

**Instituto Nacional de Aprendizaje**

**Curso**

**Programación de Aplicaciones Informáticas.**

**Módulo**

**Lógica computacional**

**Actividad**

**Práctica 10**

**Estudiante:**

**Mariel Daniela Rojas Sanchez**

**Profesor:**

**Oscar Pacheco Vásquez**

**2022**

**PLANTILLA PARA LA SOLUCIÓN DE PROBLEMAS**

**Propuesta de***:* Mariel Daniela Rojas Sanchez.

|  |
| --- |
| **Planteamiento del Problema** |
| Crear un algoritmo que imprima todos los divisores de un número ingresado por el usuario. |

|  |  |
| --- | --- |
| **Identificación del problema** | |
| ***Datos de entrada*** | ***Datos de Salida*** |
| 1. Número ingresado. | 1. Cantidad de divisores que contiene el número ingresado. 2. Divisores del número. |
| **Observaciones *(Limitaciones / Excepciones / Suposiciones)*** | |
| Únicamente se imprimen los números que si son divisores del número digitado, los que no son, se descartan. | |
|  | |
| **Planteamiento de alternativas** | |
| ***Alternativa # 1*** | ***Alternativa #2*** |
| 1. Inicio. 2. Se define num como entero. 3. Después del algoritmo principal se crea un SubProceso llamado numDigi, en el cúal se pone de parámetro el valor de num Por referencia, y en su bloque de instrucciones se le solicita al usuario que digite un número y ese número se lee y se llama a imprimir en el Algoritmo principal. 4. Fin de Subproceso. 5. Después del último subproceso, o sea, fuera del algoritmo principal se crea una función llamada dividir en el cual se pone de parámetro el valor de num. 6. Se declaran la variables cantidadDiv,n Como Entero. 7. En el bloque de instrucciones la variable cantiDiv se inicializa en cero. 8. En un ciclo para n hasta el número que digito el usuario, donde si el numero ingresado al dividirlo da como residuo cero, entonces, dicho número por el que se dividió es un divisor, entonces la variable cantiDiv va a aumentarse. 9. Se imprime el valor de n, el son los divisores. 10. Fin de función. 11. En el algoritmo principal se llama a imprimir la función Dividir(num), para que se impriman la cantidad de divisores. 12. Fin de Algoritmo principal. | 1. Inicio 2. Se define num,cantidadDiv,n como entero. 3. Se le solicita al usuario que digite un número. 4. Se lee el número. 5. La variable cantiDiv se inicializa en cero. 6. En un ciclo para n hasta el número que digito el usuario, donde si el numero ingresado al dividirlo da como residuo cero, entonces, dicho número que divide es un divisor del número ingresado, entonces la variable cantiDiv va a aumentarse. 7. Se imprime la cantidad de divisores del número ingresado. 8. Fin de Algoritmo. |

|  |  |
| --- | --- |
| **Elección de Alternativa** | |
|  | |
| ***Alternativa Seleccionada*** | ***Justificación*** |
| Alternativa #1 | Se elige la alternativa 1, ya que, al estar modularizada con un subproceso y una función, el algoritmo se divide en partes más simples, siendo el algoritmo más eficaz y fácil de trabajar. |

Ejercicio 1. DivisoresCantidad

Diagrama

Descripción generada automáticamente

**PLANTILLA PARA LA SOLUCIÓN DE PROBLEMAS**

**Propuesta de***:* Mariel Daniela Rojas Sanchez.

|  |
| --- |
| **Planteamiento del Problema** |
| Crear un algoritmo que reciba como parámetro un texto y lo escriba con un espacio adicional tras cada letra. |

|  |  |
| --- | --- |
| **Identificación del problema** | |
| ***Datos de entrada*** | ***Datos de Salida*** |
| 1. Texto ingresado | 1. Texto ingresado con un espacio tras cada letra. |
| **Observaciones *(Limitaciones / Excepciones / Suposiciones)*** | |
| Funciona tanto para texto, como para números. | |
| **Planteamiento de alternativas** | |
| ***Alternativa # 1*** | ***Alternativa #2*** |
| 1. Inicio. 2. Se define palabra,letra,espacio como carácter. 3. Se define i, tam Como Entero 4. Después del algoritmo principal se crea un SubProceso llamado Ingreso, en el cúal se pone de parámetro el valor de Palabra, por referencia, y en su bloque de instrucciones se le solicita al usuario que digite palabra por separar y esa palabra se lee. 5. Fin de Subproceso1. 6. Después del último subproceso, o sea, fuera del algoritmo principal se crea otro subproceso llamado Espaciadodividir en el cual se ponen los siguientes parámetros (letra,palabra,i,tam) . 7. En el bloque de instrucciones a la variable tam se asigna la Longitd de la Palabra ingresada. 8. En un ciclo repetir hasta que la posición sea igual al tamaño de la palabra, se dice que letra es igual a la subcadena de la posicion i,i, 9. Se imprime cada letra, con un espaciado entre ellas, y se aumenta el valor de la posicion para pasar por cada letra de la palabra ingresada. 10. Fin de Subproceso2 11. En el algoritmo principal se llama a imprimir la función del subproceso de ingreso(palabra) y el de Espaciado(letra,palabra,i,tam)   para que se impriman en la pantalla.   1. Fin de Algoritmo principal. | 1. Inicio. 2. Se define palabra,letra,espacio como carácter. 3. Se define i, tam Como Entero 4. Se le solicita al usuario que digite palabra por separar y esa palabra se lee. 5. A la variable tam se asigna la Longitd de la Palabra ingresada. 6. En un ciclo repetir hasta que la posición sea igual al tamaño de la palabra, se dice que letra es igual a la subcadena de la posicion i,i. 7. Se imprime cada letra, con un espaciado entre ellas, y se aumenta el valor de la posición para pasar por cada letra de la palabra ingresada. 8. Fin de Algoritmo. |

|  |  |
| --- | --- |
| **Elección de Alternativa** | |
|  | |
| ***Alternativa Seleccionada*** | ***Justificación*** |
| Alternativa #1 | Se elige la alternativa 1, ya que, al estar conformado con subprocesos, el algoritmo se divide en partes más simples y se va resolviendo parte por parte, siendo el algoritmo más fácil de entender y trabajar |

Ejercicio2: EscribirEspaciado

Diagrama

Descripción generada automáticamente

**PLANTILLA PARA LA SOLUCIÓN DE PROBLEMAS**

**Propuesta de***:* Mariel Daniela Rojas Sanchez.

|  |
| --- |
| **Planteamiento del Problema** |
| Crear una función que devuelva una cadena formada por las iniciales de una frase digitada, se debe mostrar la inicial de la primera letra y la letra que haya tras cada espacio. |

|  |  |
| --- | --- |
| **Identificación del problema** | |
| ***Datos de entrada*** | ***Datos de Salida*** |
| 1. Frase | 1. Iniciales de la frase digitada. |
| **Observaciones *(Limitaciones / Excepciones / Suposiciones)*** | |
| Funciona para números y texto. | |
|  | |
| **Planteamiento de alternativas** | |
| ***Alternativa # 1*** | ***Alternativa #2*** |
| 1. Inicio. 2. Se define frase como carácter. 3. Después del algoritmo principal se crea un SubProceso1 llamado Solicitar, en el cúal se pone de parámetro el valor de frase, por referencia, y en su bloque de instrucciones se le solicita al usuario que digite una frase. 4. Se lee frase. 5. Fin de Subproceso1. 6. Después del último subproceso, o sea, fuera del algoritmo principal se crea otro subproceso llamado Iniciales en el cual se pone frase como parámetro. 7. Se define i,x,tam como carácter. 8. Se le asigna al tam la Longitd de la frase ingresada. 9. x se iguala a cero. 10. En un ciclo para i=0 para hasta que la posición sea igual al tamaño de la frase,Si i es igual al tamaño entonces, escribe una subcadena con el valor de la frase y su posición, Si no, si subcadena y la frase se pone en cada posicion, entonces se debe inicializar x igual a 1, sino, si x es igual a 1 se escribe el valor de la subcadena son su frase y posicione y se inicializa x igual a cero. 11. Se imprime cada inicial de la frase. 12. Fin de Subproceso2 13. En el algoritmo principal se llama a imprimir la función del subproceso de ingreso(palabra) y el de Espaciado(letra,palabra,i,tam)   para que se impriman en la pantalla.   1. Fin de Algoritmo principal. | 1. Inicio. 2. Se define frase,i,x,tam como carácter. 3. Se le solicita al usuario que digite una frase. 4. Se lee frase.   Se le asigna al tam la Longitd de la frase ingresada.   1. x se iguala a cero. 2. En un ciclo para i=0 para hasta que la posición sea igual al tamaño de la frase,Si i es igual al tamaño entonces, escribe una subcadena con el valor de la frase y su posición, Si no, si subcadena y la frase se pone en cada posición, entonces se debe inicializar x igual a 1, sino, si x es igual a 1 se escribe el valor de la subcadena son su frase y posicione y se inicializa x igual a cero. 3. Se imprime cada inicial de la frase. 4. Fin de Algoritmo principal. |

|  |  |
| --- | --- |
| **Elección de Alternativa** | |
|  | |
| ***Alternativa Seleccionada*** | ***Justificación*** |
| Alternativa #1 | Se elige la alternativa 1, ya que, al estar conformado con subprocesos, el algoritmo se divide en partes más simples y se va resolviendo parte por parte, siendo el algoritmo más fácil de entender y trabajar, más efectivo y se ahorra tiempo a la hora de hacer prueba y error. |

Ejercicio3: InicialesFrase

**Diagrama

Descripción generada automáticamente**

**PLANTILLA PARA LA SOLUCIÓN DE PROBLEMAS**

**Propuesta de***:* Mariel Daniela Rojas Sanchez.

|  |
| --- |
| **Planteamiento del Problema** |
| Crear un algoritmo que permita leer dos números y determinar si son amistosos o no. |

|  |  |
| --- | --- |
| **Identificación del problema** | |
| ***Datos de entrada*** | ***Datos de Salida*** |
| 1. Números ingresados | 1. Mensaje indicando si son amistosos o no. |
| **Observaciones *(Limitaciones / Excepciones / Suposiciones)*** | |
|  | |
|  | |
| **Planteamiento de alternativas** | |
| ***Alternativa # 1*** | ***Alternativa #2*** |
| 1. Inicio. 2. Se define num1, num2 como entero. 3. Después del algoritmo principal se crea un SubProceso llamado PedirNumeros, en el cúal se pone de parámetro el valor de num Por referencia, y en su bloque de instrucciones se le solicita al usuario que digite dos números y esos números se leen. 4. Fin de Subproceso. 5. Después del último subproceso, o sea, fuera del algoritmo principal se crea otro subproceso llamado contarNumeros en el cual se pone de parámetro el valor de num1 y num2.. 6. Se declaran las variables suma y cont Como Entero. 7. La variable suma se inicializa en cero. 8. La variable cont se inicializa en uno. 9. En un ciclo mientras cont sea menor al num1 hacer, si el num1 ingresado al dividirlo da como residuo cero, entonces, dicho número por el que se dividió es un divisor, entonces la variable suma va a ir sumando el valor de suma anterior el valor que toma la variable contador y la variable cont va a aumentarse en uno. 10. Si suma=num2 entonces escribir que los números ingresados son amistosos, si no escribir que los números ingresados o son amistosos. 11. Fin de subproceso... 12. En el algoritmo principal se llama a imprimir la función contarNumeros(num1 y num2). 13. Fin de Algoritmo principal. | 1. Inicio. 2. Se define num1, num2 como entero. 3. Después del algoritmo principal se crea un SubProceso llamado PedirNumeros, en el cúal se pone de parámetro el valor de num Por referencia, y en su bloque de instrucciones se le solicita al usuario que digite dos números y esos números se leen. 4. Fin de Subproceso. 5. Después del último subproceso, o sea, fuera del algoritmo principal se crea otro subproceso llamado contarNumeros en el cual se pone de parámetro el valor de num1 y num2. 6. Se declaran las variables suma1, suma2 y cont Como Entero. 7. La variable suma1 y suma2 se inicializan en cero. 8. La variable cont se inicializa en uno. 9. En un ciclo repetir hasta que cont sea igual al num1, si el num1 ingresado al dividirlo da como residuo cero, entonces, dicho número por el que se dividió es un divisor, entonces la variable suma va a ir sumando el valor de suma1 anterior el valor que toma la variable contador y la variable cont va a aumentarse en uno 10. Se repite el proceso anterior pero ahora con suma2. 11. Si suma1=num2 Y suma2=num1 entonces escribir que los números ingresados son amistosos, si no escribir que los números ingresados o son amistosos. 12. Fin de subproceso... 13. En el algoritmo principal se llama a imprimir la función contarNumeros(num1 y num2). 14. Fin de Algoritmo principal |

|  |  |
| --- | --- |
| **Elección de Alternativa** | |
|  | |
| ***Alternativa Seleccionada*** | ***Justificación*** |
| Alternativa #1 | Ambas alternativas me parecen muy funcionales y optimas, sin embargo, la alternativa 1 es mucho mas corta y contiene únicamente una variable suma que hace las operaciones, mientras que la alternativa 2 cuenta con mas variables y es un algoritmo mas largo. |

Ejercicio4: NúmerosAmistosos

Diagrama

Descripción generada automáticamente