



El futuro digital  
es de todos

MinTIC

# «Misión TIC 2022»

## RETO DE LA SEMANA 5

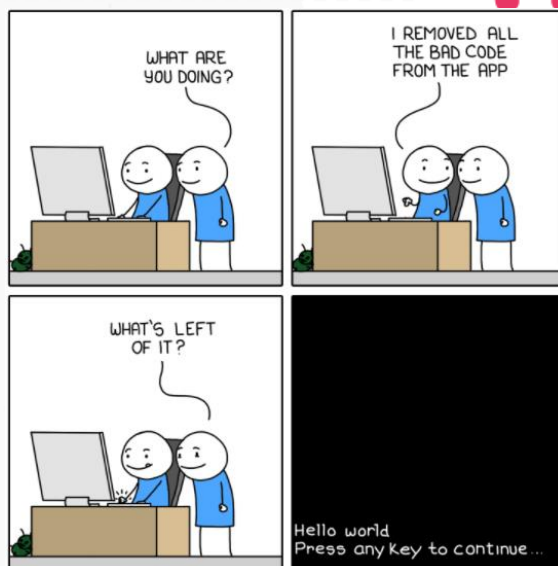
Los primeros “trabajadores” del conocimiento son los desarrolladores de software.



Universidad de Caldas



## Reto: Semana 5



## PARQUEADERO AUTOMATICO

Temas que abarca este reto:

- IDEAL
- Expresiones lógicas
- Ciclos
- Listas
- Definición de Funciones
- Parámetros y Argumentos
- Módulos de usuario
- Comentarios
- Invocación/llamado de funciones (propias y de terceros)
- Cadenas de caracteres
- Composición de funciones
- Entrada y salida por consola
- Validación de la entrada de datos



## RETO 5 – SEMANA 5

### PARQUEADERO AUTOMATICO

Como ya sabes manejar las listas, tuplas, pilas y colas; Don Antonio, ha comprado un sistema de parqueadero automático de carros, el cual se necesita programar para que reciba los carros, los organice y cuando sean solicitados por su dueño, el sistema lo busque, lo retire y se lo entregue a su dueño. Para ello, el sistema debe cumplir una serie de condiciones y restricciones. Empecemos:

El parqueadero consta de **LADO A** y **LADO B**, en total el parqueadero automático recibe 40 carros, es decir 20 carros por cada lado.

El **trabajo mínimo** que pide Don Antonio para que comience a funcionar el parqueadero, es que le puedas entregar el programa funcionando del lado A, claro está que el estaría muy contento si le entregas funcionando los 2 lados. Usted decide...

Entonces para poner a funcionar el parqueadero automático el programa debe hacer lo siguiente:

1. El sistema debe revisar el número de placa del carro y determinar la ubicación de acuerdo con:
  - a. Si entra un carro con placas terminadas en **PAR** en su penúltimo dígito, el carro se ubica en el **LADO A** del parqueadero.
  - b. Si el carro en su placa tiene como último dígito **IMPAR** se debe ubicar en el bloque **B1-IP**, pero solo puede recibir máximo 5 carros, si llenado el bloque anterior el sistema debe avisar que está lleno el bloque y debe ubicarlo en el siguiente bloque de **impares**, **B3-IC**, recuerda que este también máximo solo recibe hasta 5 carros.
  - c. Si los dos bloques de carros **impares** se llenan, debe salir un mensaje que el cupo está lleno e informar al cliente que no se puede recibir su vehículo. (Aquí es uno de los puntos mínimos para entregar a Don Antonio).
  - d. Si la placa tiene como último dígito **PAR**, se debe ubicar el vehículo en el bloque **B2-PC**, pero recuerde que solo puede recibir máximo 5 carros. Si llenado el bloque anterior, el sistema debe ubicar el carro en el siguiente bloque de **pares** **B4-PP**, ten en cuenta también aquí que máximo solo se reciben hasta 5 carros.

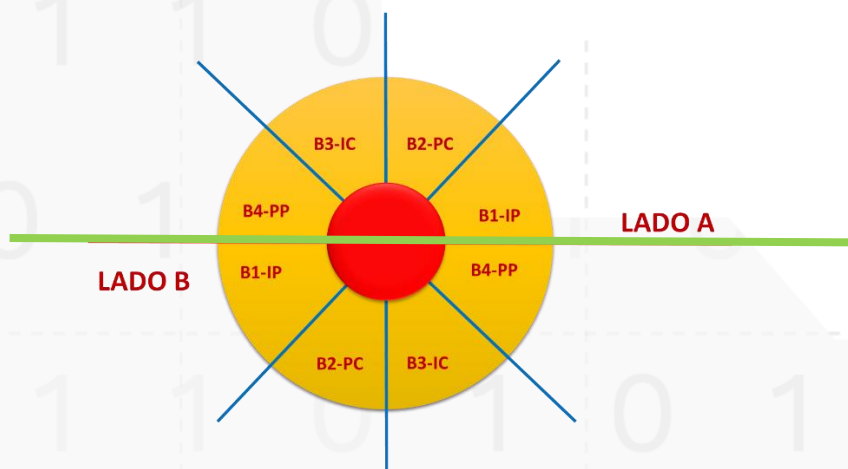


- e. Si los dos bloques de carros **pares** se llenan, debe salir un mensaje que el cupo está lleno. Aquí es lo mínimo.
- f. Ahora tanto para los carros pares e impares, si desean complacer a Don Antonio **(Esto es opcional)**, entregándole un programa completo, cuando esté lleno el **LADO A**, has que el programa confirme primero que hay espacio en el LADO B y ofrécele al cliente, ubicar su carro en este lado, sino acepta el cliente simplemente se va y termina el sistema.
- g. El LADO B, cumple los mismos parámetros del LADO A.
- h. Para saber cómo el sistema debe manejar el ingreso y retiro de vehículos, revisa la **tabla de códigos de bloque y condiciones**.
- i. Por último, Don Antonio tiene un cliente que le ha entregado 6 vehículos de exposición que se encuentran en la parte superior del parqueadero automático, **3 en lado A y 3 en el Lado B**. Él requiere que cuando el cliente venga a retirar estos vehículos, se le solicite un código de acceso para retirarlos y que consta de la multiplicación de sus 3 números que conforman la placa, solo al digitarlos, se podrá retirar esos vehículos que están con una seguridad tipo encriptación de octavo nivel llamado **TUPLA**.





Mirada del parqueadero desde arriba:



En la siguiente tabla veras la distribución y las condiciones para agregar o retirar los carros del sistema de parqueadero automático de Don Antonio. Ten en cuenta que el parqueadero solo tiene 5 niveles.

**TABLA DE CÓDIGOS DE BLOQUE Y CONDICIONES**

UBICACIÓN	CONDICION DE ENTRADA	PLACA ULTIMO DIGITO	PLACA- PENULTIMO DIGITO	LADO
B1-IP	CARRO ENTRA EN CONDICION DE PILA	IMPAR	PAR	LADO A
B2-PC	CARRO ENTRA EN CONDICION DE COLA	PAR	PAR	LADO A
B3-IC	CARRO ENTRA EN CONDICION DE COLA	IMPAR	PAR	LADO A
B4-PP	CARRO ENTRA EN CONDICION DE PILA	PAR	PAR	LADO A
B1-IP	CARRO ENTRA EN CONDICION DE PILA	IMPAR	IMPAR	LADO B
B2-PC	CARRO ENTRA EN CONDICION DE COLA	PAR	IMPAR	LADO B
B3-IC	CARRO ENTRA EN CONDICION DE COLA	IMPAR	IMPAR	LADO B
B4-PP	CARRO ENTRA EN CONDICION DE PILA	PAR	IMPAR	LADO B



Para los que deseen complacer a Don Antonio y completar la segunda parte del parqueadero, aquí está el esquema:



- Datos del cliente:** El sistema debe registrar en una agenda (diccionario) los datos de los clientes inscritos para el servicio, si el cliente no existe cuando se consulte, este se debe crear antes de dar acceso al servicio de parqueadero, la agenda debe estar ordenada alfabéticamente cuando se consulte.

**NOTA:** Tenga en cuenta que cliente que no esté registrado en la agenda no puede entrar su carro al parqueadero automático.

Los datos del cliente que la agenda debe registrar son:

Nombre del Cliente:

ID Cliente:

Marca del carro:

Modelo del carro:

Placa del Carro:



Don Antonio quiere que el sistema tenga todos los menús y submenús necesarios para poder trabajar fácilmente en el sistema y además que cuando Don Antonio desee consultar donde está ubicado el carro del cliente, **él pueda consultar la agenda y con base al Id del cliente pueda saber dónde está el carro ubicado en el parqueadero automático.**

**NOTAS FINALES:** Recuerde hacer las funciones respectivas en un archivo llamado `parqueadero.py` y en el MAIN se debe realizar todos llamados a las funciones por medio de los menús.

## A solucionar el problema

---

Plantea una solución a cada función del reto aplicando las 4 primeras actividades del método ideal, utiliza lo que necesites: dibujos, investiga fórmulas, Google, busca opciones de solución, plantea estrategias, describe algoritmos y especifica requisitos!

Te sugiero que más que seguir el método a ciegas, sácale provecho a lo que te aporta cada etapa. Si te es más fácil hacerlo con papel y lápiz o en un tablero en tu casa (o pared, vidrio, etc.) muchísimo mejor; luego le tomas fotos a la solución y las pegas a un documento en Word.

## A programar en Python

---

Recuerda que este reto debe hacerse en Repl.it, con tu cuenta de Gmail. Tu profesor formador te dará un lugar en donde podrás escribir tu código y lo más importante. ¡Probar si quedo bien!, (no desde el punto de vista sintáctico, esos errores te los informara Repl.it y podrás solucionarlos o pedir ayuda). Probaremos que los resultados obtenidos sean los esperados.

## Que entregar

---

1. Un documento con el resultado de aplicar IDEA: este debes subirlo al enlace que se te habilitará en el aula virtual de Moodle.
2. Un programa en Python (L), que solucione el reto acorde con lo entregado en el punto 1 y 2.