Clase #25 de 27
Poscionamiento en Streams & Ejercicio Intgrador

Oct 26, Jueves

Posicionamiento en Streams

Tell y Seek

Posicionamiento en Arrays y Streams

- Array
  - Matemática
    - Escribir
      - Set(a,i,v)=a'=(a_o, ..., a_{i-1}, v, a_{i+1},... ,a_n)
    - Leer
      - a_i=Get(a,i)=v
  - C++
    - Leer
      - a.at(i)
    - Escribir
      - a.at(i) = v
- Streams
  - Escribir: put, write, print, <<
    - Implica escribir valor y avanzar indicador de posicionamiento de escrigura
  - Leer: get, read, scan, >>
    - Implica ller valor y avanzar indicador de poscionamentio de lectura

- Diferencias
  - En arrays leer (escribir) no avanza el indicador de posicionamiento de lectura (escritura), en streams sí
  - Las lecturas, escrituras, y poscionamientos en streams son cinco órdenes de magnitud más costosas en tiempo coparadas con array [https://liw.fi/magnitudes/
  -]; la memoria secundaria es más lenta que la memoria principal
- Nuevas operaciones
  - **Obtener** la posoción actual de lectura (escritura)
  - **Mover** el indicador de posicionamiento de lectura (escritura) sin leer (escribir).

C++: tell y seek

- Indicador de posición de lectura ("get")
  - Obtener
    - s.**tellg**()
  - Establecer
    - Absoluta
      - s.**seekg**(pos)
    - Relativa
      - s.**seekg**(offset, base)
        - base: beg, cur, end
- Indicador de posición de escritura ("put")
  - Obtener
    - s.**tellp**()
  - Establecer
    - Absoluta
      - s.**seekp**(pos)
    - Relativa
      - s.**seekp**(offset, base)
        - base: beg, cur, end

C: fseek & ftell, fgetpos & fsetpos

- En C++ tellg, seekg, tellp, y seekp permiten manejar caracteres multibyte, porque esa una carácterística del stream, no de las funciones.
- C solo tiene un indicador de posicionamiento que se utiliza para leer o escribir en función del modo de apertura del stream.
- En C hoy dos pares de funciones, pero no porque hay dos indicadores de posicionamiento como en C++, si no, porque un par permite manejar caracteres multibyte y el otro par no.

- Clásicas, no multibytes
  - Obtener
    - long **ftell**(stream)
  - Establecer
    - **fseek**(stream,offset,base)
      - long offset
      - int base
        - SEEK_SET
        - SEEK_CUR
        - SEEK_END
- "Nuevas" mutlibyte
  - Obtener
    - **fgetpos**(stream, &pos)
  - Establecer
    - **fsetpos**(stream, &pos)

Ejemplo de Tell y Seek

https://josemariasola.wordpress.com/aed/papers/#TellingAndSeeking

Problemas resolubles con posicionamiento en streams: ***Ordenamiento***

- Set de Problemas ordenamiento de películas 4K Ultra HD
  - Problema
    - Se necesita invertir el orden de un archivo con una cantidad no acotada de películas Ultra HD Blu-Ray
    - Asuma que en disco cada película ocupa lo mismo: el máximo para una película 4K Ultra HD, es decir 100 GB.
    - Cada película se almance en bloques de tamaño también constante: 64 KB
    - Solución
      - ¿Se necesita un archivo exuliar temporal?
      - ¿Cuánto espacio en disco?
      - ¿Cómo copiar una película de un archivo a otro?
      - ¿Es posible armar una solución con stack? Justifique
      - ¿Es posible armar una solución sin stack? Justifique

Read
Push positions
Pop positions
Write

  - Variante #2 del Problema
    - Se necesita ordenarlas por duración ascendente
  - Variante #3 del Problema
    - Esta vez con películas de diferente tamaño
    - Solución
      - ¿Cómo almacenar películas de tamaño diferente de forma eficiente?

Problemas resolubles con posicionamiento en streams: ***Búsqueda***

- Set de problemas <u>búsqueda</u> de películas 4K Ultra HD
  - Problema
    - Se necesita reproducir la película asociada a un código, desde un archivo con una cantidad máxima de N de películas Ultra HD Blu-Ray
    - Cada película se almance en bloques de tamaño también constante: 64 KB
    - Solución
      - ¿Qué estructura y algoritmo aplicar?
  - Variante del Problema
    - Esta vez no hay cota en la cantidad de películas en el archivo
    - Solución
      - ¿Cómo almacenar los códigos y posciones de maner que permita una búsqueda eficiente y permita una canitdad no acotada de películas? **Árbol binario de búsqueda**.

Ejercicio Integrador

Streams, Structs, Arrays, Punteros, reserva dinámica de memoria, funcones

Ordenamiento y Búsqueda de Círculos

- Problema
  - Se requiere ordenar un conjunto de círculos por superficie, y buscar un círculo por código. Un círculo tienen un centro, un radio, un código, un color, y una descripción de hasta N caracteres.
- Solución: dos programas
  - Programa #1 que lee de cin círculos que poseean un código identificador numérico y los guarda en un archivo de bloques de longitud constante
  - Programa #2 con dos funcionalidades:
    - Ordenamiento
      - Leer el archivo de círculos
        - Almacenarlos en una lista enlazada, ordenados por superficie.
          - Construir la función GetSuperficie
          - Como los círculos son una estructura compleja, se decide no almacenarlos en memodira, solo su posición en la lista y a la superficie.
        - Almacenar el código y el puntero de cada círuclo en el arreglo, se asume que hay como máximo MAX círculos, puede haber menos
      - Los muestre por cout ordenados por superficie
    - Búsqueda
      - Ordenar el arreglo por código
      - Leer un código, buscarlo en el arreglo, utilizar método binario
      - Si lo encuentro, muestrarlo por cout.