Entornos de Desarrollo (ED)
Unidad de Trabajo: UT1
Tarea 03: Análisis de la traducción de un programa en C

Objetivos:

- Comparar los distintos tipos de traducción.
- Escribir, compilar y ejecutar un programa sencillo sin la ayuda de un entorno de desarrollo.

Descripción de la tarea:

En esta tarea, aprenderás a analizar y documentar el proceso de compilación de un programa escrito en C, desde la creación del archivo hasta la ejecución del programa final. Este proceso incluye varias fases: **preprocesado**, **compilación**, **ensamblado** y **enlazado**. A lo largo de la tarea, te guiaremos paso a paso en la creación y ejecución de un programa C en un entorno Linux, usando la terminal.

Instrucciones paso a paso:

1. **Crear el archivo fuente en C**
   Lo primero que debes hacer es crear un archivo que contenga el código fuente en C. Para ello, utilizaremos el editor de texto `nano`, que puedes ejecutar desde la terminal. Sigue estos pasos:

   - Abre la terminal de Linux.
   - Para crear un archivo llamado `main.c`, escribe el siguiente comando y presiona **Enter**:

     ```
     nano main.c
     ```

   - Dentro del editor de texto `nano`, escribe el siguiente código:

     ```c
     #include <stdio.h>
     #define MESSAGE "¡Hola estudiantes de DAM!\n"

     //los comentarios serán borrados en el preprocesado
     //Aquí comienza el programa en sí
     int main() {
         printf(MESSAGE);
         return 0;
     }
     //Aquí finaliza el programa
     ```

   - Para guardar el archivo y salir de `nano`, presiona **Ctrl + O** (luego Enter para confirmar el nombre del archivo), y después **Ctrl + X** para cerrar el editor.

2. **Cambiar los permisos del archivo para que sea ejecutable**
   En Linux, los archivos creados no son ejecutables de forma predeterminada. Antes de compilar y

ejecutar el programa, deberás cambiar los permisos del archivo. En la terminal, escribe el siguiente comando para asegurarte de que tengas permisos de lectura y escritura:

```
chmod u+rwx main.c
```

3. **Preprocesado**
   Ahora que tienes el archivo fuente, el primer paso del proceso de compilación es el **preprocesado**. El preprocesador en C maneja directivas como `#include` y `#define`. Para realizar este paso, utiliza el siguiente comando en la terminal:

   ```
   gcc -E main.c -o main.i
   ```

   Este comando genera el archivo `main.i`, que contiene el código preprocesado. Puedes abrirlo con el editor `nano` para ver cómo las macros y las cabeceras han sido reemplazadas:

   ```
   nano main.i
   ```

4. **Compilación**
   El siguiente paso es compilar el código preprocesado para convertirlo en código ensamblador. Ejecuta el siguiente comando para generar un archivo llamado `main.s`:

   ```
   gcc -S main.i -o main.s
   ```

   Abre el archivo ensamblador con `nano` para ver el resultado:

   ```
   nano main.s
   ```

   En este archivo podrás observar instrucciones en lenguaje ensamblador, que son más cercanas a las instrucciones que entiende el procesador.

5. **Ensamblado**
   El código ensamblador ahora debe ser traducido a un archivo objeto (binario intermedio) que el sistema pueda manejar. Usa el siguiente comando para generar el archivo `main.o`:

   ```
   gcc -c main.s -o main.o
   ```

   El archivo `main.o` es un archivo objeto que contiene el código en formato binario. Aunque no es necesario abrir este archivo, puedes verificar que se ha generado correctamente utilizando el siguiente comando:

```
ls -l main.o
```

También puedes investigar cómo mostrar el contenido en binario.

6. **Enlazado**
   El último paso es enlazar el archivo objeto con las bibliotecas necesarias para crear el ejecutable. Usa el siguiente comando:

   ```
   gcc main.o -o main
   ```

   Esto generará el ejecutable `main`. Sin embargo, si no puedes ejecutarlo directamente, deberás asegurarte de que tiene los permisos correctos. Cambia los permisos del ejecutable utilizando:

   ```
   chmod u+x main
   ```

7. **Ejecutar el programa**
   Una vez generado el ejecutable, puedes correr el programa usando el siguiente comando:

   ```
   ./main
   ```

   Si todo ha salido bien, verás que el programa imprime el mensaje **Hello, World!** en la terminal.

Instrucciones de entrega:

La entrega debe consistir en un documento en formato PDF con el análisis de cada una de las fases mencionadas. Incrusta capturas de pantalla del contenido de cada archivo y compara y comenta su contenido.

Además, debes incluir los archivos generados en cada paso del proceso:

- Código fuente (`main.c`)
- Archivo preprocesado (`main.i`)
- Código ensamblador (`main.s`)
- Archivo objeto (`main.o`)
- Ejecutable final (`main`)

Rúbrica

Calificación	Descripción
0	No realiza ninguna de las etapas solicitadas o entrega incompleta.
3	Realiza las etapas de manera parcial, sin análisis detallado.
6	Completa todas las etapas, pero el análisis es superficial o incompleto.

Calificación	Descripción
9	Realiza todas las etapas con un análisis detallado y documentación clara.
10	Además de lo solicitado, incluye detalles extras, como optimización o análisis comparativo con otro lenguaje.