



Universidad del Valle

Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación
Fundamentos de programación imperativa

NORMAS PARA LA ENTREGA DE LOS LABORATORIOS

- Coloque el nombre de los integrantes del grupo (máximo tres estudiantes con sus códigos estudiantiles) presentado en el encabezado de todos sus programas. En caso de no completar la información solicitada se aplicará una penalización de acuerdo a las rúbricas de evaluación.
- Se solicita que el informe tenga los nombres y códigos de los integrantes del grupo.
- Así mismo se solicita que su entrega sólo sea un archivo comprimido que contenga el archivo PDF del informe y el código fuente de su taller en archivos .py
- Entregue un sólo archivo por cada uno de los puntos del taller y use un nombre que permita su fácil identificación tal como: punto1.py, punto2.py, etc
- Debe entregar el laboratorio por el campus virtual, una entrega por grupo. Así mismo se permiten entregas atrasadas, sin embargo, estas tendrán una sanción de 0.3 por hora o fracción. Es decir, si entrega el 21 de marzo a las 00:00:01 tendrá 0.3 de sanción, si entrega a las 01:00:01 tendrá 0.6 y así sucesivamente.
- Guarde una copia del código fuente de todos sus programas y la evidencia de la ejecución. Esto se puede realizar copiando y pegando las ventanas con la entrada(s) y salida(s) solicitadas. Inclúyalas en el informe correspondiente.
- **Lugar y Medio de Entrega:** Todos los archivos que se soliciten en el informe (.py y .pdf) se deben subir al campus virtual, en el respectivo link del laboratorio.
- **Plazo:** Los estudiantes deben subir los archivos antes del día y la hora establecida por el profesor en el campus, el vínculo para esta actividad se deshabilitará automáticamente una vez se cumpla el plazo.
- Durante el curso no se recibirán informes de laboratorio enviados por correo electrónico y tampoco se podrá entregar informes al monitor.

Laboratorio No. 1 Tema: Algoritmos básicos de entrada y salida

Fecha Asignación: 11 de marzo de 2022

Fecha máxima de entrega: 20 de marzo de 2022 23:59:59.

Objetivos:

- Identificar la estructura básica de un algoritmo

- Familiarizar al estudiante con las fases del desarrollo de un algoritmo: análisis, pseudocódigo, prueba de escritorio e implementación
- Familiarizar al estudiante con la identificación de la sintaxis de un programa en Python
- Reconocer la importancia de los tipos de datos en un lenguaje de programación
- Utilizar las instrucciones aritméticas para realizar cálculos básicos en Python
- Comprender el uso y la sintaxis de las operaciones de lectura y escritura mediante la función ***input()*** y la función ***print()***

Logros evaluados

Resultado de aprendizaje 2

1. Implementa en un lenguaje de programación un algoritmo que da solución a un problema que incluya entrada y salida, estructuras de decisión, ciclos, arreglos y/o funciones
2. Rastrea la ejecución de segmentos de código y describe su cómputo
3. Aplica una metodología de desarrollo para un problema específico y entrega los elementos de cada etapa

Metodología:

El laboratorio debe ser desarrollado en grupos de tres personas.

Siga **paso a paso** la metodología planteada en el curso y resuelva los siguientes ejercicios.

Para cada ejercicio los entregables son:

1. Análisis del problema
2. Algoritmo en pseudocódigo
3. Prueba de escritorio.
4. Algoritmo en Python
5. Evidencias de ejecución del programa, mediante capturas de despliegue de pantallas

Se debe entregar un informe PDF con los puntos 1, 2, 3, y 5 de la metodología. El punto 4 corresponde a los archivos **.py**

1) Suramericano sub 20

La FIFA de Colombia requiere de un programa que permita calcular la tabla de resultados de los equipos que están participando en el Suramericano sub 20. Los resultados por equipo se expresan en función de la información que se presenta en la Tabla 1 (Ejemplo puntuación de Colombia).

Usted debe desarrollar un algoritmo que calcule los puntos obtenidos por el equipo y la diferencia de goles. Tenga en cuenta que por cada partido ganado obtendrá 3 puntos, por cada partido empatado 1 punto y por cada partido perdido 0 puntos.

Tabla 1. Información por Equipo de Futbol

Equipo	Pts.	PJ	G	E	P	GF	GC	Dif.
Colombia	5	3	1	2	0	4	2	2

Glosario - **Pts.**: Puntos, **PJ**: Partidos, **G**: Ganados, **E**: Empates, **P**: Derrotas, **GF**: Goles a Favor, **GC**: Goles en Contra, **Dif.**: Diferencia de Goles

Recuerde que debe solicitar los siguientes datos:

- Nombre del equipo
- Cantidad de partidos ganados
- Cantidad de partidos empatados
- Cantidad de partidos perdidos
- Número de goles a favor

- Número de goles en contra.

Además, debe presentar resultados del equipo con los valores solicitados y los puntajes obtenidos(Como se indica en la tabla de ejemplo).

Debe imprimir la siguiente información de acuerdo al ejemplo de la tabla 1

Nombre: Colombia: Pts: 5, PJ: 3, G:1, E:2, P:0, GF: 4, GC: 2, Dif: 2

La forma de presentar la información es libre, pero debe mostrar todos los datos.

2. Cálculo de número de CDs

Los profesores de la Universidad del Valle desean conocer la cantidad de CDs que necesitan comprar para hacer una copia de seguridad del disco duro de sus computadores.

Lo han contratado a usted para que realice un algoritmo que calcule la cantidad de los CDs requeridos y el precio más IVA de estos. Para ello debe solicitar el nombre del profesor, la capacidad de almacenamiento del disco duro en GB y el valor unitario del CD. Una vez se han calculado los valores se debe visualizar los resultados obtenidos.

Tenga en cuenta que el disco duro está lleno y que su capacidad está dada en GigaBytes (GB). Además, un CD en blanco tiene una capacidad de 700 MegaBytes (MB) y que un GB equivale a 1,024 MB.

Ejemplo:

Tamaño del disco: 40GB

Valor del CD: 5000

Proceso: Total a almacenar: 40960, Número de CD: 58,51 CD, aplicando techo son 59.

La salida debería ser así:

“Para almacenar 40GB se requieren 59 Cds que cuesta 295000”

Para aplicar techo a una división:

```
import math
```

math.ceil()

Revisar: https://www.w3schools.com/python/ref_math_ceil.asp

3. Condominio Campestre

El condominio campestre “Nuevo Amanecer” tiene una serie de lotes para la venta y desea implementar un programa que calcule el número de hectáreas de los lotes y el valor de venta correspondiente. Los lotes son rectangulares y sus dimensiones están dadas en metros (ver Fig. 1).

Para el desarrollo del algoritmo debe solicitar las dimensiones del lote (su base y su altura) y el precio por metro cuadrado del lote. Y debe mostrar el número de hectáreas, el valor del lote y el valor del impuesto predial si se conoce que el porcentaje de este impuesto es del 1.3% sobre el valor del lote.

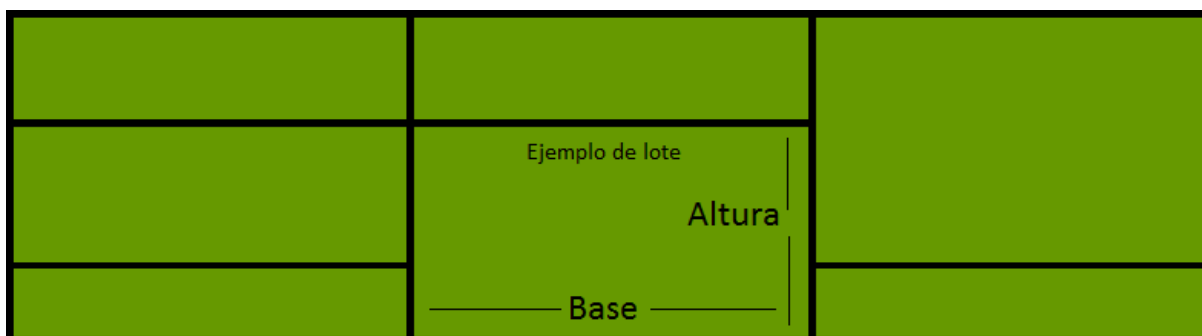


Figura 1. Distribución de lotes

Rubricas de evaluación

Cumplimiento de las reglas del taller

Concepto	Nivel 0 (0 puntos)	Nivel 1 (5 puntos)	Nivel 2 (10 puntos)
Organización de la entrega	No realiza la entrega	Entrega más de un archivo .py por cada punto del taller o sus nombres no son claros con respecto al punto que resuelven.	Entrega exactamente un archivo .py para cada punto del taller y sus nombres son claros, por ejemplo: punto1.py, punto2.py, etc.
Formato del informe	No entrega el informe	Entrega el informe en un formato diferente al formato PDF	Entrega el informe en formato PDF
Archivo de entrega	No realiza la entrega	Entrega un archivo comprimido que	Entrega un archivo comprimido que

		contiene archivos comprimidos dentro	únicamente contiene el informe en formato PDF y los archivos .py de los puntos
Nombres de los estudiantes dentro del código	No incluye dentro de las primeras líneas de los archivos un comentario que tenga los nombres y códigos de los estudiantes del grupo de trabajo	Incluye dentro de las primeras líneas de los archivos un comentario que tenga los nombres y códigos de los estudiantes del grupo de trabajo, pero hace falta en uno o más de los archivos .py o bien a alguno de los estudiantes le hace falta su código estudiantil	Incluye dentro de las primeras líneas de los archivos un comentario que tenga los nombres y códigos de los estudiantes del grupo de trabajo en todos los archivos .py

Estas rubricas corresponden a 40 puntos del taller.

Cada uno de los puntos

Estas rubricas corresponden a la evaluación de cada uno de los puntos de taller de forma independiente de acuerdo a los logros del curso.

Logro	Nivel 0 0 puntos	Nivel 1 10 puntos	Nivel 2 20 puntos
Implementa en un lenguaje de programación un algoritmo que da solución a un problema que incluya entrada y salida, estructuras de decisión, ciclos, arreglos y/o funciones	No implementa el código en Python	Implementa el código en Python, pero no incluye input para solicitar la información o el código tiene algún error con respecto a la especificación o no muestra la salida con un print.	Implementa el código en Python, incluye input para solicitar la información al usuario y el código no tiene ningún error con respecto a la especificación y muestra la salida con un print.
Rastrea la ejecución de segmentos de código y describe su cómputo	No realiza la prueba de escritorio o no adjunta capturas sobre evidencias de ejecución	Realiza la prueba de escritorio, pero esta presenta fallas con respecto al pseudocódigo propuesto o bien no realiza al menos 3 ejemplos conceptuales de su algoritmo o adjunta capturas de menos de 3 ejemplos de ejecución en el código dentro del informe.	Realiza la prueba de escritorio correctamente con respecto al pseudocódigo propuesto y realiza al menos 3 ejemplos conceptuales de su algoritmo y adjunta capturas de al menos 3 ejemplos de ejecución en el código dentro del informe.

Aplica una metodología de desarrollo para un problema específico y entrega los elementos de cada etapa	No aplica la metodología vista en clase para la solución de problemas en programación o no realiza dos o más pasos de ella o no la evidencia dentro del informe en formato PDF	Aplica la metodología vista en clase, pero aplica incorrectamente uno de sus pasos o no lo realiza. Evidencia la metodología en el informe en formato PDF	Aplica la metodología vista en clase y aplica correctamente sus pasos. Evidencia la metodología en el informe en formato PDF
--	--	---	--

Cada punto tiene 60 puntos, por lo que en total tenemos 180 puntos

Cálculo de la nota del laboratorio

El total del laboratorio **n** es de 220 puntos, por lo que la nota se calculará así tomando en cuenta **p** como los puntos obtenidos en todo el laboratorio.

$$n=5,0 \frac{p}{220}$$