

Indicaciones específicas:

- El desarrollo de esta práctica es estrictamente individual.
- Esta evaluación contiene 12 páginas (incluyendo esta página) con 4 preguntas. El total de puntos son 20.
- El tiempo límite para la evaluación es 110 minutos.
- Cada pregunta deberá ser respondida con el nombre de archivo solicitado.
 1. (a) `pregunta1_1.png`
(b) `pregunta1_2.png`
(c) `pregunta1_3.png`
(d) `pregunta1_4.png`
 2. `pregunta2.py`
 3. `pregunta3.md` o `pregunta3.txt`
 4. `pregunta4.sql` o `pregunta4.txt`
- Sólo se corregirá los archivos con dichos nombres. Otro archivo con otro nombre o extensión no será considerado para la calificación.

Calificación:

Tabla de puntos (sólo para uso del professor)

| Question | Points | Score |
|----------|--------|-------|
| 1 | 5 | |
| 2 | 5 | |
| 3 | 5 | |
| 4 | 5 | |
| Total: | 20 | |

1. (5 points) Compuertas Lógicas.

En el laboratorio 2 del curso, aprendimos que la computadora utiliza transistores para almacenar los datos sobre los cuales opera. Por esta razón, el procesador se ve en la necesidad del uso de los operadores lógicos AND, OR y NOT para poder realizar operaciones booleanas o aritméticas.

Un ejemplo de un operador matemático es el operador de suma, el cual recibe dos números a operar y devuelve un resultado.

El operador `medio sumador` realiza la operación de suma pero de sólo dos bits. Devolviendo un bit de acarreo y un bit de suma. Su tabla de verdad puede ser apreciada en la Figura 1:

| INPUTS | | OUTPUTS | |
|--------|---|---------|-----|
| A | B | CARRY | SUM |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Figure 1: Tabla de verdad del operador `medio sumador`

Donde SUM es la Suma de ambos bits, y CARRY es el acarreo de la suma.

Implemente el operador `medio sumador` utilizando la herramienta online usada en el laboratorio y únicamente los operadores AND, OR y NOT.

La operación `medio sumador` esta representada por las siguientes ecuaciones:

$$SUM = A\bar{B} + B\bar{A}$$

$$CARRY = A \cdot B$$

Ambas ecuaciones comparten las mismas entradas, por lo tanto, realizar ambas en el mismo circuito.

Tip: Empiece analizando la cantidad de entradas y salidas necesarias. Después de ello, empiece a agregar los operadores necesarios. Recuerde que solo puede utilizar los operadores AND, OR y NOT.

Suba una imagen del circuito para cada uno los casos mostrados en la tabla de verdad, en archivos llamados `pregunta1_1.png`, `pregunta1_2.png`, `pregunta1_3.png` y `pregunta1_4.png`.

La rúbrica para esta pregunta es:

| Criterio | Excelente | Adecuado | Mínimo | Insuficiente |
|---|--|--|---|---|
| Uso de compuertas lógicas (3pts) | Utiliza correctamente las compuertas lógicas para el circuito solicitado. (3pts) | Realiza correctamente el circuito solicitado para SUM, sin embargo, existe un error en las compuertas que calculan el CARRY. (2pts) | Realiza correctamente el circuito solicitado para el CARRY, sin embargo, existe un error en las compuertas que calculan el SUM. (1pts) | No calcula correctamente ni el SUM ni el CARRY. (0pts) |
| Identificación de entradas y salidas (2pts) | Identifica correctamente la cantidad de entradas y salidas en el circuito. (2pts) | Identifica correctamente la cantidad de entradas, pero no la cantidad de salidas. (1pts) | Identifica correctamente la cantidad de salidas, pero no la cantidad de entradas. (0.5pts) | No demuestra conocimiento de las entradas y salidas en el circuito. (0pts) |

2. (5 points) Operaciones con bits en Python

En el laboratorio 3, aprendimos algunos operadores a nivel de bits de python. Entre ellos, los operadores:

- `&`: realiza bit a bit la operación AND en los operandos.
- `|`: realiza bit a bit la operación OR en los operandos.
- `^`: realiza bit a bit la operación XOR en los operandos.

Se le presenta una función en python llamada `numero_unico`. Dicha función recibe 7 números, todos los números están repetidos una vez. **Sólo uno de ellos no está repetido.**

Se le pide modificar la función `numero_unico` de tal manera que guarde en la variable `respuesta` el número que no está repetido.

Restricción: Sólo puede utilizar los operadores mencionados. En otras palabras, solo puede utilizar `&`, `|`, `^`.

Tiene permitido utilizar el símbolo `=` para asignar el resultado a la variable `resultado`.

Listing 1: Funcion numero_unico

```
def numero_unico(num1,num2,num3,num4,num5,num6,num7):  
    # realice aqui sus operaciones  
    # respuesta = ...?  
  
    print(respuesta)
```

Ejemplo 1:

Invocar la función con los siguiente valores:

Listing 2: Ejemplo 1

```
numero_unico(4, 7, 15, 7, 4, 15, 9)
```

Retornará:

Listing 3: Output 1

```
9
```

Ejemplo 2:

Invocar la función con los siguiente valores:

Listing 4: Ejemplo 2

```
numero_unico(3, 1, 1, 4, 4, 3, 5)
```

Retornará:

Listing 5: Output 2

| |
|---|
| 5 |
|---|

Suba su respuesta en un archivo llamado `pregunta2.py`.

La rúbrica para esta pregunta es:

| Criterio | Excelente | Adecuado | Mínimo | Insuficiente |
|--|--|--|--|--|
| Desarrollo del código en Python (5pts) | El código funciona correctamente e imprime la respuesta con el comando indicado. (5pts) | El código funciona correctamente, sin embargo no imprime la respuesta con el comando indicado. (4pts) | El código funciona correctamente, utiliza adecuadamente las funciones binarias, sin embargo no devuelve la respuesta esperada. (0.5pts) | No realizó procedimiento alguno. (0pts) |

3. (5 points) Git

En el laboratorio 4 aprendimos a utilizar el Markdown de GitHub para crear nuestro archivo README en nuestra página de GitHub. Se le pide que replique el archivo ubicado en la siguiente página utilizando en Markdown de GitHub.

Puede obtener la imagen del siguiente link

Donde necesite especificar algún link, puede utilizar cualquier URL.

No utilice HTML. Sólo puede utilizar el Markdown de GitHub.

Suba su archivo como `pregunta3.md` o `pregunta3.txt`

My App

My App es una aplicación creada para ayudar a las personas a lograr cosas asombrosas!



Contribuir

Si quieres contribuir a nuestro proyecto, tienes que seguir los siguientes pasos

1. Ve a nuestro proyecto en GitHub y realiza un [fork!](#)
2. Realiza las modificaciones que consideres necesarias en tu copia del proyecto.
3. Envía un [PR!](#)
4. Revisaremos tu PR tan pronto como podamos. Agradecemos mucho tu ayuda!

Instalación

Necesitaras las siguientes herramientas para poder ejecutar el proyecto:

- Node.js 6.5+
- MySQL 8+
- npm 4+
- Angular cli
- Angular 8.x
- Ionic 2.x

Ejecutar el proyecto

Una vez instalados todos los programas necesarios, solo necesitas clonar nuestro repositorio:

```
gh repo clone <repo>
```

Despues de ello, entra a nuestro proyecto y ejecútalo:

```
npm start
```

Progreso

| Cargo | Encargado |
|----------|----------------|
| Frontend | Magnus Carlsen |
| Backend | MVL |
| Database | Andrew Tang |

La rúbrica para esta pregunta es:

| Criterio | Excelente | Adecuado | Mínimo | Insuficiente |
|---|--|---|---|---|
| Uso de títulos y subtítulos (1pts) | Identifica y utiliza adecuadamente títulos y subtítulos donde corresponde. (1pts) | Usa adecuadamente títulos, sin embargo no identifica subtítulos. (0.75pts) | Usa adecuadamente títulos y subtítulos, sin embargo, faltó incluir un título/subtítulo de los solicitados. (0.25pts) | No demuestra conocimiento del uso de títulos o subtítulos. (0pts) |
| Creación de tablas (1pts) | Realiza correctamente la creación de la tabla solicitada. (1pts) | Realiza la creación de la tabla sin definir correctamente la cabecera de la tabla. (0.75pts) | Realiza la creación de la tabla, sin embargo no incluye todas las filas o columnas solicitadas. (0.25pts) | No demuestra conocimiento del uso adecuado de tablas. (0pts) |
| Creación de listas enumeradas y sin enumerar (1pts) | Realiza correctamente la creación de listas enumeradas y sin enumerar. (1pts) | Realizar correctamente la creación de listas enumeradas y sin enumerar, pero no toda la información solicitada. (0.5pts) | Realiza la creación de listas enumeradas pero no las sin enumerar, o viceversa. (0.25pts) | No demuestra conocimiento de listas enumeradas y sin enumerar. (0pts) |
| Creación de links, bloques de código (1pto) | Realiza correctamente la creación de links y bloques de código. (1pts) | Realizar correctamente la creación de links y bloques de código, pero no toda la información solicitada. (0.5pts) | Realiza la creación de links pero no los bloques de código, o viceversa. (0.25pts) | No demuestra conocimiento de links y bloques de código. (0pts) |
| Inclusión de imágenes y texto (1pts) | Realiza correctamente la creación imágenes y el texto solicitado (1pts) | Realizar correctamente la creación de la imagen, sin embargo el archivo no tiene todo el texto requerido (0.5pts) | No realiza correctamente la creación de la imagen, pero el archivo tiene el resto del texto (0.25pts) | No demuestra conocimiento de inclusión de imágenes ni de texto plano. (0pts) |

4. (5 points) SQL

En el laboratorio 6 aprendimos a utilizar SQL para la creación de tablas en una Base de Datos online.

El Gobierno del Perú desea realizar la creación de una base de datos para almacenar la información de las personas que se han vacunado, y los detalles de las dosis que recibieron.

Se le pide para ello, con los siguientes detalles:

1. Crear una tabla llamada **persona_vacunada**. Debe almacenar la siguiente información

- Nombre completo (Campo obligatorio)
- Género (Campo opcional)
- Email (Campo opcional)
- Nombre de la vacuna (Campo obligatorio)
- Cantidad de dosis (Campo obligatorio)

Usa una columna **id** como llave primaria, esta debe incrementarse con cada inserción.

2. Crear las sentencias **INSERT** para ingresar los siguientes datos a la base de datos:

- La persona con nombre Francisco Vilchez de género Masculino ha recibido 2 dosis de la vacuna Pfizer. No ha especificado email.
- La persona con nombre Mauree Turner recibió 2 dosis de la vacuna Sinopharm. No especificó género, pero indicó que su correo es mauree@turner.com
- La persona con nombre Rameshbabu Pragganandhaa recibió 2 dosis de la vacuna Covaxin. No especificó su genero ni su correo.

Suba sus comandos en el orden en que deben ser ejecutados en el archivo **pregunta4.sql** o **pregunta4.txt** en su defecto.

La rúbrica para esta pregunta es:

| Criterio | Excelente | Adecuado | Mínimo | Insuficiente |
|---------------------------|--|--|---|---|
| Creación de tablas (2pts) | Realiza la creación de la tabla utilizando el comando adecuado, con los atributos necesarios para guardar la información solicitada y respectivos tipos de datos adecuados. (2 pts) | Realiza la creación de la tabla utilizando el comando adecuado con los tipos de dato adecuados, sin embargo, los atributos utilizados no permiten almacenar la información en su totalidad. (1 pts) | Realiza la creación de la tabla utilizando el comando adecuado, sin embargo los atributos utilizados y los tipos de datos usados son incorrectos ya que no permiten almacenar la información solicitada. (0.5 pts) | No demuestra conocimiento de la sintaxis del comando para la creación de tablas. (0 pts) |
| Inserción de datos (3pts) | Realiza la correcta inserción de cada uno de los datos solicitados. (3 pts) | Realiza correctamente inserción de alguno de los datos solicitados. (2 pts) | El comando de inserción de datos tiene la sintaxis correcta, sin embargo los tipos de datos utilizados o el orden de ellos no va conforme con la estructura de la tabla. (1 pts) | No demuestra conocimiento de la sintaxis del comando para la inserción de datos. (0 pts) |