

**Metodología-Datos Primitivos-**

**Estructuras de Control de Programas**

**Módulos-Estructuras de Datos:**

**Registros-Arreglos-Archivo de Texto.**

0 8 / 0 7 / 2025

\_ \_ / \_ \_ / \_ \_

\_ \_ / \_ \_ / \_ \_ \_ \_

\_ \_ / \_ \_ / \_ \_

**Supermercado Kotto**

**UTN.BA Trabajo Práctico # 1**

**Temas :**

**AyED**

**Curso: K1\_ \_ \_ \_ Dia: \_ \_ \_ \_ Turno: \_ \_ \_ \_**

**Fecha 1er.Entrega:**

**Fecha 2da. Entrega:**

**Fecha 3er. Entrega:**

**Nro.Leg.: Apellido, Nombres e-mail** 🖂

**Docente: Lic. Hugo Cuello. Es condición necesaria tener aprobado en tiempo y**

**forma el TP para promocionar la asignatura.**

**Observaciones: 🖆 -del docente-**

**Fecha T.P. aprobado: Firma:**

**Presentación de entrega de T.P.: ESTRUCTURAS DE DATOS ESTÁTICAS**

* Carpeta tamaño **A4** de tapa transparente y sujetador de hojas con gancho metálico, que puede estar recubierto en plástico. Entregar con el siguiente orden:
  1. Carátula (presentada anteriormente).
  2. Presentación del T.P. (esta misma, no modificarla).
  3. Enunciado del problema, emitir el archivo .docx original.
  4. Dibujos o gráficos de las estructuras de datos utilizadas y rotuladas con valores de inicialización, espacio de memoria requerida y un ejemplo.
  5. Estrategia diagramada (Bloque o Programa Principal).
  6. Algoritmo diagramado (módulos) indicando el pasaje de parámetros en la cabecera.
  7. Listado de la codificación en C++, sangría de dos espacios y escribir hasta una **columna 80 como máximo** para que no enrolle al siguiente renglón (listado fuente usando el Code Blocks, libre de errores de compilación), numerando las filas, tomar como modelo de estilo de escritura, los listados de los programas codificados en C++ entregado por el docente.

**Atención**: El código C++ de este listado debe coincidir exactamente con la versión entregada en penDrive, que se devolverá una vez copiado.

* 1. Además el pendrive deberá contener la Muestra de los datos y Resultados de la ejecución del programa, según muestra de datos de cada uno de los archivos, realizados por el grupo, emitidos en formato tabular generado por el proceso según indicado en el enunciado.
* El nombre del archivo fuente debe ser **TP1VnK1\_ \_ \_G\_-Apellido Nombre.cpp** Ej. de nombre para el T.P.1, **TP1V1K1023G3-Perez Juan.** Además se entregarán los archivos de datos de texto, y el archivo de salida de los listados.
* **La salida debe estar dirigida a archivo de texto en formato tabular –en columnas- y perfectamente alineado**, nada debe salir ni por pantalla ni por impresora, ni realizar pausas de espera.
* Los nombres de archivos físicos en el *modo de apertura* **no** deben tener indicado ni la unidad ni la ruta, debe ser por defecto, solo el nombre y la extensión del archivo, para el archivo de texto, **Txt**. Se deben respetar exactamente estos nombres físicos para los archivos indicados anteriormente como los indicados en el enunciado del problema a resolver. El nombre del archivo de salida debe ser Ticket.Txt conteniendo el Ticket del único Cliente y el Listado ordenado por Código de Rubro dentro de los cuales estarán los artículos correspondientes.

**Observaciones** :

Definir el registro de cada archivo de acuerdo a lo solicitado en las consignas del TP, con idéntico tipo, el mismo orden en que fueron mencionados. Una diferencia en orden o tipos hará que no funcione adecuadamente.

**ACLARACIÓN**: En los archivos de texto se van a leer/grabar los datos en forma individual y no toda la línea, por lo que, se van a utiliar tanto en la lectura como en la escritura, el formato r.c, en donde r es el nombre del registro y c indica el nombre del campo, y cada campo será del tipo de dato que se vaya a leer/grabar. Ej.: si una linea tiene 4 datos: 34 153.87 pala de punta 1

Definimos 4 campos de tipos y nombres ; short codArt float importe str20 descrip bool estado y siendo rArt el nombre de la de tipo struct y Art es el nombre lógico del archivo, podemos leer de la siguiente manera:

Art >> rArt.codArt >> rArt.importe; Art.get(rArt.descrip Art >> rArt.estado

**Todas las líneas deben tener la misma longitud física**. Esto es muy importante si en algún momento queremos realizar un acceso directo a una componene (línea) del archivo.

**IMPORTANTE**: Estas indicaciones se complementan con las consignas dadas en el enunciado del problema a resolver, bajo el título de **Observaciones, restricciones y recursos disponibles**.