

Indicaciones generales:

- El proyecto, es de **Programación Estructurada en C++**. No utilice clases y objetos.
 - El proyecto se desarrollará en grupos de 3 estudiantes. Los **nombres y secciones** de los integrantes deberán aparecer al inicio del **informe de proyecto**
 - Debe escoger **uno** de los 3 proyectos propuestos.
 - Los archivos de proyecto se subirán directamente a www.gradescope.com (**Proyecto 01**)
 - Solo un integrante del grupo entregará el proyecto. No olvide **registrar en Gradescope a los integrantes de proyecto**. En caso sea un grupo de distintas secciones, los nombres+secciones aparecerán en el informe
 - La entrega de proyecto será hasta el **martes 17 de mayo**, a medianoche
 - La sección 4 describe la **Entrega de proyecto** y la sección 5, la **Rúbrica**
-

1 Fabrica Textil

Te acaban de nombrar Gerente de Producción en una empresa textil, que tiene el problema de incumplimiento de pedidos a sus clientes. Para solucionar el problema, decides automatizar el proceso de producción de dos tipos de tela (jersey y franela). La empresa debe cumplir con un pedido de:

- 200 Polos (0.2 Kg/prenda) y 100 Camisas (0.5 Kg/prenda) de jersey
- 100 Polos (0.5 Kg/prenda) y 200 Camisas (1.0 Kg/prenda) de franela

Las etapas del proceso son:

- Materia prima: La materia prima es el hilo. Cada kg. de hilo para jersey produce 0.7 kgs. de tela, y para franela, 0.3 kg. de tela/kg hilo.
- La velocidad de tejido es de 1 Kg/hora, de teñido 10 Kg/hora, de acabado 20 Kg/hora y de confección, es para Jersey: 0.3 Kg/hora, y para Franela: 0.5 Kg/hora

El código debe calcular:

- El tiempo total de producción requerido para el pedido, en horas
- La cantidad de materia prima (en Kg. de hilo) necesaria para cumplir el pedido.
- Las ganancias esperadas (precio de venta - costos de producción) por tipo de tela, en Soles. Considere que los polos de jersey se venden a 30 Soles c/u, y las camisas a 50 Soles c/u., los polos de franela se venden a 60 Soles c/u, y las camisas a 80 Soles c/u. El Kg. de hilo cuesta 10 Soles.

Pregunta Opcional: (1pt):

Genere una lista de fechas de entrega, de prendas terminadas (por tipo de tela), en formato de fecha, contando desde el día de entrega del presente proyecto.

Como Gerente de Producción y observando los tiempos y ganancias obtenidos por tipo de tela, ¿Qué prendas recomendaría priorizar en la producción?

1.1 Recomendaciones para el proyecto

- **Diseñe el algoritmo:** defina las funciones necesarias para resolver el problema, utilizando un diagrama de flujo. Señale correctamente entradas y salidas de código. Puede tomar como ejemplo el siguiente enlace Flowchart
- **Optimice el uso de memoria** manejando la data en forma eficiente (utilice punteros). El *main()* debe consistir solo en la declaración/inicialización de variables y las llamadas a funciones.
- **Programe en C++** el código que solucione el problema planteado e imprima los resultados

2 Supply Chain Management (SCM)



Se pide desarrollar un software para el inventario de productos que se clasifican en **stock** (en almacén), y **en tienda**. El precio de los productos es un decimal random entre 1 y 500, y la cantidad de artículos por producto es un entero random entre 1 y 100. Los productos en stock o tienda están sujetos a un descuento del 10% , que se aplica si la cantidad de artículos del producto en stock o en tienda es mayor que 50.

El código debe:

- almacenar en memoria 200 productos (150 en stock y 50 en tienda) con sus características (precio y cantidad de artículos por producto)
- aplicar el descuento a todos los productos, de acuerdo al criterio mencionado.
- calcular la ganancia esperada por artículos en stock y en tienda con cantidad de artículos por producto menor a 80 y mayor a 20
- aplicar un descuento del 15 % si la cantidad de artículos por producto en stock o en tienda es mayor que 70. Y comparar este resultado con el resultado del descuento del 10 % ¿Cuál genera mayor ganancia? Comente el resultado
- ¿Cuál sería la ganancia si se aplica un costo de almacenamiento de 1% del precio por producto?

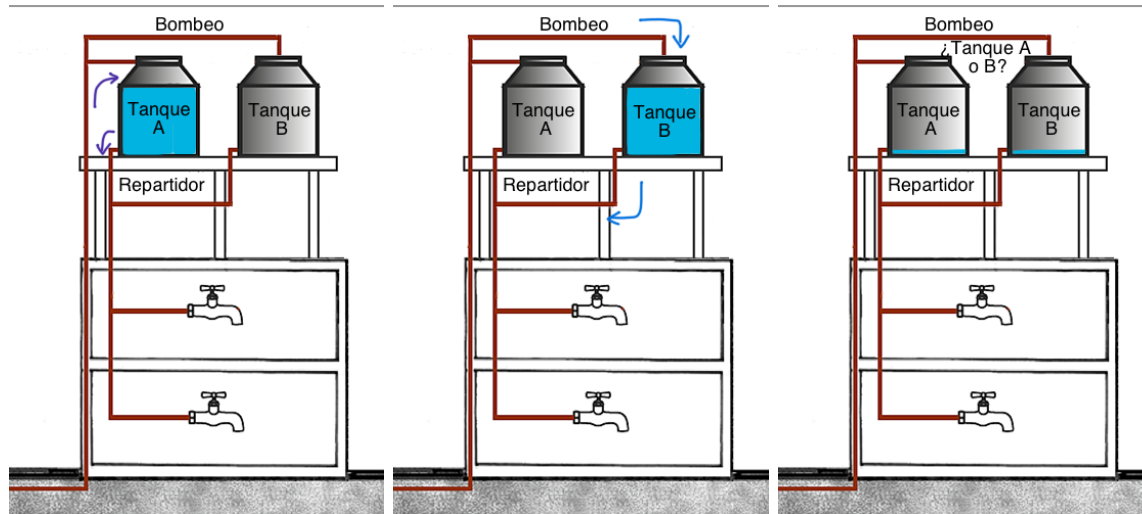
Pregunta Opcional: (1pt):

Implemente una función para trasladar un producto de stock a tienda. Traslade los productos de stock que tengan mas de 70 artículos por producto y aplique el descuento que genera mayor ganancia (10 o 15 %). Indique cuál es la ganancia esperada.

2.1 Recomendaciones para el proyecto

- **Diseñe el algoritmo:** defina las funciones y contenedores adecuados (arrays, vectores, matrices) para resolver el problema. Genere un diagrama de flujo. Puede tomar como ejemplo el siguiente enlace Flowchart

- **Optimice el uso de memoria** manejando la data en forma eficiente (utilice punteros). El *main()* debe consistir solo en la declaración/inicialización de variables y las llamadas a funciones.
- **Programa en C++** el código que solucione el problema planteado



3 Sistema de bombeo

En un condominio se planea un nuevo sistema de bombeo diario, y se te contrata para estimar la **cantidad de bombeos posibles** en **dos tanques** para poder abastecer a la mayor cantidad de departamentos en el edificio. Los **tanques**, tienen una cantidad inicial de 15 litros de agua. Los tanques son sujetos a **bombeo** y **reparticion** de agua a los departamentos. El **bombeo** consiste en suministrar un numero aleatorio de litros entre 1 y 10, la **reparticion** extrae también un numero aleatorio de litros de agua entre 1 y 10. El **bombeo** se hace inmediatamente luego de cada **reparticion**.

Primero se utiliza el **tanque A**, y cuando esta vacío o ya no puede repartir la cantidad requerida, el **tanque B**.

- calcule e imprima la **cantidad total de bombeos** logrados por ambos tanques (un numero entero que depende del momento en que ambos se queden vacíos)
- calcule e imprima la **cantidad de agua restante** en cada tanque (un numero entero que no debe ser negativo)
- calcule la cantidad de departamentos a los que se puede suministrar de agua en el condominio, si cada departamento consume 20 litros de agua al dia.

Considere que los tanques tienen una **capacidad ilimitada** de almacenamiento de agua. Además, los tanques se consideran **vacíos** cuando ya no puede ser agua extraída de ellos (es decir, no necesariamente deben tener 0 litros)

Pregunta Opcional: (1pt):

Considere que cuando ya no se puede extraer agua de los tanques, se puede bombear una vez mas el tanque que tenga mas agua restante. Entonces, éste último sigue repartiendo y siendo bombeado, hasta que finalmente, ambos tanques estén vacíos.

3.1 Recomendaciones para el proyecto

- **Diseñe el algoritmo:** defina las funciones necesarias para resolver el problema, utilizando un diagrama de flujo. Puede tomar como ejemplo el siguiente enlace [Flowchart](#)
- **Optimice el uso de memoria** manejando la data en forma eficiente (utilice punteros). El *main()* debe consistir solo en la declaración/inicialización de variables y las llamadas a funciones.
- **Programa en C++** el código que solucione el problema planteado

4 Entrega de proyecto

El proyecto se realizará en forma **grupal** (3 integrantes). En caso se haga el proyecto solo, o en grupo de dos, se deberá justificar el motivo

Se pide realizar un informe de los resultados del proyecto, que contenga lo siguientes:

- **Nombre del proyecto**, nombre y sección de los **integrantes**
- **Porcentaje de participación** de cada integrante en el proyecto (de 0 a 100 %)
- **Introducción** (descripción del problema)
- **Método** (diseño del algoritmo, donde debe mostrar y describir el **diagrama de flujo**)
- **Resultados** (descripción de partes relevantes del código, sus salidas y análisis de las mismas)
- **Conclusiones** (resumen del trabajo y posibles mejoras)

El proyecto se subirá a Gradescope (Proyecto 01)

Presente su proyecto en formato **.pdf**. Puede usar una plantilla de Plantillas Latex o convertir un documento a pdf. No olvide adjuntar el código de programación (en formato **.cpp**). Puede comprimir todos los documentos en un **zip** y subirlos a Gradescope

5 Rúbrica

Se utilizará la siguiente rúbrica para la calificación del proyecto

Criterio	Logrado	Parcialmente Logrado	No Logrado
Diseño del algoritmo	El diseño del algoritmo es correcto y corresponde a la implementación en C++ (1.5 pts)	Existen algunos errores menores en el diseño del algoritmo (1 pts)	Existen muchos errores en el diseño del algoritmo(0pts).
Código	El código no muestra errores, es ordenado y corresponde al algoritmo planteado (2 pts)	El código contiene errores menores de programación. (1 pts).	El código contiene múltiples errores de programación (0pts).
Presentación (informe escrito)	Describe los puntos del informe en forma ordenada y satisfactoria (1pts)	No describe en forma adecuada todos los puntos del informe(0.5pts).	No describe en forma adecuada ningún punto del informe (0pts).
Trabajo grupal	Se desarrollo el proyecto en un grupo de 3 (0.5pts)	El proyecto se hizo en un grupo de 2 (0.5 pts) ⁽¹⁾	Se hizo el proyecto solo (0pts) ⁽¹⁾ .

(1): En caso de no justificar el trabajo individual