Fundamentos de programación.

Tarea 2. Algoritmos y programas.

Antes de iniciar la tarea debes hacer fork al proyecto que se encuentra en:

<https://github.com/FundamentosProgramacion-201713/Tarea_02>

Después de hacer fork, clona el proyecto en tu computadora y completa este documento como se pide en cada uno de los **5** problemas.

Al terminar, debes subir a tu cuenta de github los cambios y hacer un **Pull Request** para que califique tu trabajo. **NO OLVIDES agregar tu nombre directa o indirectamente para que se registre tu calificación de manera correcta.**

**1.** Elabora un algoritmo y escribe un programa que muestre en la pantalla la siguiente información:

* Tu nombre completo.
* Tu matrícula.
* Tu carrera.
* Tu escuela de procedencia (Preparatoria, si es Prepa Tec el programa que cursaste)
* Dos o tres líneas que te describan de manera general (gustos, habilidades, deportes, libros preferidos, viajes, etc)

|  |
| --- |
| **Análisis**.  Entradas: (No hay)  Salidas: Nombre completo, Matrícula, Carrera, Escuela y programa, Descripción personal.  Relación E/S: (No hay, ya que no hay Entrada)  Algoritmo:   1. Imprimir mi Nombre completo 2. Imprimir Matrícula 3. Imprimir Carrera 4. Imprimir Escuela y programa 5. Imprimir Descripción personal.   El programa lo escribes directamente en el archivo en **github**, **miInfo.py**. |
| Ejemplo de salida:  Nombre:  Margarito Pérez  Matrícula:  A01112131  Carrera:  ISC  Escuela de procedencia:  Prepa Tec, programa Bicultural  Descripción:  Me gusta la tecnologia y todo lo relacionado con la computacion.  Practico el futbol americano y me gusta tocar la guitarra. |

**2.** La velocidad de un auto puede calcularse con la fórmula *v = d/t*. (v-velocidad, d-distancia, t-tiempo). Elabora un algoritmo y escribe un programa que pregunte al usuario la velocidad a la que viaja un auto (km/h) y calcule e imprima lo siguiente:

* La distancia en km. que recorre en 6 hrs.
* La distancia en km. que recorre en 10 hrs.
* El tiempo en horas que requiere para recorrer 500 km.

|  |
| --- |
| **Análisis**.  Entradas: Velocidad  Salidas: Distancia 6 hrs, Distancia 10 hrs, y Tiempo 500km  Relación E/S: Distancia 6 hrs = Velocidad \* 6 ; Distancia 10 hrs = Velocidad \* 10 ; Tiempo 500km= 500 / Velocidad.   1. Leer Velocidad 2. Distancia1 = Velocidad \* 6 3. Distancia2 = Velocidad \* 10 4. Tiempo = 500/Velocidad 5. Imprimir Distancia1 6. Imprimir Distancia2 7. Imprimir Tiempo   El programa lo escribes directamente en el archivo en **github**, **auto.py**. |
| Ejemplo de salida:  **Velocidad del auto en km/h: 115**  Distancia recorrida en 6 hrs: 690 km  Distancia recorrida en 10 hrs: 1150 km  Tiempo para recorrer 500 km: 4.3478260869565215 hrs. |

**3.** Elabora un algoritmo y escribe un programa que calcula el costo total de una comida en un restaurante.

* El programa le pregunta al usuario el total de la comida.
* Agrega 12% de propina y 16% de IVA.
* Cada porcentaje se calcula con respecto al costo de la comida.
* Imprime:
  + El subtotal (costo de la comida)
  + La propina.
  + IVA.
  + Total a pagar. (subtotal + propina + IVA)

|  |
| --- |
| **Análisis**.  Entradas: Total de la comida  Salidas: Subtotal, Propina, IVA, Total a pagar.  Relación E/S: Subtotal = Total de la comida, Propina = Total de la comida \* 0.12 , IVA = Total de la comida \* 0.16 , Total a pagar = Subtotal + Propina + IVA  Algoritmo:   1. Leer Total de la Comida 2. Subtotal = Total de la Comida 3. Propina = Total de la Comida \* 0.12 4. Impuesto = Total de la Comida \* 0.16 5. Total a pagar = Subtotal + Propina + Impuesto 6. Imprimir Subtotal 7. Imprimir Propina 8. Imprimir Impuesto 9. Imprimir Total a pagar   El programa lo escribes directamente en el archivo en **github, cuenta.py**. |
| Ejemplos de salida:  **Costo de su comida: 100**  Propina: $12.00  IVA: $16.00  Total a pagar: $128.00  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Costo de su comida: 255**  Propina: $30.60  IVA: $40.80  Total a pagar: $326.40 |

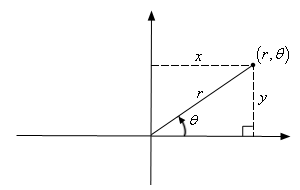
**4.** Elabora un algoritmo y escribe un programa que calcula el porcentaje de hombres y mujeres inscritos en una clase.

* El programa le pregunta al usuario el número de mujeres y el número de hombres inscritos.
* Imprime:
  + El número total de alumnos inscritos.
  + El porcentaje de mujeres.
  + El porcentaje de hombres.

|  |
| --- |
| **Análisis**.  Entradas: Número de Mujeres y Número de Hombres  Salidas: Número Total de Alumnos, Porcentaje de Mujeres y Porcentaje de Hombres  Relación E/S: Número Total de Alumnos = Número de Mujeres + Número de Hombres ; Porcentaje de Mujeres = Número de Mujeres \* 100 /Número Total de Alumnos ; Porcentaje de Hombres = Número de Hombres \* 100 /Número Total de Alumnos  Algoritmo:   1. Leer Num de Mujeres 2. Leer Num de Hombres 3. Num Total de Alumnos = Num de Mujeres + Num de Hombres 4. Porcentaje de Mujeres = Num de Mujeres \* 100 /Num Total de Alumnos 5. Porcentaje de Hombres = Num de Hombres \* 100 /Num Total de Alumnos 6. Imprimir Num Total de Alumnos 7. Imprimir Porcentaje de Mujeres 8. Imprimir Porcentaje de Hombres   El programa lo escribes directamente en el archivo en **github**, **porcentajes.py**. |
| Ejemplo de salida (por ahora no te preocupes por los acentos):    **Mujeres inscritas: 13**  **Hombres inscritos: 15**  Total de inscritos: 28  Porcentaje de mujeres: 46.4%  Porcentaje de hombres: 53.6% |

5. Elabora un algoritmo y escribe un programa que convierta de coordenadas cartesianas a coordenadas polares. Usa la función *atan2(y,x)* en Python que regresa el arcotangente de y/x en el rango -π a π.

* El programa le pregunta al usuario el valor de *x* y *y*.
* Imprime:
  + El valor de la magnitud *r*.
  + El valor del ángulo *θ* en grados.



|  |
| --- |
| Análisis.  Entradas: Valor de x & y  Salidas: Magnitud y Ángulo  Relación E/S: Ángulo = arctan (y/x) ; Magnitud = y / sin( Ángulo)  Algoritmo:   1. Leer valor de x 2. Leer valor de y 3. Ángulo = arctan (y/x) 4. Magnitud = y / sin (Ángulo) 5. Ángulo 1 = Ángulo \* (180/pi) 6. Imprimir Magnitud 7. Imprimir Angulo 1     **Crea el programa desde cero y lo agregas al repositorio** en **github, coordenadas.py.** |
| Ejemplo de salida (por ahora no te preocupes por los acentos):  **x: 6**  **y: 4**  Magnitud: 7.211102550927978  Angulo: 33.690067525979785 |