

1. **(De regalo) 25P.** Escriba una aplicación que muestre los siguientes patrones por separado, uno debajo del otro. Use ciclos 'for' para generar los patrones. Todos los asteriscos (*) deben imprimirse mediante una sola instrucción de la forma `System.out.print('*');`, la cual hace que los asteriscos se impriman uno al lado del otro. Puede utilizar una instrucción de la forma `System.out.println();` para posicionarse en la siguiente línea. Puede usar una instrucción de la forma `System.out.print(' ');` para mostrar un espacio para los últimos dos patrones. No debe haber ninguna otra instrucción de salida en el programa. [Sugerencia: los últimos dos patrones requieren que cada línea empiece con un número apropiado de espacios en blanco].

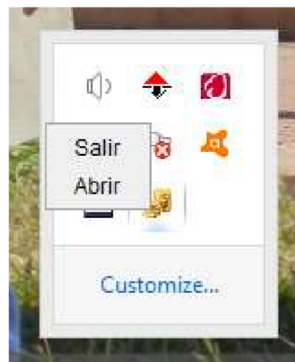
(a)	(b)	(c)	(d)
*	*****	*****	*
**	*****	*****	**
***	*****	*****	***
****	*****	*****	****
*****	*****	*****	*****
*****	*****	*****	*****
*****	*****	*****	*****
*****	****	****	*****
*****	***	***	*****
*****	**	**	*****
*****	*	*	*****

2. **(Calcular las ventas) 25P.** Un vendedor minorista vende cinco productos cuyos precios de venta son los siguientes: producto 1, Gs.10.000; producto 2, Gs. 15.000; producto 3, Gs.20.000; producto 4, Gs.25.000 y producto 5, Gs.30.000. Escriba una aplicación que lea una serie de pares de números, como se muestra a continuación:

- a) número del producto;
- b) cantidad vendida.

Su programa debe utilizar una instrucción switch para determinar el precio de venta de cada producto. Debe calcular y mostrar el valor total de venta de todos los productos vendidos. Use un ciclo controlado por centinela para determinar cuándo debe el programa dejar de iterar para mostrar los resultados finales.

3. **(Calculadora de suma) 25P.** Desarrolle una aplicación en Java que determine la suma de dos números ingresados por teclado. La aplicación debe ser una ventana que pueda contener dos campos de textos para introducir los valores y otro para mostrar el resultado, al pulsar la tecla enter en el segundo campo de texto de debe validar el ingreso de los datos y expresar el resultado en el tercer campo de texto. Al minimizar la aplicación, la misma se debe agregar a la bandeja del sistema (system tray o systray), también desde la bandeja del sistema se debe poder volver a visualizar la ventana. [Sugerencia: vea la implementación presentada en Ejemplo_eje3.zip].



4. **(Calculadora de costo de transacción)** Desarrolle una aplicación Java que determine si alguno de los clientes de una entidad financiera se ha excedido en el límite de transacciones gratuitas por mes en una cuenta. Para cada cliente se tienen los siguientes datos:

- a) El número de la cuenta.
- b) El total de todas las transacciones realizadas por el cliente en el mes.
- c) El límite de las transacciones gratuitas permitidas.

El programa debe recibir como entrada cada uno de estos datos en forma de números enteros, debe validar el total de las transacciones con el límite gratuito y establecer el costo de las transacciones cuando se supere dicho límite, estableciendo el costo en un 2% del monto de la transacción si la cantidad superada no alcanza 5(cinco) y en 5% si supera dicha cantidad.

Como resultado el programa debe mostrar el siguiente mensaje para los clientes que exceden el límite: "Se excedió el límite de operaciones gratuitas y el sistema generará un costo de x% sobre el monto de la transacción".