persona que se encontraba calmada desde el principio, luego se enojó y finalmente volvió a calmarse hasta el final de la conversación. Definiremos que alguien está enojado si la intensidad media en al menos 1 segundo sobrepasa el umbral umbralHulk.

5. **proc ardillizar(inout m: $seq\langle$*señal*$\rangle$, in prof: $\mathbb{Z}$, in freq: $\mathbb{Z}$)**: Que modifica las señales para que hablen el doble de rápido. Para ello, de cada serie deberemos quedarnos con la mitad de muestras (las muestras en posiciones pares). Tener en cuenta que la señal resultante debe ser válida.

6. **proc flashElPerezoso(inout m: $seq\langle$*señal*$\rangle$, in prof: $\mathbb{Z}$, in freq: $\mathbb{Z}$)**: Que modifica las señales para que hablen el doble de lento. Para ello se debe interpolar las señales. Interpolar una señal en este caso significa crear muestras ficticias entre medio de las tomadas en la señal. Para ello, por cada par de puntos, se agregará en medio el promedio de los dos puntos vecinos.

7. **proc silencios(in s: *señal*, in prof: $\mathbb{Z}$, in freq: $\mathbb{Z}$, in umbral: amplitud, out tiempos: $seq\langle intervalo\rangle$)**: Que dada una serie determina todos los inicios y finales de silencios. Un silencio estará definido por los momentos en el que el valor absoluto de la señal no pasa cierto umbral por al menos 0.1 segundos. El procedimiento devuelve pares de tiempo inicio (t_i) y tiempo final (t_f) en donde en cada par (t_i, t_f) se define el comienzo y fin de todos los silencios de la señal. Es importante aclarar que antes del inicio y después del final de los silencios encontrados, no puede seguir habiendo silencio.

8. **proc hayQuilombo(in m: $seq\langle$*señal*$\rangle$, in prof: $\mathbb{Z}$, in freq: $\mathbb{Z}$, in umbral: amplitud, out result:** Bool**)**: Que dice si en algún momento hay más de una persona hablando (tomando en cuenta la definición de silencio del punto anterior).