Fundamentos de programación.

Tarea 2. Algoritmos y programas.

Nombre: Michelle Ojeda Manjarrez

Grupo: 02

Completa este documento como se pide en cada uno de los **5** problemas. El análisis y algoritmo se agregan en este mismo documento. Los programas se escriben en los archivos .py que se descargaron desde github.

Al terminar, debes:

* Subir a tu cuenta de github:
  + Los archivos .py (5 archivos, uno por cada problema).
  + Este documento con la información solicitada.
* Hacer un **Pull request** para que califique tu trabajo. **NO OLVIDES agregar tu nombre 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 (Si es Prepa Tec, el programa que cursaste)
* Dos o tres líneas que te describan de manera general (gustos, habilidades, deportes, libros preferidos, viajes, pasatiempos, etc.)

|  |
| --- |
| **Análisis**.  Entradas: (No hay)  Salidas: nombre, matricula, carrera, escuela y descripción.  Relación E/S: (No hay)  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)     1. Imprimir datos   El programa lo escribes directamente en el archivo de **que descargaste**, **misDatos.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, número entero) y calcule e imprima lo siguiente:

* La distancia en km. que recorre en 6 hrs.
* La distancia en km. que recorre en 3.5 hrs.
* El tiempo en horas que requiere para recorrer 485 km.

|  |
| --- |
| **Análisis.**  Entradas: la velocidad a la que viaja un auto  Salidas: velocidad, distancia y tiempo  Relación E/S: v = d/t  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer la velocidad en la que viaja el auto 2. Calcular la velocidad v= d/t, distancia = v\*t, tiempo =v/d 3. Imprimir velocidad, distancia, tiempo   El programa lo escribes directamente en el archivo de **github**, **velocidad.py**. |
| Ejemplo de salida: (Siempre imprime un decimal)  **Velocidad del auto en km/h: 95**  Distancia recorrida en 6 hrs: 570.0 km  Distancia recorrida en 3.5 hrs: 332.5 km  Tiempo para recorrer 485 km: 5.1 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 16% 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: total de la comida  Salidas: Subtotal, propina, IVA, total a pagar  Relación E/S: propina = total\_comida \* .13  IVA = total \_comida \*.16  Total\_pagar = total\_ comida + propina + iva  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer el total de la comida 2. Calcular   propina = total\_comida \* .13  IVA = total \_comida \*.16  Total\_pagar = total\_ comida + propina + iva   1. Imprimir total\_comida, propina, IVA, total\_pagar   El programa lo escribes directamente en el archivo de **github, cuenta.py**. |
| Ejemplos de salida: (mostrar centavos en las cantidades (siempre dos decimales))  **Costo de su comida: 100**  Propina: $13.00  IVA: $16.00  Total a pagar: $129.00  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Costo de su comida: 255**  Propina: $33.15  IVA: $40.80  Total a pagar: $328.95 |

**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 inscritas.
  + El número de hombres inscritos.
* Imprime:
  + El número total de alumnos inscritos.
  + El porcentaje de mujeres.
  + El porcentaje de hombres.

|  |
| --- |
| **Análisis**.  Entradas: el número de mujeres inscritas, el número de hombres inscritos.  Salidas: El número total de alumnos inscritos, porcentaje de mujeres, porcentajes de hombres  Relación E/S: total de inscritos = mujeres inscritas + hombres inscritos  Porcentaje de mujeres = total de alumnos \* 100 /mujeres inscritas  Porcentaje de hombres = total de alumnos \* 100 / hombres inscritos  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer el número de mujeres inscritas, el número de hombres inscritos 2. Total de inscritos = mujeres inscritas + hombres inscritos   Porcentaje de mujeres = total de alumnos \* 100 /mujeres inscritas  Porcentaje de hombres = total de alumnos \* 100 / hombres inscritos   1. Imprimir mujeres inscritas, hombres inscritos, total de inscritos, porcentaje de mujeres, porcentaje de hombres   El programa lo escribes directamente en el archivo de **github**, **clase.py**. |
| Ejemplo de salida (un decimal en los porcentajes):    **Mujeres inscritas: 14**  **Hombres inscritos: 12**  Total de inscritos: 26  Porcentaje de mujeres: 53.8%  Porcentaje de hombres: 46.2% |

**Extras.**

a) 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, y1) y (x2, y2)  Salidas: Distancia entre los dos puntos  Relación E/S: d= ((x2-x1)\*\*2 + (y2-y1)\*\*2 )\*\*0.5  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer x1, y1, x2, y2 2. Calcular distancia= ((x2-x1)\*\*2 + (y2-y1)\*\*2 )\*\*0.5 3. Imprimir distancia   **Crea un nuevo archivo para este programa y lo agregas al repositorio** de **github, extraDistanciaPuntos.py.** |
| Ejemplo de salida (3 decimales en el resultado):  **x1: 6**  **y1: 4**  **x2: 10**  **y2: 5**  Distancia: 4.123 |

c) Construye un algoritmo para preguntar al usuario cuántas galletas quiere elaborar e indique la cantidad de ingredientes que se requieren.

Sabemos que, para 48 galletas, se requiere: 1.5 tazas de azúcar, 1 taza de mantequilla, 2.75 tazas de harina.

|  |
| --- |
| **Análisis**.  Entradas: Cuántas galletas quiere elaborar  Salidas: Cuántas tazas de azúcar, de mantequilla y de harina se requieren.  Relación E/S: azúcar = galletas x 1.5 / 48  mantequilla = galletas /48  harina = galletas x 2.75 / 48  Inserta aquí la imagen con el **algoritmo**. (foto, captura de pantalla, texto, etc.)   1. Leer el número de galletas 2. Calcular: azúcar = galletas \*1.5 / 48   mantequilla = galletas /48  harina = galletas \* 2.75 / 48   1. Imprimir el número de galletas, las tazas de azúcar, de mantequilla y de harina   **Crea un nuevo archivo para este programa y lo agregas al repositorio** de **github, extraGalletas.py.** |
|  |
|  |

**Evaluación.**

La tabla muestra los HP que ganarás si resuelves correctamente cada ejercicio. Los extras aportan XP

|  |  |  |  |
| --- | --- | --- | --- |
| Problema | Análisis | Algoritmo | Programa |
| 1 | 100 | 100 | 100 |
| 2 | 100 | 100 | 100 |
| 3 | 100 | 100 | 100 |
| 4 | 100 | 100 | 100 |
| a), b) y c) | 15 | 15 | 20 |

Si finalizas exitosamente la misión, ganarás 1200 HP y hasta 150 XP. Además, los primeros 5 en entregarla obtienen 100 XP.