**Evaluación Parcial 3**

**Nombre: Arreglos y Funciones\_ Forma D**

| **Sigla** | **Nombre Asignatura** | **Tiempo Asignado** | **% Ponderación** |
| --- | --- | --- | --- |
| **PGY1121** | **Programación de Algoritmos** | **120 minutos** | **35%** |

1. **Agente evaluativo**

| **X** | **Heteroevaluación** |  |  | **Coevaluación** |  |  | **Autoevaluación** |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. **Tabla de especificaciones**

| **Unidad de Competencia** | **Indicador de Logro (IL)** | **Indicador de Evaluación (IE)\*** | **Ponderación Indicador Logro** | **Ponderación Indicador de Evaluación** |
| --- | --- | --- | --- | --- |
| Desarrolla pensamiento lógico-analítico para la construcción de algoritmos para soportar los requerimientos | Utiliza arreglos y/o listas que permitan el almacenamiento de datos según los requerimientos del problema planteado. | Crea el menú de opciones de forma correcta. | 25% | 10% |
| Declara arreglo(s) para almacenar los datos del caso planteado | 15% |
| Define funciones que permitan optimizar el código del problema planteado. | Ingresa los datos declarados en el caso | 25% | 10% |
| Crea la función para grabar los datos. | 15% |
| Integra funciones que son invocadas desde el programa principal para dar solución al requerimiento del problema planteado. | Crea la función para buscar persona, de acuerdo a un dato específico. | 25% | 10% |
| Crea la función para imprimir los certificados solicitados | 15% |
| Utiliza parámetros y retornos necesarios | Crea la función para salir del programa | 25% | 10% |
| Graba y sube el programa en GitHub | 15% |
| **Total** | | | **100%** | **100%** |

1. **Instrucciones para el/la estudiante**

| Esta es una evaluación que corresponde a una **Ejecución Práctica** y tiene un **35%**deponderaciónsobre la nota final de la asignatura. El **tiempo** para desarrollar esta evaluación es de 2 horas en la **semana 17** y se realiza de manera **individual** en  **Taller de PC Avanzado Laboratorio de soporte (A).**  **La evaluación consiste en:**   * Construir soluciones de algoritmos de acuerdo con las instrucciones necesarias que den solución al requerimiento del cliente   **Contexto y requerimiento:**  La peluquería “Mechas Locas” desea manejar el registro de atenciones que realiza en su centro de belleza, para ello necesita registrar de cada una de las citas, numero de atención, el nombre del cliente, fecha, hora, descripción atención y costo. Aquellas atenciones con un costo mayor a 45000 pesos tendrán un masaje facial gratis.  El código de atención posee la siguiente estructura:   * A-001 * A-002 * B-022   En el registro y administración de cada cita debe construir una aplicación que contenga un menú principal con las siguientes opciones disponibles:  Opción 1   * Grabar. Corresponde a guardar una cita con los datos anteriormente señalados. Además, debe verificar que el código de atención sea el correcto, el nombre del cliente debe tener un mínimo de 5 caracteres, la descripción debe estar entre 5 a 45 caracteres y el precio debe ser mayor o igual a 20000.   Opción 2   * Buscar: Corresponde buscar una cita por su código y mostrar toda su información almacenada.   Además, debe mostrar si la atención tendrá masaje gratis o no  Opción 3   * Imprimir atención: Esta opción debe permitir imprimir boleta atención o cita gratis ingresando el número de atención. Para cada uno de los casos debe presentar el listado con todos los datos de la cita, además presente una propina aleatoria la cual puede ser un número entre 100 y 1000 pesos para aquellos que tengan masaje gratis.   Opción 4   * Salir. Corresponde a salir del programa emitiendo un mensaje de salida. Considere, además, su nombre y apellido y la versión del programa.   **Instrucciones Generales:**  Escribir un programa que contenga:   1. Diseñe un menú con las opciones para acceder a cada función requerida. 2. Cree las funciones solicitadas por cada requerimiento 3. Considere el ingreso de datos y el despliegue de información.   **Entrega:**   * Para la entrega debe guardar el archivo del programa con el siguiente formato nombreAlumno\_PGY1121\_SECCION\_JORNADA y súbalo a GitHub.   **Ejemplo:** JuanPerez\_PGY1121\_001\_D |
| --- |

**Pauta de Evaluación**

**Pauta tipo: Escala de valoración**

| **Categoría** | **% logro** | **Descripción niveles de logro** |
| --- | --- | --- |
| **Muy buen desempeño** | **100%** | Demuestra un desempeño destacado, evidenciando el logro de todos los aspectos evaluados en el indicador. |
| **Buen desempeño** | **80%** | Demuestra un alto desempeño del indicador, presentando pequeñas omisiones, dificultades y/o errores. |
| **Desempeño aceptable** | **60%** | Demuestra un desempeño competente, evidenciando el logro de los elementos básicos del indicador, pero con omisiones, dificultades o errores. |
| **Desempeño incipiente** | **30%** | Presenta importantes omisiones, dificultades o errores en el desempeño, que no permiten evidenciar los elementos básicos del logro del indicador, por lo que no puede ser considerado competente. |
| **Desempeño no logrado** | **0%** | Presenta ausencia o incorrecto desempeño. |

| **Indicador de Evaluación** | **Categorías de Respuesta** | | | | | **Ponderación del Indicador de Evaluación** |
| --- | --- | --- | --- | --- | --- | --- |
| **Muy buen desempeño**  **100%** | **Buen desempeño**  **80%** | **Desempeño aceptable**  **60%** | **Desempeño incipiente**  **30%** | **Desempeño no logrado**  **0%** |
| Crea el menú de opciones de forma correcta. |  |  |  |  |  |  |
| Declara arreglo(s) para almacenar los datos del caso planteado |  |  |  |  |  |  |
| Ingresa los datos declarados en el caso |  |  |  |  |  |  |
| Crea la función para grabar los datos. |  |  |  |  |  |  |
| Crea la función para buscar persona, de acuerdo a un dato específico. |  |  |  |  |  |  |
| Crea la función para imprimir los certificados solicitados |  |  |  |  |  |  |
| Crea la función para salir del programa |  |  |  |  |  |  |
| Graba y sube el programa en GitHub |  |  |  |  |  |  |
| **Total** | | | | | | **100%** |