Fundamentos de programación.

Tarea 2. Algoritmos y programas.

Nombre: Alexys Martín Coate Reyes

Grupo: 04

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

https://github.com/FundamentosProgramacion-2018-13/Mision-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 archivos .py y este documento modificado, y hacer un **Pull request** para que califique tu trabajo. **NO OLVIDES agregar tu matrícula, nombre y grupo 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: texto  Relación E/S: (No hay)  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Imprimir texto (definido en la instrucción print)   El programa lo escribes directamente en el archivo de **github**, **miInfo.py**. |
| Ejemplo de salida:  Nombre:  Margarito Pérez  Matrícula:  A01112131  Carrera:  ISC  Escuela de procedencia:  Prepa Tec, programa Bicultural  Descripción:  Me gusta la tecnología y todo lo relacionado con la computación.  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 7 hrs.
* La distancia en km. que recorre en 4.5 hrs.
* El tiempo en horas que requiere para recorrer 791 km.

|  |
| --- |
| **Análisis.**  Entradas: v (velocidad)  Salidas: R1, R2, R3  Relación E/S: R1 = v\*t1 ; R2 = v\*t2 ; R3 = d/v  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Introducir la velocidad 2. Crear y definir t1 = 7, t2 = 4.5, d = 791 3. Definir la relación entrada salida de: R1 = v\*t1 ; R2 = v\*t2 ; R3 = d/v 4. Imprimir la velocidad del auto, la distancia recorrida en t1, la distancia recorrida t2 y el tiempo que requiere recorrer en d.   El programa lo escribes directamente en el archivo de **github**, **auto.py**. |
| Ejemplo de salida: (Siempre imprime un decimal)  **Velocidad del auto en km/h: 115**  Distancia recorrida en 7 hrs: 805.0 km  Distancia recorrida en 4.5 hrs: 517.5 km  Tiempo para recorrer 791 km: 6.8782 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 13% de propina y 15% de IVA.
* Cada porcentaje se calcula con respecto al costo de la comida.
* Imprime:
  + El subtotal (costo de la comida)
  + La propina.
  + El IVA.
  + El total a pagar. (Suma del subtotal, la propina y el IVA)

|  |
| --- |
| **Análisis**.  Entradas: subtotal  Salidas: total, propina, iva, subtotal  Relación E/S: total = subtotal + iva + propina ; propina = subtotal\*.13 ; iva = subtotal\*.15 ; subtotal = subtotal  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer subtotal 2. Calcular propina 🡨 subtotal\*.13 3. Calcular iva 🡨subtotal\*.15 4. Calcular total 🡨 subtotal + iva +propina 5. Imprimir Total, subtotal, propina e iva   El programa lo escribes directamente en el archivo de **github, cuenta.py**. |
| Ejemplos de salida: (mostrar centavos en las cantidades)  **Costo de su comida: 100**  Propina: $13.00  IVA: $15.00  Total a pagar: $128.00  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Costo de su comida: 255**  Propina: $33.15  IVA: $38.25  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: hombres, mujeres  Salidas: total, pHombres, pMujeres  Relación E/S: total = hombres + mujeres ; pHombres = (hombres\*100)/total ; pMujeres = (mujeres\*100)/total  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)     1. Ingresar el número de hombres y mujeres 2. Calcular el total 🡨 hombres + mujeres 3. Calcular pHombres 🡨(hombres\*100)/total 4. Calcular pMujeres 🡨 (mujeres\*100)/total 5. Imprimir total, pHombres y pMujeres   El programa lo escribes directamente en el archivo de **github**, **porcentajes.py**. |
| Ejemplo de salida (un decimal en los porcentajes):    **Mujeres inscritas: 9**  **Hombres inscritos: 13**  Total de inscritos: 22  Porcentaje de mujeres: 40.9%  Porcentaje de hombres: 59.1% |

5. Elabora un algoritmo y escribe un programa que calcula la distancia entre dos puntos.

* El programa le pregunta al usuario las coordenadas (x1, y1) del primer punto y, también, las coordenadas (x2, y2) del segundo punto.
* Imprime:
  + La distancia entre los dos puntos.

 

|  |
| --- |
| **Análisis**.  Entradas: x1, x2, y1, y2  Salidas: distancia  Relación E/S: distancia = ((x2-x1)\*\*2 + (y2-y1)\*\*2)\*\*(1/2)  Inserta aquí la imagen con el algoritmo. (foto, captura de pantalla, texto, etc.)   1. Leer x1, x2, y1, y2 2. Calcular distancia 🡨 ((x2-x1)\*\*2 + (y2-y1)\*\*2)\*\*(1/2) 3. Imprimir distancia   **Crea un nuevo archivo para este programa y lo agregas al repositorio** de **github, coordenadas.py.** |
| Ejemplo de salida (4 decimales en el resultado):  **x1: 6**  **y1: 4**  **x2: 10**  **y2: 5**  Distancia: 4.1231 |

**Evaluación.**

La tabla muestra los HP que conservarás si resuelves correctamente cada ejercicio.

|  |  |  |  |
| --- | --- | --- | --- |
| Problema | Análisis | Algoritmo | Programa |
| 1 | 90 | 90 | 70 |
| 2 | 90 | 90 | 70 |
| 3 | 90 | 90 | 70 |
| 4 | 90 | 90 | 70 |
| 5 | 90 | 90 | 70 |

Si finalizas exitosamente la misión, conservarás 1250 HP.