Universidad de San Carlos de Guatemala Facultad de Ingeniería

Curso: Lenguajes Formales y de Programación



Erik Vladimir Girón Márquez Carnet # 200313492 Sección A+

Guatemala, 4 de Noviembre de 2005

# Introducción:

Conjuntivity es un intérprete para operaciones de conjuntos, implementando analisis léxico y sintáctico sobre el código escrito por el usuario y al mismo tiempo generando los resultados para las operaciones especificadas en el código.

La aplicación implementa las operaciones de union, interseccion, diferencia, diferencia simétrica, complemento y Contención, aplicando éstas a conjuntos previamente definidos en un archivo ASCII o UNICODE fuente previamente seleccionado. Además cuenta con un editor de texto básico desde el que se podrá modificar el código fuente seleccionado e interpretarlo en tiempo de edicion.

El fin de este documento es guiar paso a paso al usuario a través de los diferentes comandos del programa, sin embargo la interfaz es tán simple que permite un fácil manejo de los comandos del programa, ya sea desde el ratón, o utilizando teclas de acceso rápido.

El programa está desarrollado en el lenguaje Visual Basic .NET compilado bajo Visual Basic .net, para ser ejecutado en la plataforma .net Framework ver 1.0.3705 + o en plataformas compatibles a .net (Mono, etc).

# <u>Índice:</u>

INTRODUCCIÓN:	2
ÍNDICE:	3
REQUISITOS DEL SISTEMA:	4
INSTRUCCIONES DE USO:	4
PRE-PASOS: PASOS:	4
SINTAXIS DEL ARCHIVO DE ENTRADA.	5
Gramática Informal: Archivo de Entrada de Ejemplo:	5
COMANDOS DEL MENU:	6

## Requisitos del Sistema:

- Ordenador con tarjeta gráfica, Unidad de CD-ROM y Ratón Funcionales.
- Sistema operativo que soporte el .net Framework (por su relativa estabilidad, se recomienda Windows 2000 SP4+ o Windows XP SP2+, o preferiblemente cualquier sistema operativo UNIX sobre los anteriores si se dispone de plataforma .net para este tipo de sistemas, NO SE RECOMIENDA EN ABSOLUTO WINDOWS 98 O ME).
- .net Framework (disponible en <a href="http://www.microsoft.com">http://www.microsoft.com</a>) ver 1.0 o mayor correctamente instalado.

(NOTA: Asegúrese de cumplir los requisitos del sistema, o el programa no correrá como debería.)

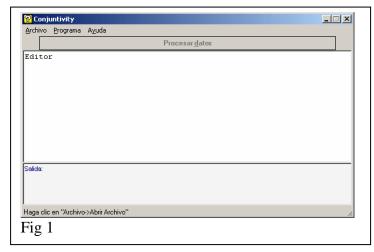
## Instrucciones de Uso:

### Pre-Pasos:

- Encienda su computador y verifique que cumple con los requisitos del sistema.
- Ingrese a su sistema operativo y verifique que tiene instalado el .net Framework.

#### Pasos:

- 1. Inserte en la unidad de CD-ROM, el disco del programa.
- 2. Si no se autoejecuta el programa diríjase a *Unidad de CD/DVD-ROM* desde *mi PC* y haga doble clic en ésta.
- 3. Abra el directorio bin y haga doble clic en conjuntivity.exe.
- 4. Si la plataforma *.net Framework* está correctamente instalado, inmediatamente se mostrará la ventana principal (figura 1).
- 5. Dentro de esta ventana deberá primero seleccionar un archivo previamente editado desde el menu *Archivo-Abrir archivo*.
- 6. Luego de que el archivo haya sido correctamente cargado, podrá hacer clic en el botón Procesar Datos, o presionar F9 desde el teclado.
- Luego de procesados los datos podra ver la salida en el cuadro inferior de la ventana, en donde saldrán los resultados o errores de interpretación.
- 8. Podrá modificar en cualquier momento el archivo de entrada y luego guardarlo al presionar CTRL+S.
- 9. Si desea Salir del programa simplemente cierre la ventana o presione CTRL+W.



## Sintaxis del archivo de Entrada.

### Gramática Informal:

```
La sintaxis es la siguiente:
```

```
programa [nombre]
conjuntos
         (definicion_de_conjuntos)
inicio
          (asignacion_de_conjuntos) | (bloque_si)
Resultados
         Salida (pantalla|archivo|impresora)
         Escribir([conjunto1],[conjunto2],...,[conjuntoN])
          :
Fin
Donde:
(bloque_si):
         si [conjunto|elem]@[conjunto] entonces
          (asignacion_de_conjuntos)
          [sino (asignacion_de_conjuntos) ] ?
         Fin si;
 (definicion_de_conjuntos):
          [conjuntoA] = \{ [elem1A], [elem2A], ..., [elemNA] \}
          [conjuntoM] = \{ [elem1M], [elem2M], ..., [elemNM] \}
(asignacion_de_conjuntos):
          ([conjuntoA] = [conjunto1] [+|-|*|=][conjunto2] ...);
Siendo:
?:optativo
Texto en negrita: Palabras Reservadas
Texto entre corchetes e Itálica: Identificadores
Pueden incluirse comentarios que serán ignorados por el intérprete entre bloques
/*comentario*/
```

Nota: Para una version mas formal de la gramática(en formato Backus-Naur), revisar manual Técnico

### Archivo de Entrada de Ejemplo:

```
programa ejemplo /*Nombre del programa*/
conjuntos /*Lista de Conjuntos*/
           conjunto1 = {uno, dos, tres, 4}/*conjunto 1*/
           c2 = \{\} /*conjunto 2*/
           c3 = \{uno, dos\} / *conjunto 3*/
inicio /*Bloque Inicio*/
           c2 = conjunto1 + c3;
           si c3 % c2 /*Si c3 contenido en c2*/
           entonces
                      c2 = conjunto1;
                      c3 = conjunto1;
           sino
c2 = c3;
           fin si;
resultados /*Bloque resultados*/
           salida pantalla /*Salida a pantalla*/
           escribir (conjunto1,c2,conjunto1)
           escribir (conjunto1)
           escribir (C3, U)
fin /*fin*/
```

# Comandos del Menu:

