**E.E.T. Nº 2 - TRABAJO PRÁCTICO N°2 PROGRAMACION II – Herencia, Encapsulamiento y Polimorfismo**

**Enviar a: jeremias420@gmail.com**

De ser posible aplicar los conceptos de herencia, encapsulamiento o polimorfismo en los siguientes ejercicios:

1. Crear las clases **alumno**, **profesor** y **preceptor**. Se requiere lo siguiente:

1. Cada clase debe tener un método para mostrar los datos personales.
2. La clase **alumno** debe tener un método para calcular las faltas, si estas exceden las 20, debe mostrar el siguiente cartel "el alumno excedió el límite de faltas"
3. La Clase **profesor** debe poder calcular el sueldo del mismo aplicando una bonificación del 15% si trabajo más de 30 hs.
4. La clase **preceptor** debe tener un método para calcular el porcentaje de asistencia e inasistencia de los alumnos en un día, para ello este método solicita el cupo máximo de alumnos y la cantidad de alumnos que asistieron.

2. Crear las clases **Heladera**, **Cocina**, **Lavarropas**, las cuales comparten los siguientes atributos, largo, alto, marca, modelo, año de fabricación y años de garantía. Además, Se requiere lo siguiente:

1. Cada una debe tener un método para mostrar sus atributos, además de un método para el año en que vence la garantía.
2. La clase **Heladera** debe tener un método para calcular la cantidad de cubos de hielo producido por cubetera en n cantidad de minutos. Sabiendo que una cubetera produce 8 cubos de hielo en 25 min.El usuario deberá ingresar la cantidad de cubeteras y los minutos.
3. La clase **cocina** debe tener un método para calcular el tiempo de cocción dependiendo si el fuego está en mínimo o máximo de acuerdo a la siguiente tabla:

|  |  |  |
| --- | --- | --- |
|  | **Máximo** | **Mínimo** |
| (1)Verduras | 5 minutos | 10 minutos |
| (2)Pollo | 25 minutos | 35 minutos |
| (3)Carne | 40 minutos | 50 minutos |

1. La clase lavarropas debe permitir al usuario ingresar la cantidad de kg de ropa a lavar y el programa tendrá que calcular la combinación más optima (siguiendo la tabla) de lavados a realizar permitiendo asi ahorrar litros de agua. Por ejemplo, si el usuario ingresa 19 kg de ropa a lavar el sistema debería sugerir: 2 lavados Económicos y 1 Normal. Para ello hay que tener en cuenta la siguiente tabla:

|  |  |  |
| --- | --- | --- |
| **Tipo de Lavado** | **Litros** | **Kilogramos** |
| Económico | 9 | 6 |
| Normal | 12 | 7 |
| Full | 15 | 9 |

3. Crear las clases **empleado**, **cliente** y **artículo**. Donde cada clase tenga un método para mostrar sus atributos. También es necesario lo siguiente:

1. La clase empleado debe tener un método para calcular el salario. Sabiendo que cada 5 años de servicio se le realiza un aumento del 7%. Además de tener en cuenta que en el mes de diciembre y julio se paga el aguinaldo (el 50% del sueldo base).

Ejemplo:

sueldo base = 1000

antiguedad = 11 años

mes = 12 (diciembre)

el resultado deberia ser :

antiguedad = 140(14% por que son 10 años 5 y 5 y son 7% cada 5 años)

aguinaldo = 500 (50% sobre el sueldo base)

sueldo final = 1640 (1000 base + 140 antiguedad + 500 aguinaldo)

1. La clase cliente debe tener un método para calcular el total a pagar por compra pudiendo ingresarse n cantidad de artículos, solicitando por cada articulo el precio del artículo y la cantidad. Además, tener presente que si el total de la compra supera los 5000 pesos se aplica un descuento de 250 pesos, y si la misma es en efectivo se realiza un 15% de descuento.
2. La clase articulo debe tener un método para calcular el precio final dependiendo del margen de utilidad que se quiera obtener,

sabiendo que la utilidad se calcula de la siguiente forma: precio final - precio base = utilidad. El método solicita que se ingrese el % de utilidad.

4. Crear las clases **triangulo**, **rectángulo**, y **Rombo**. Donde cada clase tenga un método para calcular la superficie. También es necesario lo siguiente:

1. La clase **triangulo** tiene que tener un método que dibuje en pantalla un triángulo formado por asteriscos. El usuario deberá ingresar la altura del mismo, y se deberá mostrar en pantalla la cantidad de asteriscos utilizados en total. Ejemplo

|  |  |  |
| --- | --- | --- |
| Altura 6  \*  \*\*  \*\*\*  \*\*\*\*  \*\*\*\*\*  \*\*\*\*\*\* |  |  |

1. La clase **rectángulo** debe mostrar un rectángulo en pantalla. El usuario deberá ingresar la altura del rectángulo y este debera dibujarse en la pantalla con guiones (-), se dibujaran tantas filas de guiones como el número de altura ingresado por el usuario , pero hay que tener en cuenta que en el medio deberá dibujarse 2 líneas con una O en medio de esta forma ( -------OO---------) cuando la altura sea par sino se dibujará una sola. La línea diferente siempre será dibujada en el medio de la figura. Ejemplo

|  |  |
| --- | --- |
| Altura 5 (impar)  --------------------  --------------------  -------OO---------  --------------------  -------------------- | Altura 4 (par)  --------------------  -------OO---------  -------OO---------  -------------------- |

1. La clase **rombo** debe dibujar en pantalla un rombo. El usuario ingresara la altura del mismo. Tener en cuenta que la altura siempre tiene que ser par.

|  |  |  |
| --- | --- | --- |
| //\\  // \\  // \\  \\ //  \\ //  \\// |  |  |