

# [Trabajo Práctico Integrador]

- Año 2017-

Algoritmos y Estructura de Datos  
Matemática Discreta

# **Generalidades**

## **Marco contextual del TP**

Este año proponemos que enfoquen la resolución de cada Ejercicio del TP a una situación de actualidad, particularmente vamos a centrarnos en las elecciones legislativas P.A.S.O. Las elecciones P.A.S.O. tienen lugar el segundo domingo del mes de agosto de los años impares, y es un método de selección de candidatos para cargos públicos electivos, los ganadores serán quienes se presentarán como candidatos en las elecciones definitivas.

## **Consultas sobre el trabajo práctico**

Para consultas puntuales, puede utilizarse el foro disponible en el campus virtual. Por favor, no revelar resoluciones para otros grupos, intentar que las consultas sean de concepto.

## **Publicación de notas**

Mediante el código de grupo que reciban, se podrá chequear la nota obtenida en una planilla que se publicará en el campus virtual apenas se haya realizado la corrección.

# Instructivo para Algoritmos y Estructura de Datos

Este trabajo práctico está pensado para los cursos **ANUALES**.

## Organización de los grupos

Cada grupo deberá auto-gestionarse, determinando un representante que se ocupará de las tareas administrativas del grupo. Una vez que se inscriban recibirán un código de grupo, con el cual deben identificarse para realizar las entregas, realizar preguntas, etc. Para inscribir el grupo deberán enviar el archivo “Planilla para la Inscripción de Grupos” que estará presente el campus virtual al mail [tp.aed.utnba@gmail.com](mailto:tp.aed.utnba@gmail.com). Este mail sólo lo usaremos para realizar la inscripción de grupos, el resto de las cuestiones que surjan se canalizarán a través del campus virtual.

Los grupos estarán conformados entre 7 y 10 integrantes, que pueden ser intercurso. Estos grupos permanecerán hasta fin de año, sin posibilidad de reagruparse a lo largo del año.

## Condición de aprobación

Para aprobar el trabajo práctico se deben realizar correctamente **TODOS** los ejercicios propuestos utilizando el lenguaje C++ utilizando el Paradigma Procedural. Si bien, el lenguaje permite el uso del Paradigma de Objetos no se puede utilizar para el presente trabajo práctico. Se deben cumplir las fechas de entrega, no cumplirlas significa quedar fuera de la carrera por la Aprobación Directa. Una vez realizada la entrega (completa) si se debe ajustar algo, habrá una instancia de recuperación.

Una vez aprobado el trabajo práctico, se citará a aquellos estudiantes que estén en condiciones de promocionar la Asignatura para hacer un coloquio individual sobre el trabajo práctico. El coloquio debe aprobarse.

Si se detectase una situación de copia entre grupos, todos los integrantes quedarán automáticamente fuera de la Aprobación Directa.

Se recomienda utilizar notación Camel Case para los identificadores, tanto de variables como de funciones. Asimismo será obligatorio que los nombres de dichos identificadores se consignen en español; y que las estructuras de datos a utilizar sean las vistas en clase; se rechazarán todas las soluciones bajadas directamente de internet.

Una vez que se haya aprobado **TODO** el trabajo práctico, se debe entregar una carpeta al docente del curso (1 por equipo) que cuente con:

- Una carátula con los integrantes ACTUALES del equipo y el código de grupo asignado.
- Los instructivos de cada ejercicio impresos.
- Un CD con las entregas realizadas.

***No se va a considerar como válido la Aprobación del Trabajo Práctico del año anterior.***

## Entrega del trabajo práctico

Una vez que llegue la fecha de entrega de cada una de las partes del trabajo práctico, el líder del grupo debe subir al campus virtual un .rar que debe llevar el siguiente nombre: AED-XXXX-PARTE1 o AED-XXXX-PARTE2 según corresponda. Donde XXXX será el código de grupo.

Se debe respetar la siguiente estructura de archivos:

AED-XXXX-PARTEX (.rar)

- Carátula indicando explícitamente los integrantes ACTUALES del grupo.
- Ejercicio X (carpeta)
  - Ejercicio X - Instructivo (.pdf)
  - Ejercicio X - Código (.cpp)
  - Ejercicio X - Nombre de Archivo (.txt) (Si los hubiese)

**La estructura de los ejercicios debe repetirse por cada uno. El instructivo comprende:**

- Un instructivo de uso para el programa junto con capturas del programa funcionando.
- La justificación de cualquier decisión tomada a la hora de resolver el problema.
- Si el ejercicio tuviese archivos asociados se debe replicar en el documento la estructura de los archivos con los datos que contienen.

<u>Primera Entrega:</u>	Conformación de grupos al mail <a href="mailto:tp.aed.utnba@gmail.com">tp.aed.utnba@gmail.com</a> .	Fecha límite 31 de Mayo de 2017 a las 23:59 horas.
<u>Segunda Entrega:</u>	Comprende la primera parte del trabajo práctico.	La entrega estará habilitada desde el 3 de julio de 2017 a las 00:00 horas y hasta el 31 de julio de 2017 a las 23:59 horas.
<u>Tercera Entrega:</u>	Comprende la segunda parte del trabajo práctico.	La entrega estará habilitada desde el 2 de octubre de 2017 a las 00:00 horas y hasta el 30 de octubre de 2017 a las 23:59 horas.

Si la entrega no respeta lo expuesto arriba, no será considerada y se perderá la posibilidad de la Aprobación Directa.

# Parte 1:

## Presentación

El objetivo de esta primera parte es poner en práctica el concepto de Estructuras de control de flujo de un programa (secuencia, análisis de caso y repetición), como asimismo trabajar con tipos de datos simples y estructurados (arrays), con expresiones lógicas, asignación interna y externa. También usaremos flujos de texto conocidos como “Archivos” y patrones de ordenamiento.

## Ejercicio 1

En este punto se deberá escribir un programa que interactúa con el usuario para asignar el valor de verdad a un conjunto de proposiciones lógicas, luego el sistema con la información que ingresa el usuario, debe resolver el valor de verdad de la implicación lógica e informar por pantalla el resultado de la misma. A continuación se detallan las dos implicaciones lógicas que deberá resolver el sistema:

<u>Implicación lógica 1:</u>	<b><i>Si una persona pertenece al partido político y cuenta con el aval del 2% de los afiliados a dicho partido, entonces la persona puede ser precandidato.</i></b>	$p \wedge q \Rightarrow r$
<u>Implicación Lógica 2:</u>	<b><i>Si el voto es emitido mediante boleta no oficializada, o con papel de cualquier color con inscripciones o imágenes de cualquier naturaleza; o mediante dos o más boletas de la misma categoría de candidatos y diferente agrupación política; entonces el voto se considerará nulo.</i></b>	$p \vee q \vee r \Rightarrow s$

El sistema le deberá solicitar al usuario los valores de verdad de las proposiciones que se encuentran en el antecedente de la implicación e informará por pantalla el valor de verdad del consecuente

## Ejercicio 2

Generar archivos con las actas que se confeccionan luego de realizar el escrutinio de las mesas para enviarlos al centro de cómputos. La escuela (lugar de votación) que nos asignaron posee 5 mesas y los archivos deben nombrarse de la siguiente manera:

Mesa\_4568.txt  
Mesa\_4569.txt  
Mesa\_4579.txt

Mesa\_4581.txt

Acta de la Mesa **Número de Mesa**

Agrupación Política - Número de Lista - Nombre del Candidato - Cantidad de Votos

• • • •

Votos impugnados	Cantidad de Votos
------------------	-------------------

-----

Agrupación Política - Número de Lista -Nombre del Candidato - Cantidad de Votos

• • • •

Votos Impugnados	Cantidad de Votos
------------------	-------------------

### Ejercicio 3

- Nombre Agrupación
- Número de Lista
- Nombre Candidato
- Cantidad de Votos

1. Cantidad de Votos.
2. Nombre de la Agrupación Política.
3. Número de Lista
4. Nombre del Candidato.
5. Nombre del la Agrupación Política y Cantidad de Votos.
6. Nombre del Candidato y Cantidad de Votos.
7. Nombre de la Agrupación Política, Nombre del Candidato y Cantidad de Votos.

## Ejercicio 4

Sobre el Array cargado en el punto anterior se pide calcular:

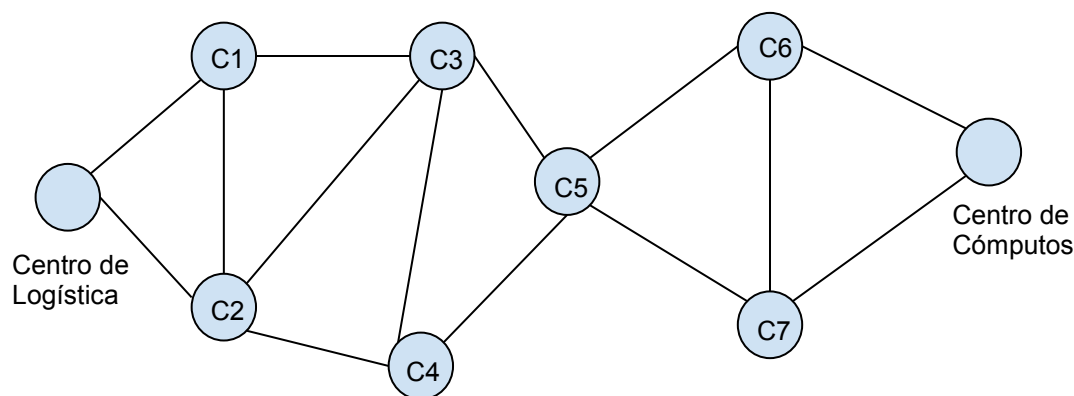
1. Los Candidatos que en esa escuela hicieron una buena elección. Una buena elección se considera cuando un Candidato saca entre un 17% y 30% de los votos totales (en esa escuela).
2. Los primeros tres Candidatos de Agrupaciones Políticas distintas con mayor cantidad de votos para cada cargo.
3. Los Candidatos con mayor cantidad de votos dentro de cada Agrupación Política.

## Parte 2:

### Presentación

El objetivo de esta segunda parte es poner en práctica estructuras dinámicas.

### Ejercicio 1



Se necesita conocer el camino óptimo para retirar las urnas de los colegios y llevarlas hacia el centro de cómputos. Para el camino óptimo se debe utilizar las distancias entre los distintos colegios, sumadas a la demora, que se calcula con la siguiente ecuación:

**(Número Aleatorio entre 0 y 1) x (Velocidad Promedio del Móvil)**, que será ingresada al Sistema por el Usuario.

El camino más óptimo será entregado al conductor de la camioneta en una estructura dinámica que permita obtener cuál es el siguiente colegio por el que debe pasar.

### Ejercicio 2

Redactar un enunciado sobre un problema relacionado con las P.A.S.O. y luego resolverlo. El ejercicio debe contener como mínimo:

- Archivos
- Estructuras Dinámicas

- Patrones de Ordenamiento
- Patrones de Búsqueda



# Instructivo para Matemática Discreta

## Organización de los grupos

Cada grupo deberá auto-gestionarse, determinando un representante que se ocupará de las tareas administrativas del grupo. El código único de grupo será la manera en la que se identifiquen unívocamente a lo largo de todo el ciclo lectivo, ya sea para las entregas, como para las devoluciones, etc... El mismo, es otorgado por la coordinadora del TP, luego de que el representante envíe la conformación de su grupo al mail [danielalbello@gmail.com](mailto:danielalbello@gmail.com) respetando el formato de la plantilla ubicada en el campus a tal efecto. El formato del código tendrá la siguiente estructura: KCCCCXX, donde K es el prefijo de sistemas, CCCC es el número de curso y XX es el identificador único de grupo dentro del curso, valor determinado por orden partiendo del 01. Es tarea del representante gestionar las modificaciones en el grupo, enviar el trabajo práctico para su corrección y recibir los resultados de la misma.

## Contenido del trabajo práctico

El trabajo práctico consta de cuatro secciones, cada una de ellas correspondiente a un tema que se irá dando en la materia Matemática Discreta a lo largo del curso. En cada sección hay varias tareas con su correspondiente puntuación. Una tarea correcta, sumará los puntos que allí se indica.

La condición de aprobación para CADA sección es obtener 4 puntos sumados con todas las tareas presentadas que estén correctas.

Por lo tanto, deberán elegir las tareas que prefieran y que sumen al menos los 4 puntos solicitados, y presentarlas para su corrección.

## Entrega del trabajo práctico

Una vez que el equipo tenga el trabajo práctico terminado, lo deberá subir como tarea al espacio del Aula generado a tal fin. En el foro de novedades se publicarán las fechas de entrega oportunamente. El nombre del archivo deberá respetar la siguiente estructura: TP-MD-n-Kxxxxxx siendo:

n-> **1** para primera parte, **2** para segunda parte, **3** para entrega única de cursos cuatrimestrales

xxxxxx-> código de grupo

Los archivos deben estar en formato editable (.doc, .docx, etc...)

El/Los archivo/s adjuntos deben estar precedidos por una carátula que indique explícitamente los integrantes ACTUALES del grupo.

- TP ANUAL
- Primera Entrega: Sección 1 y Sección 2. La entrega estará habilitada desde el 16 de junio de 2017 a las 00:00 horas y hasta el 06 de julio de 2017 a las 23:59 horas. Luego se realizará la corrección y se comunicará el resultado.

- Segunda Entrega: Sección 3 y Sección 4. La entrega estará habilitada desde el 06 de octubre de 2017 a las 00:00 horas y hasta el 26 de octubre de 2017 a las 23:59 horas. Luego se realizará la corrección y se comunicará el resultado.
- TP CUATRIMESTRAL
- Entrega única: Sección 2 y Sección 4. La entrega estará habilitada desde el 06 de octubre de 2017 a las 00:00 horas y hasta el 26 de octubre de 2017 a las 23:59 horas. Luego se realizará la corrección y se comunicará el resultado.

# Sección 1: Lógica

## Presentación

El objetivo de esta sección es consolidar los conocimientos vistos en la lógica proposicional, ampliándolos y dejando lugar a la investigación. Los puntos prácticos incluyen utilizar temas de las elecciones legislativas P.A.S.O. y poder pasarlos al campo de la lógica.

## Tareas

- Buscar o generar dos silogismos lógicos reales (vinculados con el proceso electoral), estudiar sus valores de verdad, formalizarlos e indicar conclusiones (si se trata de uno real, indicar por qué; si es un 'falso silogismo' indicar cuál es el punto de fallo) -> **2 puntos**.
- Generar reglas lógicas (proposiciones) que a partir de un hecho (proposición de valor conocido), permita llegar a otro hecho. Este sistema de reglas, debe resolverse mediante deducción natural (Modus Ponens, Eliminación de la Conjunción, etc.) -> **2 puntos**.
- Investigar sobre Deducción Automática (la Teoría de Herbrand), elaborar ejemplos propios e intentar explicar con términos propios el propósito de esta teoría -> **2 puntos**.
- Investigar sobre Deducción Natural (uso de reglas Modus Ponens, Eliminación de la Conjunción, de la Disyunción, etc.). Generar ejemplos y explicar con términos propios las aplicaciones de esta teoría -> **2 puntos**.

**Tip:** para ver ejemplos de los **silogismos**, consultar la parte I del TP de la materia 2013, almacenado en el campus virtual. Sobre la tarea de **reglas lógicas**, podrán encontrar ejemplos y aplicaciones en los TP de 2012 y 2013. En el 'Anexo' del Punto I del TP de 2012, encontrarán un apunte simplificado muy útil para resolver los problemas que requieran **deducción natural**.

# Sección 2: Relaciones

## Presentación

En este apartado se proponen distintas aplicaciones orientadas particularmente a las relaciones de equivalencia, de orden y aplicaciones de relaciones de congruencia. El objetivo final de esta sección es ampliar sobre tópicos de la ingeniería en sistemas actuales y avanzados, usando métodos matemáticos de base aprendidos en la materia.

## Tareas

- Encriptar un mensaje redactado por ustedes (vinculado con las elecciones legislativas P.A.S.O.), utilizando un sistema de cifrado basado en corrimientos de caracteres (Encriptador del César). Realizar los cálculos correspondientes para encriptar o desencriptar los mensajes e indicar ventajas y desventajas del mismo -> **1 punto**.
- Encriptar un mensaje redactado por ustedes (vinculado con las elecciones legislativas P.A.S.O.), utilizando un sistema de cifrado RSA, basado en congruencias. Se debe armar el esquema desde el comienzo, armando la clave pública y la privada; los pasos para encriptar y los pasos para desencriptar los mensajes -> **2 puntos**.
- Generar una relación de equivalencia que dadas las contraseñas para ingresar a la terminales de impresión de Boletas (5 letras), éstas puedan dividirse en clases seguras, medias y débiles. La relación elegida puede manejarse con las posiciones de las palabras, longitud, etcétera. Presentar un caso práctico donde la relación genere las particiones deseadas -> **1 punto**.
- Investigar un método de ordenamiento algorítmico (Bubble Sort, Insertion Sort, Merge Sort, Heap Sort, Shell Sort, Quick Sort) indicando funcionamiento en términos propios y con ejemplos de funcionamiento. Asociarlo a la teoría de relaciones de orden -> **2 puntos**.

**Tips:** encontrarán un instructivo simplificado del **algoritmo RSA** en el 'Anexo' del punto III del TP del año 2012 en el campus virtual.

## Sección 3: Grafos

### Presentación

En esta sección se intenta destacar la importancia de los grafos como herramientas capaces de modelar situaciones complejas en representaciones simplificadas sobre los que pueden realizarse diversos análisis. Se introducen además diversos algoritmos operados sobre grafos.

#### Tareas

- Tomar un mapa de ciudades, estaciones o puntos específicos y remarcar sobre el mismo: ríos, caminos, rutas o vías que las unan para poder realizar la recolección de urnas por el Correo el día del sufragio. Asignar a cada uno de estos tramos, un valor único de referencia. Luego, seleccionar un punto origen y un punto destino y aplicar un algoritmo de camino más corto **sin heurísticas** para hallar formalmente el mejor camino entre ambos lugares -> **1 punto**.
- Igual al punto anterior, con la única diferencia de que se deberá definir una heurística para poder aplicar el algoritmo A\*. Definir la lista de nodos abiertos y cerrados en cada momento para indicar el funcionamiento del algoritmo -> **2 puntos**.

- Igual al punto anterior, definir la cantidad máxima de transportes, personas, barcos y otros, que pueden pasar por esos caminos y tomando dos ciudades, aplicar un algoritmo de flujo máximo para calcular numéricamente este valor para todo el tramo planteado -> **2 puntos**.
- Sabiendo que un 'HOP' es un salto de un router a otro y que el 'TTL' (time to live) es la cantidad máxima de saltos que puede dar un mensaje antes de ser eliminado; definir cuántas conexiones (vistas como si fueran cables) debería haber como mínimo entre las terminales de impresión de Boletas para que una red de 6 routers reciba todos los mensajes enviados por otro router, si el TTL vale 3 -> **1 punto**.

**Tips:** encontrarán ejemplos de flujo máximo y costes de camino en el TP de 2013. Encontrarán un apunte acerca de **HOPS y TTL** en el Anexo IV del TP de 2012.

## Sección 4: Lenguajes

### Presentación

En este apartado se busca agilizar el uso de los conceptos presentados en la materia respecto a lenguajes, gramáticas y autómatas con un fin mayormente práctico.

### Tareas

- Generar una gramática que cree contraseñas válidas para el acceso a las terminales de impresión de Boletas a partir de ciertas reglas de seguridad definidas en sus producciones. Utilizar criterios como los de contener una mayúscula, longitud dada y elementos alfanuméricos -> **1 punto**.
- Generar un autómata que sea capaz de reconocer contraseñas válidas para el acceso a las terminales de impresión de Boletas a partir de una serie de reglas definidas previamente. El autómata debe ser no determinístico -> **1 punto**.
- Investigar el funcionamiento de la máquina de Turing, dar ejemplos de usos y conclusiones respecto a los temas vistos en la materia -> **2 puntos**.
- Analizar el funcionamiento de un compilador de código y asociarlo con los temas vistos en la materia -> **1 punto**.
- Estudiar y aplicar en un ejemplo el método de transformación de un autómata finito no determinístico en un autómata finito determinístico -> **2 punto**.