



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



Asignatura: Estructura de Datos y Algoritmos 1

**Actividad 1: Repaso de lo que aprendí en la asignatura de
Fundamentos de Programación.**

Alumna: Hernández Vázquez Daniela

Profesor: M.I. Marco Antonio Martínez Quintana

Fecha: 24/02/2021

2021-2



Actividades:

- *Realizar un escrito de 5 cuartillas recopilando lo aprendido dentro del semestre pasado (se pueden incluir imágenes).*

Las primeras actividades que realizamos fueron principalmente para conocer acerca de la historia de las computadoras y de los antecedentes históricos de lo que hoy conocemos como computación y comenzar a verla como una herramienta de trabajo profesional que debemos aplicar en la ingeniería y que resulta sumamente útil.

También aprendimos acerca de la Licencia de software libre, aquel que se puede adquirir de manera gratuita, La Licencia Pública General de GNU o GNU General Public License (GNU GPL) es una licencia creada por la Free Software Foundation en 1989 y está orientada principalmente a proteger la libre distribución, modificación y uso de software, y decir en qué circunstancias se puede utilizar, modificar y redistribuir sin fines de lucro y buscando siempre la seguridad de los consumidores y de los miembros de la comunidad. Y en la cual cualquier persona interesada puede tener acceso a este tipo archivos, claro, teniendo en cuenta las políticas de privacidad y ética.

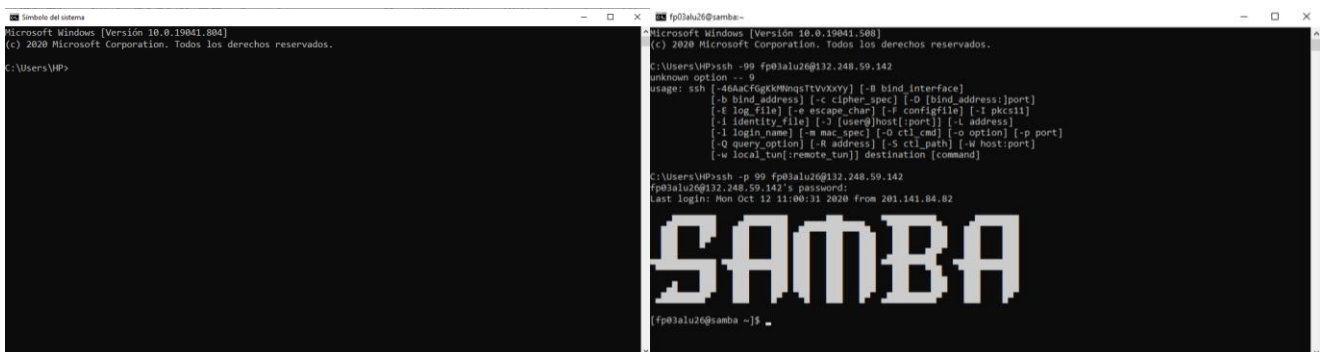
Antes de entrar de lleno con los lenguajes de programación aprendimos conceptos básicos como el control de versiones el cual es sistema el cual lleva a cabo el registro de los cambios sobre uno o más archivos (sin importar el tipo de archivo) a lo largo del tiempo y que a hacer esto nos permiten regresar a versiones específicas de dichos archivos y modificarlos y estos no solo son aplicables para programadores, su uso también se extiende a escritores, diseñadores o cualquiera que busque tener un control sobre sus archivos.

Dentro del curso nos vimos en la necesidad de crear y utilizar nuestro propio repositorio el cual fue nuestro directorio de trabajo en el cual registrábamos nuestro avance dentro de las clases, así como el avance que tuvimos en nuestro proyecto de trabajo final que nos dejó elegir el profesor y que desarrollamos a lo largo de todo el curso.

En este caso relacionaré lo aprendido en el curso y la forma en que apliqué este dentro de las prácticas y programas realizados, así como dentro del desarrollo de mi proyecto titulado FRANKY el cual es un chatbot de respuestas predeterminadas, es decir que no utiliza inteligencia artificial, que programé en lenguaje C.

Luego de ver estos conceptos comenzamos posteriormente a conocer los conceptos básicos para la programación y comenzamos a estudiar y trabajar un poco con el lenguaje C.

Dentro de nuestra segunda práctica trabajamos vía remota con las computadoras de la facultad y trabajamos de modo que el Shell de GNU/Linux nos debía de permitir introducir órdenes (comandos) y ejecutar programas en el sistema operativo. Revisamos los comandos básicos desde nuestra terminal o consola, en mi caso esta nombrada como Símbolo del Sistema la cual es una ventana donde aparece la “línea de comandos” en donde se escribirán estos y realizarán sus funciones correspondientes a cada uno.



```
Microsoft Windows [Versión 10.0.19041.804]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\HP>

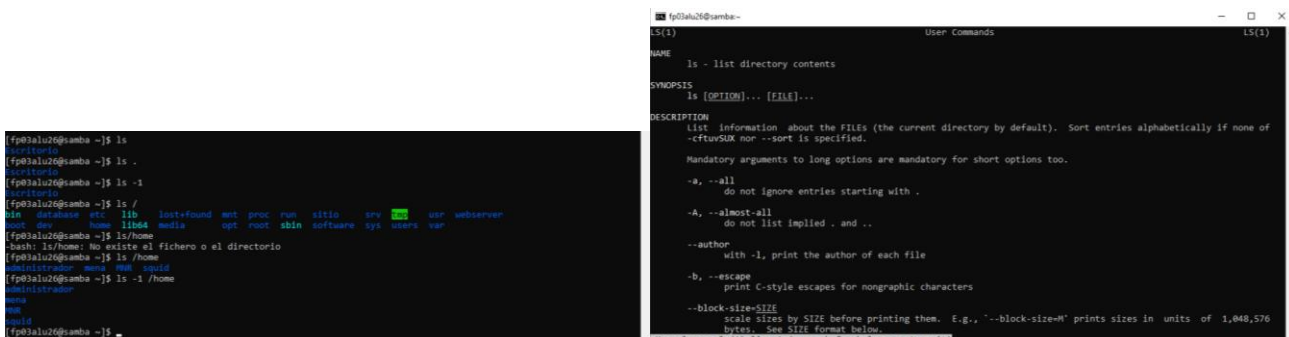
Microsoft Windows [Versión 10.0.19041.500]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\HP>ssh -p 99 fp03alu26@132.248.59.142
Warning: Permanently added 'fp03alu26@132.248.59.142' (ssh) to the list of known hosts.
fp03alu26@132.248.59.142's password:
Last login: Mon Oct 12 11:00:11 2020 from 201.141.84.82

Samba

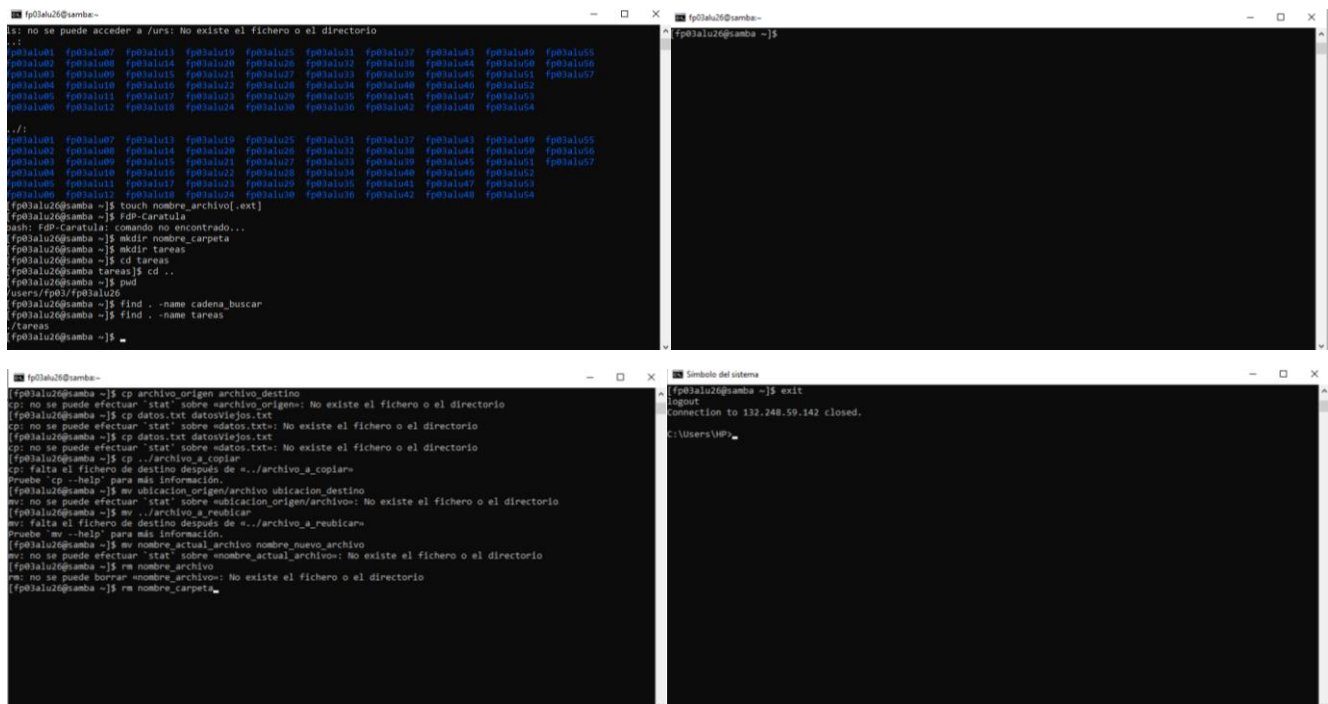
[fp03alu26@samba ~]$
```

Por ejemplo, el comando ls permite listar los elementos dentro de un carpeta u ubicación.



```
fp03alu26@samba ~]$ ls
.
..
autorun
fp03alu26@samba ~]$ ls -l
total 4
-rw-r--r-- 1 fp03alu26 fp03alu26 4096 Oct 12 11:00 autorun
fp03alu26@samba ~]$ ls /
bin  dev  home  lib64  media  opt  root  sbin  software  sys  usr  var
fp03alu26@samba ~]$ ls /home
fp03alu26@samba ~]$ ls /home
ls: /home: no existe el fichero o el directorio
fp03alu26@samba ~]$ ls /home
ls: /home: no existe el fichero o el directorio
fp03alu26@samba ~]$ ls -l /home
total 4
-rw-r--r-- 1 fp03alu26 fp03alu26 4096 Oct 12 11:00 autorun
fp03alu26@samba ~]$
```

El comando touch permite crear un archivo de texto. El comando mkdir crea una carpeta. El comando cd nos ubica en una carpeta. El comando pwd no deja conocer la ubicación actual o ruta. El comando find busca un elemento dentro del sistema de archivos, entre muchos otros, los cuales no mencionaré completamente porque serían demasiado extensos como el comando clear o cls (dependiendo) que nos permite limpiar la pantalla.



```
fp03alu26@samba:~$ ls
ls: no se puede acceder a /urs: No existe el fichero o el directorio
./
fp03alu01 fp03alu07 fp03alu13 fp03alu19 fp03alu25 fp03alu31 fp03alu37 fp03alu43 fp03alu49 fp03alu55
fp03alu02 fp03alu08 fp03alu14 fp03alu20 fp03alu26 fp03alu32 fp03alu38 fp03alu44 fp03alu50 fp03alu56
fp03alu03 fp03alu09 fp03alu15 fp03alu21 fp03alu27 fp03alu33 fp03alu39 fp03alu45 fp03alu51 fp03alu57
fp03alu04 fp03alu10 fp03alu16 fp03alu22 fp03alu28 fp03alu34 fp03alu40 fp03alu46 fp03alu52
fp03alu05 fp03alu11 fp03alu17 fp03alu23 fp03alu29 fp03alu35 fp03alu41 fp03alu47 fp03alu53
fp03alu06 fp03alu12 fp03alu18 fp03alu24 fp03alu30 fp03alu36 fp03alu42 fp03alu48 fp03alu54

./1/
fp03alu01 fp03alu07 fp03alu13 fp03alu19 fp03alu25 fp03alu31 fp03alu37 fp03alu43 fp03alu49 fp03alu55
fp03alu02 fp03alu08 fp03alu14 fp03alu20 fp03alu26 fp03alu32 fp03alu38 fp03alu44 fp03alu50 fp03alu56
fp03alu03 fp03alu09 fp03alu15 fp03alu21 fp03alu27 fp03alu33 fp03alu39 fp03alu45 fp03alu51 fp03alu57
fp03alu04 fp03alu10 fp03alu16 fp03alu22 fp03alu28 fp03alu34 fp03alu40 fp03alu46 fp03alu52
fp03alu05 fp03alu11 fp03alu17 fp03alu23 fp03alu29 fp03alu35 fp03alu41 fp03alu47 fp03alu53
fp03alu06 fp03alu12 fp03alu18 fp03alu24 fp03alu30 fp03alu36 fp03alu42 fp03alu48 fp03alu54

fp03alu26@samba ~$ touch nombre_archivo[.ext]
fp03alu26@samba ~$ fdP-Caratula
bash: fdP-Caratula: comando no encontrado...
fp03alu26@samba ~$ mkdir nombre_carpeta
fp03alu26@samba ~$ mkdir tareas
fp03alu26@samba ~$ cd tareas
fp03alu26@samba tareas$ cd ..
fp03alu26@samba ~$ pwd
/users/fp03/fp03alu26
fp03alu26@samba ~$ find . -name cadena_buscar
fp03alu26@samba ~$ find . -name tareas
./tareas
fp03alu26@samba ~$

fp03alu26@samba ~$ cp archivo_origen archivo_destino
cp: no se puede efectuar 'stat' sobre 'archivo_origen': No existe el fichero o el directorio
fp03alu26@samba ~$ cp datos.txt datosviejos.txt
cp: no se puede efectuar 'stat' sobre 'datos.txt': No existe el fichero o el directorio
fp03alu26@samba ~$ cp datos.txt datosviejos.txt
cp: no se puede efectuar 'stat' sobre 'datos.txt': No existe el fichero o el directorio
fp03alu26@samba ~$ cp ../archivo_a_copiar
cp: falta el fichero de destino después de '../archivo_a_copiar'
Pruebe 'cp --help' para más información.
fp03alu26@samba ~$ mv ubicacion_origen/archivo ubicacion_destino
mv: no se puede efectuar 'stat' sobre 'ubicacion_origen/archivo': No existe el fichero o el directorio
fp03alu26@samba ~$ mv ../archivo_a_reubicar
mv: falta el fichero de destino después de '../archivo_a_reubicar'
Pruebe 'mv --help' para más información.
fp03alu26@samba ~$ mv nombre_actual_archivo nombre_nuevo_archivo
mv: no se puede efectuar 'stat' sobre 'nombre_actual_archivo': No existe el fichero o el directorio
fp03alu26@samba ~$ rm nombre_archivo
rm: no se puede borrar 'nombre_archivo': No existe el fichero o el directorio
fp03alu26@samba ~$ rm nombre_carpeta
```

Ahora bien, conociendo ya los comandos básicos de la terminal ahora comenzamos un trabajo de análisis en el cual buscamos la solución de problemas de programación y conocemos el concepto de “Algoritmo” que mencionaré más adelante.

Para poder solucionar un problema nos apoyamos en la Ingeniería de Software que, de acuerdo a la IEEE, acrónimo del “Institute of Electrical and Electronics Engineers” que es una institución científica de carácter internacional compuesta por técnicos e ingenieros cuyo objetivo es la estandarización y el desarrollo de las áreas técnicas que mejoran las nuevas tecnologías con un enfoque sistemático y disciplinado hacia el desarrollo, operación y mantenimiento del software. Es de suma importancia por lo que el uso y establecimiento de principios de ingeniería sólidos, son

básicos para obtener un software que sea económicamente fiable y funcione eficientemente. Este tema lo tratare también más adelante.

Continuaré por mencionar un tema que me resulto muy interesante, el ciclo de vida de un software. La ISO (International Organization for Standardization) en su norma 12207 define al ciclo de vida de un software como: “Un marco de referencia que contiene las actividades y las tareas involucradas en el desarrollo, la explotación y el mantenimiento de un producto de software, abarcando desde la definición hasta la finalización de su uso”.



Figura 1: Ciclo de vida del software.

Dentro del ciclo de vida del software, en el análisis se busca comprender la necesidad, es decir, entender el problema. En el análisis se busca averiguar qué es lo que requiere el usuario del sistema de software (especificación de requisitos).

Para ello es importante identificar dos grandes conjuntos dentro del sistema: el conjunto de entrada y el conjunto de salida.

- El *conjunto de entrada* está compuesto por todos aquellos datos que pueden alimentar al sistema.
- El *conjunto de salida* está compuesto por todos los datos que el sistema regresará como resultado del proceso. Se obtienen a partir de los datos de entrada.

La unión del conjunto de entrada y el conjunto de salida forman lo que se conoce como el dominio del problema, es decir, los valores que el problema puede manejar. La etapa del análisis es crucial para la creación de un software de calidad, ya que si no se entiende qué es lo que se desea realizar, no se puede generar una solución.

Una vez realizado el análisis, o sea, ya que se entendió qué es lo que está solicitando el usuario y ya identificado el conjunto de entrada y el conjunto de salida, se puede proceder al diseño de la solución o a la generación del *algoritmo*.

Dentro del ciclo de vida del software, la creación de un algoritmo se encuentra en la etapa de diseño.

Y conocemos formalmente a un algoritmo como un conjunto de reglas, expresadas en un lenguaje específico, para realizar alguna tarea en general, o sea, un conjunto de pasos, procedimientos o acciones que permiten alcanzar un resultado o resolver un problema. Estas reglas o pasos pueden ser aplicados un número ilimitado de veces sobre una situación particular.

Este es la parte más importante dentro de las ciencias de la computación ya que puede ser creado de manera independiente del lenguaje de programación y de características físicas del equipo que lo va a ejecutar. Un algoritmo debe ser, entre otras cosas: Preciso, definido, finito, correcto, debe de tener al menos una salida perceptible, debe ser sencillo, legible, eficaz y eficiente.

PROYECTO: FRANKY

PROBLEMA: Responder a preguntas predeterminadas

DATOS DE ENTRADA: Pregunta

DATOS DE SALIDA: Respuesta

ALGORITMO:

1. Correr el programa utilizando GCC.
2. El chatbot dará un mensaje de bienvenida.
3. El chatbot esperará la pregunta del cliente.
4. Usuario ingresará una pregunta. Si el robot conoce la respuesta ir al paso 5, si no, ir al paso 6.
5. El chatbot imprimirá la respuesta en pantalla y solicitará otra pregunta.
6. El chatbot imprimirá un mensaje de error en la pantalla y solicitará otra pregunta.
7. Si el usuario desea conocer otra respuesta regresar al paso 3, si no ir a paso 8
8. El chatbot imprimirá en la pantalla un mensaje de despedida
9. Salir-Fin

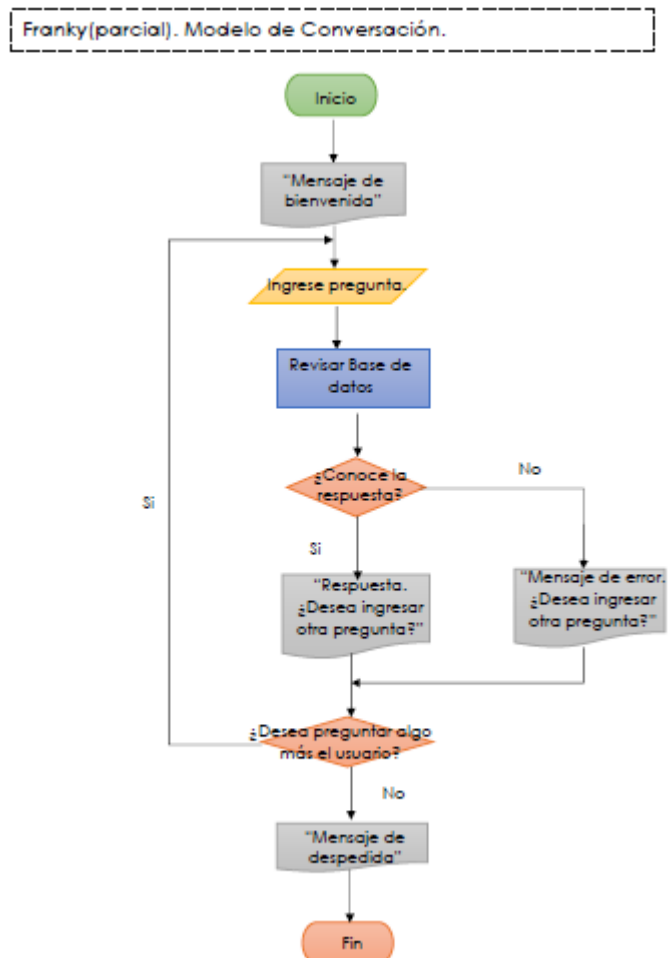
Durante el diseño se busca proponer una o varias alternativas viables para dar solución al problema y con base en esto tomar la mejor decisión para iniciar la construcción. Los algoritmos no solo son aplicables a los problemas relacionados con el software, muchas cosas que utilizamos al principio tenían algoritmos y los seguimos cotidianamente, pero no con este nombre, los conocemos como instrucciones o recetas, por ejemplo.

Un problema matemático es computable si éste puede ser resuelto por un dispositivo computacional. La teoría de la computabilidad es la parte de la computación que estudia los problemas de decisión que pueden ser resueltos con un algoritmo.

Aprendimos también que es un diagrama de flujo que es la representación gráfica de un Algoritmo, muestra gráficamente el flujo de acciones a seguir para cumplir con una tarea específica, a partir del diagrama de flujo es posible codificar un programa en algún lenguaje de programación.

Los diagramas de flujo poseen símbolos que permiten estructurar la solución de un problema de manera gráfica y todos ellos tienen características en común como por ejemplo: Deben tener un inicio y un fin, utilizan líneas para indicar la dirección del flujo del diagrama deben ser rectas verticales u horizontales exclusivamente y deben de estar conectadas a un símbolo, el diagrama debe ser construido de arriba hacia abajo (top-down) y de izquierda a derecha (left to right), su notación debe ser independiente del lenguaje de programación en el que se va a codificar la solución, se pueden poner comentarios que ayuden a su comprensión.

Sus estructuras de control de flujo permiten la ejecución condicional y la repetición de un conjunto de instrucciones. Existen 3 estructuras de control: secuencial (orden), condicional (si-sino, if-else) y repetitivas o iterativas (cíclicas, do-do while).



Una vez que hayamos el problema, este ha sido analizado, y se ha creado un algoritmo para su resolución, procedemos a la etapa de codificación del algoritmo, es básicamente escribir el algoritmo de tal forma que muestre en forma de texto los pasos a seguir para solucionar un problema. Es la representación escrita del Algoritmo.

Este es el código parcial del proyecto una que son muchas líneas de código, posteriormente se incluirá el. Puede continuar revisando en el siguiente apartado el código fuente.

1. Inicio
2. Declaración de variables a utilizar: opción, nombre, apellido
3. Imprimir " Hola ¡¡¡Bienvenido!!! Yo soy Franky un chatbot personalizado ¿En qué puedo ayudarte?"
4. Imprimir">>"
 - a. Si leer" hola"
 - i. Imprimir " Hola, ¿que tal? ¿Con quién tengo el gusto?"
 - ii. Leer "soy %s %s", nombre,apellido
 - iii. Imprimir"Mucho gusto %s, espero que nos llevemos bien",nombre
 - b. Si no, si leer "como estas?"
 - i. Imprimir "Muy bien,¿y tu? "
 - c. Si no, si leer "bien tambien"
 - i. Imprimir "¡¡¡ Super !!!"
 - d. Si no, si leer "¿como te llamas? "
 - i. Imprimir "Me llamo Franky, aunque tu puedes decirme como quieras "
 - e. Si no, si leer "¿que eres? "
 - i. Imprimir "Soy un chatbot personalizado y estoy tu servicio :3"
 - f. Si no, si leer "¿quien es tu creador? "
 - i. Imprimir "Fuí creado por Daniela Hernández V. Estudiante de la Facultad de Ingeniería de la UNAM"
 - g. Si no, si leer "¿que puedes hacer? "
 - i. Imprimir "pues...por ahora solo puedo hablar contigo. ¡Pero no te desanimes! posteriormente mi creadora me pondrá más funciones y podré servirte de una mejor manera"
- ...
- ...

Aprendí también lo básico del lenguaje C, e primer problema que tuve fue instalar GCC en mi PC y tuve que modificar el entorno, esto al principio para mi fue muy complicado, pero después de ello también aprendí las funciones básicas dentro de este lenguaje de programación los cuales los tuve que poner dentro de un editor



de texto, en este caso me acomodé mucho con Notepad++ y DevC++ ya que su interfaz gráfica es muy llamativa, además que en el caso de DevC++ el mismo editor nos permite compilar, ejecutar y depurar nuestro código fuente, mientras que Notepad++ no, este es mas llamativo visualmente pero el programa debe de compilar se y ejecutarse desde nuestra terminal.

Un lenguaje de programación nos permite expresar las instrucciones de un problema de tal forma que pueden ser realizadas por una computadora. El lenguaje C es un lenguaje de alto nivel, esto significa a que las instrucciones podrían ser entendidas fácilmente por un humano.

Mientras que un lenguaje de bajo nivel, son instrucciones que son mayor comprendidas por la maquina y de difícil entendimiento por una persona que no tenga conocimientos de la máquina en que operarán.

He aquí algunas de las cosas que aprendí, en forma de lista y algunas de las cuales se verán en las imágenes de mi proyecto más adelante.

- Comentarios
- Declaración de variables y tipos de datos
- Almacenar e imprimir variables
- Modificadores de alcance
- Expresiones lógicas
- Estructuras de selección (if, if-else, switch-case), repetición (while, do, do-while, for), (break, continue)
- Depuración de programas
- Arreglos unidimensionales y multidimensionales
- Apuntadores ~ en realidad no comprendí muy bien este tema~
- Funciones
- Lectura y escritura de datos ~ no me salieron las practicas ~

Ahora este es el código fuente de FRANKY, como podemos ver esta comentado y tiene tanto estructuras de repetición como de selección al igual que lectura y escritura de datos entre otros.

```
1  /*
2  *
3  * This program is free software: you can redistribute it and/or modify
4  * it under the terms of the GNU General Public License as published by
5  * the Free Software Foundation, either version 3 of the License, or
6  * (at your option) any later version.
7  *
8  * This program is distributed in the hope that it will be useful,
9  * but WITHOUT ANY WARRANTY; without even the implied warranty of
10 * MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
11 * GNU General Public License for more details.
12 *
13 * You should have received a copy of the GNU General Public License
14 * along with this program. If not, see <http://www.gnu.org/licenses/>.
15 *
16 * Author: Jorge A. Solano
17 *
18 */
19
20 //PROYECTO FINAL --FRANKY--
21 //Bibliotecas para entrada y salida de datos
22 #include <stdio.h>
23 #include <conio.h>
24 #include <string.h>
25 #include <process.h>
26 #include <stdlib.h>
27
28 int main() //Función
29 {
30     //Declaración variables a utilizar
31     int op;
32     char input[128], nombre[20], apellido[30]; //Podrá registrar cadenas de Caracteres
33     char sa=160, ae=130, si=161, ao=142, au=150, enic=164;
34     char sa=173, si=166, bc=126, Mq=62, mq=60, L=170, b=218, c=34, p8=40, p9=41;
35
36     //mensaje de bienvenida
37     printf("\n\t\t\tHola %c%c%cBienvenido!!!",sa,sa,sa);
38     printf("\n\t\t\tYo soy Franky un chatbot personalizado");
39     printf("\n\t\t\t¿En qué puedo ayudarte?\n",si,ae);
40
41     do //Implementación estructura do-while
42     {
43         printf(">>>");
44         gets (input); //Realizar una conversación implementando if-else
45
46         if (strcmp(input,"hola" )==0)
47         {
48             printf("\t\t\tHola,%cque tal?\n",si); //imprimir en pantalla con printf
49             printf("\t\t\t¿Con quién tengo el gusto?\n",si,ae);
50             printf("%c%c",Mq,mq);
51             scanf("soy %s %s", &nombre,&apellido); //Dar valor a una variable utilizando scanf
52             printf("\t\t\tMucho gusto %s, espero que nos llevemos bien\n",nombre);
53             continue;
54         }
55         else if (strcmp(input,"como estas?" )==0)
56         {
57             printf("\t\t\tMuy bien, ¿y tu?\n",si);
58         }
59         else if (strcmp(input,"bien tambien" )==0)
60         {
61             printf("\t\t\t%c%c%c Super !!! \n",sa,sa,sa);
62         }
63         else if (strcmp(input,"como te llamas?" )==0)
64         {
65             printf("\t\t\tMe llamo Franky, aunque tu puedes decirme como quieras %c%c%c%c%c\n",b,p8,L,L,p9,b);
66         }
67         else if (strcmp(input,"que eres?" )==0)
68         {
69             printf("\t\t\tSoy un chatbot personalizado y estoy tu servicio :3\n");
70         }
71     } while (1);
72 }
```

```

}
else if (strcmp(input,"quien es tu creador?")==0)
{
    printf("\t\t%c%c creado por Daniela Hernáandez V.\n\t\t%cEstudiante de la Facultad de Ingeniería de la UNAM\n",ai,aa,ai);
}
else if (strcmp(input,"que puedes hacer?")==0)
{
    printf("\t\t%cPues...por ahora solo puedo hablar contigo\n");
    printf("%cPero no te desanimes! posteriormente mi creadora me pondrá mis funciones\n\t\t%cPodrás servirte de una mejor manera\n",sa,aa,aa,ae);
}
else if (strcmp(input,"tienes sentimientos?" )==0)
{
    printf("\t\t%cNo, yo no puedo sentir nada :'( \n");
}
else if (strcmp(input,"hablame de ti" )->=0)
{
    printf("\t\t%c¿Ojalá deseas saber de mí?\n",si,ae);
    //Mostrar menú
    printf("\t\t%c\t\t 1)Edad\n\t\t\t 2)Genero\n\t\t\t 3)Cumpleaños\n\t\t\t 4)Hobby\n\t\t\t 5)Película favorita\n\t\t\t 6)Nada\n",en1e);
    do{
        //Solicitar la opción
        printf("\t\t%cElige la opción que desees realizar:",ao);
        scanf("%d",&op);

        switch (op){            //se ejecutará el caso de la opción seleccionada mediante switch -->case --> break
            case 1:
                printf("\n\t\t%cSolo tengo unos 3 meses de edad,Puedes crearlo?! ");
                printf("\n\t\t%cSoy todavía un bebé en su mas tierna infancia :3\n",ai,ae);
                break ;
            case 2:
                printf("\n\t\t%cPues en realidad yo no tengo género, de hecho ni siquiera tengo vida :D\n",ae);
                break ;

```

```

case 3:
    printf("\n\t\t\tYo cumplo años el 18 de septiembre, un mes muy colorido aquí en mi país\n\n",enie,ai);
    break ;
case 4:
    printf("\n\t\t\tNo tengo un hobby particular,\n\t\t\tpero disfruto recostarme por la tarde a dormir un buen rato");
    printf("\n\t\t\tPero shh... no le digas a mi creadora de ello o se enoja conmigo %c%c%c\n\n",aa,Mq,bc,mq);
    break ;
case 5:
    printf("\n\t\t\tMis películas favoritas son %cYo Robot%c y %cTerminator%c",C,C,C,C);
    printf("\n\t\t\tMe encantan las películas de ciencia ficción\n\n",ai,ao);
    break ;
case 6:
    printf("\n\t\t\tOk\n",aa);
    break ;
default:
    printf("\n\t\t\t%c%c%cOpción no válida!!!\n\n",sa,sa,sa,ao,aa);
    break ;
}
}while(op!=5);           //cuando op==6 se terminará la acción de switch y regresará a la estructura inicial de if-else
continue;
}
else if (strcmp(input,"adios")==0)
{
    printf("\n\t\t\tbye bye,%c%c%c¡es super conocerte!!! E) \n",sa,sa,sa);
    printf("\n\t\t\tVuelve pronto a hablar conmigo :) \n",sa,sa,sa);

    exit(0);           //sale del programa
}
else {
    printf("\n\t\t\tMis respuestas son limitadas\n\t\t\tdebes hacer las preguntas correctas\n",ao,aa); //imprimir este mensaje en caso de no
}
}
}while (strcmp(input,"adios")!=0);

getch ();
return 0;

```

```
C:\Users\HP\Desktop\LenguajeC\Franky>gcc Franky1.0.c -o Franky1.0.exe
```

```
C:\Users\HP\Desktop\LenguajeC\Franky>Franky1.0.exe
```

```

        Hola !!!Bienvenido!!!
        Yo soy Franky un chatbot personalizado
        ¿En qué puedo ayudarte?
>>hola
        Hola,¿que tal?
        ¿Con quién tengo el gusto?
>>soy Daniela Hernández
        Mucho gusto Daniela, espero que nos llevemos bien
>>
        Mis respuestas son limitadas
        debes hacer las preguntas correctas
>>como estas?
        Muy bien, ¿y tu?
>>bien tambien
        ¡¡¡ Super !!!
>>que eres?
        Soy un chatbot personalizado y estoy tu servicio :3
>>como te llamas?
        Me llamo Franky, aunque tu puedes decirme como quieras 「(-w-)」
>>quien es tu creador?
        Fui creado por Daniela Hernández V.
        Estudiante de la Facultad de Ingeniería de la UNAM
>>que puedes hacer
        Mis respuestas son limitadas
        debes hacer las preguntas correctas
>>tienes sentimientos?
        No, yo no puedo sentir nada :'(
>>
```

```
>>hablame de ti
        ¿Qué deseas saber de mi?
        1)Edad
        2)Genero
        3)Cumpleaños
        4)Hobby
        5)Película favorita
        6)Nada
        Elige la opción que desees realizar:1

        Solo tengo unos 3 meses de edad,Puedes creerlo?!
        Soy todavía un bebé en su mas tierna infancia :3

        Elige la opción que desees realizar:2

        Pues en realidad yo no tengo género, de hecho ni siquiera tengo vida :D

        Elige la opción que desees realizar:5

        Mis películas favoritas son "Yo Robot" y "Terminator"
        Me encantan las películas de ciencia ficción

        Elige la opción que desees realizar:3

        Yo cumpla años el 18 de septiembre, un mes muy colorido aquí en mi país

        Elige la opción que desees realizar:4

        No tengo un hobby particular,
        pero disfruto recostarme por la tardes a dormir un buen rato
        Pero shh... no le digas a mi creadora de ello o se enojará conmigo >~<

        Elige la opción que desees realizar:
```

```
No tengo un hobby particular,  
pero disfruto recostarme por la tardes a dormir un buen rato  
Pero shh... no le digas a mi creadora de ello o se enojaa conmigo >~<  
  
Elige la opción que desees realizar:6  
Ok  
  
>>  
Mis respuestas son limitadas  
debes hacer las preguntas correctas  
  
>>adios  
bye bye,!!!fue super conocerte!!! B)  
Vuelve pronto a hablar conmigo :)  
  
C:\Users\HP\Desktop\LenguajeC\Franky>
```

Al desarrollar un proyecto del gusto del alumno honestamente nos motiva y no obliga al mismo tiempo a tener una correcta planeación y estimación del proyecto, a analizar los requerimientos del sistema y software, diseñar su estructura de datos, la arquitectura del programa y el procedimiento algorítmico, así como su codificación y revisión.

Decidí que mi proyecto fuera un chatbot porque cada vez hay una mayor necesidad de mecanizarse y poder atender a tareas repetitivas de forma rápida y efectiva. Dentro del curso también aprendimos a darle un enfoque emprendedor a nuestro proyecto, las empresas tienen una constante necesidad de expandirse y de reinventarse, deben de ser capaces de dar una respuesta optima al aumento de demanda que les exigen sus consumidores.

Este proyecto se centró principalmente en la implementación de un chatbot del tipo predeterminado. Es decir que solo podrá responder a preguntas y estimulo predeterminados. Hoy en día este tipo de programas son ampliamente utilizados como una alternativa para poder dar un servicio automatizado al cliente y reducir la necesidad de la intervención humana dentro de una empresa. Es una herramienta fácil de utilizar y que no es demasiado compleja y que tiene muchas ventajas.

Hasta aquí me encantó como aprendí a tener un razonamiento lógico y ahora soy capaz de crear programas sencillos. A lo largo de todo mi semestre anterior en Fundamentos de Programación y de acuerdo a las practicas realizadas la verdad es que considero que aprendí mucho ya que hay que mencionar que yo no tenía ninguna formación en este ámbito en absoluto.

El proyecto de crear un chatbot (FRANKY) me hizo darme cuenta de algunas cosas, no es tan sencillo programar, me gustó su creación e implementación, pero el tiempo fue muy reducido, además de que solo utilicé un editor de texto y una terminal para programarlo. Podemos decir que estamos un paso más cerca de una era completamente digitalizada, y este es el objetivo que perseguimos desde hace mucho tiempo, por ahora son chatbots, robots capaces de simular una conversación humana, mañana que podría ser, es un tema muy interesante.

Referencias:

1. M.I. Marco Antonio Martínez Quintana, UNAM, Apuntes de la clase de Fundamentos de programación, marco.mtz.fi@gmail.com
2. Facultad de Ingeniería Área/Departamento: Laboratorio de computación salas A y B. (6 de abril de 2018). Manual de prácticas del laboratorio de Fundamentos de programación. México: UNAM.