



**Universidad Nacional Autónoma
de México
Facultad de Ingeniería**



Asignatura: Estructura de Datos y Algoritmos 1

Actividad 1: Repaso

Alumna: Hernández Vázquez Daniela

Profesor: M.I. Marco Antonio Martínez Quintana

**Fecha: 07/06/2021
2021-2**



Actividades:

- Repasar sus conocimientos de Lenguaje C adquiridos en Fundamentos de Programación y el primer tema de arreglos en C, de Estructura de Datos y Algoritmos I

Para esta actividad no tuve que hacer demasiado debido que ya tenía un acordeón de lenguaje C que nos había pedido anteriormente como actividad asíncrona, este lo tomé como referencia principal sumado a apuntes tomados en clase.

Control de versiones: Registro de cambios Local, Centralizado, Distribuido	Repositorio: Directorio de trabajo Local, Remoto, Nube	Sistema operativo: Conjunto de programas y datos que administra recursos de hardware y software de un equipo de computo y/o comunicación
Linux: SO tipo Unix de libre distribución Núcleo (kernel), programas y bibliotecas.	Comandos básicos (Linux): Terminal <ul style="list-style-type: none">- ls lista elementos- touch crea archivos de texto <code>touch nombre_archivo[.ext]</code>- mkdir crea una carpeta <code>mkdir nombre_carpeta</code>- cd cambio de directorio- pwd conocer ruta- find <code>find . -name cadena_buscar</code>- clear- cp copy <code>cp archivo_origen archivo_destino</code>- mv mv ubicación_origen/archivo ubicación_destino- rm eliminar <code>rm nombre_archivo</code> Ó <code>rm nombre_carpeta</code>	Ciclo de vida del software: <ul style="list-style-type: none">- Definición de necesidades.- Análisis [datos de entrada y salida]- Diseño de estructura de datos, arquitectura del programa y el procedimiento algorítmico.- Codificación.- Pruebas, Mantenimiento y evolución.
Algoritmo: Serie de pasos ordenados para dar solución a un problema.	Diagramas de flujo: Representación grafica de un algoritmo	Pseudocódigo: Representación escrita del Algoritmo.
Estructuras de control de flujo {secuencial (orden) {condicional (if-else) {repetitivas o iterativas(cíclicas, do-do while)		

Funciones: Subprocesos

Identificador (nombre) y un valor de retorno.

Editores (de texto), compilación y ejecución. Lenguaje C, de alto nivel.

Símbolo del sistema:

`gcc calculadora.c -o calculadofra.exe`
`Calculadora.exe`

- Comentarios `{// ó /* →*/}`
- Declaración de variables y tipos de datos (Signed [-], unsigned[+];{char, short, int, long, enum}): [modificadores] tipoDeDato identificador [= valor];
 - Caracteres (char): codificación definida or la maquina (`%c`, `%d`, `%i`, `%o`, `%x`)
 - Cadena de caracteres (`%s`)
 - Ennteros (`()`): números sin punto decimal (`%d`, `%i`, `%ld`, `%o`, `%x`)
 - Flotantes(float): números reales de precisión normal (`%f`, `%lf`, `%e`, `%g`)
 - Dobles (double): números reales de doble precisión
- Almacenar e imprimir variables
 - `printf("El valor de la variable real es: %lf", varReal);`
 - `scanf ("%i", &varEntera);`

`\a` (alarma), `\b` (retroceso), `\f` (avance de hoja), `\n` (salto de línea), `\r` (regreso de carro), `\t` tabulador horizontal, `\v` (tabulador vertical), `'\0'` carácter nulo

- Modificadores de alcance
 - Const y static
- Operadores `{+, -, *, /, %, >>, <<, &, |, ~, !, &&, | |}`
- Expresiones lógicas `{==, !=, >, <, >=, <=}`
- Estructuras de selección (if, if-else, switch-case), repetición (while, do, do-while, for), (define, break, continue)

Función main

```
valorRetorno nombre (parámetros){  
    // bloque de código de la función  
}
```

Depuración de programas:

Error. Acción humana.

Defecto (Fault). Error en el software.
Por una Falla (failure).

Falla (failure). Es una desviación del servicio o resultado esperado.

Apuntadores (while, for):

variable que contiene la dirección de una variable.

Trabajan directamente con la memoria, a través de ellos se accede con rapidez a un dato.

TipoDeDato *apuntador, variable;

apuntador = &variable;

Inicia con el carácter *.

Arreglos:

Conjunto de datos contiguos del mismo tipo.

A cada elemento (dato) del arreglo se le asocia una posición particular, el cual se requiere indicar para acceder a un elemento en específico.

Depuración de C con GCC y GDB:

gcc -g -o calculadora calculadora.c
gdb ./calculadora

- *list* o *l*: listar Ej: *list 4,6*
- *b*: punto de ruptura Ej: *b 5*
- *d* o *delete*: Elimina un break point. Ej: *d 5*
- *clear*: Elimina todos break point. Ej: *clear*
- *info line*: Ej: *info line 8*
- *run* o *r*: Ejecuta el programa
- *c*: Continúa con la ejecución del programa después de un punto de ruptura.
- *s*: continua con la siguiente instrucción después de un break point.
- *n*: salta a la siguiente línea de código después de un break point.
- *p* o *print*: Muestra el valor de una variable, Ej: *p suma_acumulada*
- *ignore*: Ignora un break point. Ej: *ignore 5*
- *q* o *quit*: Termina la ejecución de GDB.

Arreglos unidimensionales(while, for)

tipoDeDato nombre[tamaño]

Arreglos multidimensionales(while, for)

tipoDeDato nombre[tamaño] ...[tamaño]

Lectura y escritura de datos: FILE *F

*FILE fopen(char *nombre_archivo, char *modo);

- r: Abre un archivo de texto para lectura.
- w: Crea un archivo de texto para escritura.
- a: Abre un archivo de texto para añadir.
- r+: Abre un archivo de texto para lectura / escritura.
- w+: Crea un archivo de texto para lectura / escritura.
- a+: Añade o crea un archivo de texto para lectura / escritura.
- rb: Abre un archivo en modo lectura y binario.
- wb: Crea un archivo en modo escritura y binario.

int fclose(FILE *apArch);

char *fgets(char *buffer, int tamaño, FILE *apArch);

char *fputs(char *buffer, FILE *apArch);

int fprintf(FILE *apArch, char *formato, ...);

int fscanf(FILE *apArch, char *formato, ...);

int fread(void *ap, size_t tam, size_t nelem, FILE *archivo)

int fwrite(void *ap, size_t tam, size_t nelem, FILE *archivo)