Universidad Andres Bello  
Facultad de Ingeniería  
Ingeniería en Computación e Informática

Informe de proyecto final:

Estructuras de datos

Alumnos:

* Rodrigo Gonzalez A.
* Jorge Sepúlveda S.

Profesor: Pamela Landero S.  
Ayudante: Tomas Lara  
Asignatura: INS127 – Estructuras de datos  
Fecha: 22-11-2017

1. **RESUMEN DE ESTRATEGIA**

Principalmente, la estrategia utilizada consistió en lo siguiente:

* 1. **Análisis del problema:**

Se procedió a realizar un análisis del problema fundamental, es decir, el cálculo de las operaciones de los ciclos de vida de una bacteria, para ello, se dividió el problema en los siguientes puntos:

* + 1. Análisis de condiciones de vida y muerte: Según lo descrito
    2. Análisis de vecinos de la bacteria: Utilizando axiomas simples, en los cuales, considerando una bacteria en la coordenada (0,0) sus vecinos están representados por la notación (+-X,+-Y) referentes al punto (0,0) del plano. Las operaciones están descritas en el archivo Vecinos.h a modo de comentario.
    3. Análisis de los ciclos de vida. En donde se utilizó.
    4. Análisis de las estructuras principales a utilizar: Coor, Instrucciones y más.
  1. **Diseño**

Se procedió a diseñar los algoritmos necesarios para dar las condiciones del juego de manera tal que cumplieran con lo solicitado.

Los algoritmos más relevantes tienen relación con el cálculo de vecinos, ejecución de ciclos de vida y la Pila de tareas a ejecutar cuando se ejecutan los ciclos de vida.

Una operación completa que considere un ciclo de vida de una matriz de 40 x 40 con 10 Ciclos de Vida, toma 128.000 operaciones.

* 1. **Implementación**

Para implementar los algoritmos de acuerdo a los análisis descritos anteriormente, se procedió a utilizar funciones de manera aisladas que en conjunto resolvieran el problema dando un enfoque simple de entender y fácil de corregir.

* 1. La estructura del programa principal es la siguiente:

1. **ESTRUCTURAS DE DATOS UTILIZADAS**

El proyecto consiste en el uso de varias librerías creadas con la finalidad de definir en ellas, tanto funciones como estructuras de datos que fueran a ser utilizadas para lograr lo solicitado en el laboratorio.

Las siguientes librerías fueron utilizadas en el proyecto:

* 1. Vecinos.h
     1. Descripción: Esta librería define una estructura de datos llamada Coor, que contiene las coordenadas de los nodos (bacterias).
     2. Estructuras consideradas: Coor: Esta estructura contiene las coordenadas de las bacterias.
     3. Funciones: Existe una función por cada vecino a calcular, un total de 8 funciones controlan todos los vecinos de una bacteria en particular.
  2. Pila.h
     1. Descripción: Esta librería define estructuras de datos y funciones que son utilizadas en el programa.
     2. Estructuras consideradas:
        1. Nodo: Esta estructura considera 3 atributos: Coord Coordenadas, int acción (estado 1 vive o 0 muere) y un puntero al nodo siguiente.
        2. Pila: Esta estructura considera 2 atributos: un puntero al último nodo y una variable de tipo int size.
     3. Funciones consideradas:
        1. crearPila(): Esta función crea una pila, asigna memoria, valida si está vacía de lo contrario arroja un mensaje al no poder asignar memoria, asigna un largo y apunta el puntero del nodo a NULL.
        2. Push(): Esta función apila los nodos en la pila creada, asigna las coordenadas a los nodos (bacterias).
        3. Top(): Esta función retorna el nodo tope que se encuentra en la pila.
        4. Pop(): Esta función va des-apilando los nodos de la pila y asignando datos a los punteros de la pila.
  3. Ciclos.h
     1. Descripción: Esta librería controla las operaciones relacionadas con la ejecución de ciclos de vida del juego.
     2. Estructuras consideradas: Coor {x,y}
     3. Funciones consideradas:
        1. CalcularVecinosVivos()
        2. CiclodeVida()
        3. BacteriasVivas()
  4. ManejoDeArchivos.h
     1. Descripción: Librería encargada de controlar los aspectos necesarios para el uso de archivos dentro del programa.
     2. Estructuras consideradas:
        1. DescriptorBacterias: Esta estructura consideró 4 atributos: número mágico, N para cantidad de filas de la matriz, M para la cantidad de columnas de la matriz y cantidad para la cantidad de bacterias existentes.
        2. Coordenadas: Esta estructura contiene las coordenadas de alas bacterias.
     3. Funciones consideradas:
        1. InforArchivo()
        2. GuardarArchivo(): Esta funcion guarda el archivo binario, con los resultados del programa.
        3. Manejador(): Esta función maneja el archivo de origen (bacterias.bin), inicializa la matriz vacía para albergar los elementos que se leerán de las instrucciones de creación

1. **VALIDACIONES**

Parte de las validaciones del código se realizaron en 2 tipos de laptop personales:

* 1. macOS:
     1. IDE: Clion 2017.2.3 de JetBrains
     2. Compilador: GCC Apple LLVM version 9.0.0 (clang-900.0.38), Target: x86\_64-Apple-darwin16.7.0, Thread model: posix
  2. Windows 10
     1. IDE: Codeblock 16.01
     2. Compilador: gcc version 5.3.0 (GCC) - CygWin