PROBLEMA:

**I Entender el Problema**

|  |  |  |
| --- | --- | --- |
| **Incógnita:** | **Datos disponibles:** | **Restricción** |
| * Sumar el resultado del radical del valor en cada posición de la lista con los resultados siguientes. | * Lista numerica. | * Debe ser una lista. |

**II Obtener el Plan**

|  |
| --- |
| Importar la libreria math.  Invocar la función de forma recursiva.  Pasar como parámetro de la función la lista menos el valor de la posición 0.  Sumar el resultado del radical del valor que está en la posición cero con los siguientes resultados.  Parámetros  lista---Se asume que el valor de la lista es [1,2,3] |

**III Aplicar el plan**

|  |
| --- |
| 1. Comprobar si los valores ingresados son una lista. 2. Si el valor ingresado no es una lista válida: 3. El programa termina y regresa un error. 4. Cada vez que se llama la función auxiliar: 5. Comprueba si la lista está vacía. De ser así la función termina y retorna la suma de los radicales. 6. Sumar el resultado del radical del valor que está en la posición cero con los siguientes resultados y llama a la función de forma recursiva. 7. El paso 3 se repite mientras la lista esté llena o contenga almenos un valor. 8. Terminar proceso |

**I Revisar la solución**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **VARIABLES** | | | **Corre Línea** | **Comentario** |
| **Iteración 1** | | |
| lista-parametro | Operación | Suma-return |
| [1,2,3] |  |  | 1 | 6.Comprobar si los valores ingresados son una lista. |
|  |  |  | 2 | 7.Si el valor ingresado no es una lista válida: |
|  |  |  | 3 | 7.1El programa termina y regresa un error. |
|  |  |  | 4 | 8.Cada vez que se llama la función auxiliar: |
| [1,2,3] |  |  | 5 | 8.1Comprueba si la lista está vacía. De ser así la función termina y retorna la suma de los radicales. |
|  |  | 1 | 6 | 8.2Sumar el resultado del radical del valor que está en la posición cero con los siguientes resultados y llama a la función de forma recursiva. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración 2** | | | |
| [2,3] |  |  | 1 | 6.Comprobar si el valor de la variable es mayor o igual a 1. |
|  |  |  | 2 | 7.Si el valor es igual a cero: |
|  |  |  | 3 | 7.1El programa termina y regresa un error. |
|  |  |  | 4 | 8.Cada vez que se llama la función auxiliar: |
| [2,3] |  |  | 5 | 8.1Comprueba si el valor es igual a cero. De ser así la función termina y retorna la suma de los valores. |
|  | + | 2.41 | 6 | 8.2Suma el valor recibido como parámetro con el anterior cada vez que se corre la función y llama a la función de forma recursiva. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración 3** | | | |
| [3] |  |  | 1 | 6.Comprobar si el valor de la variable es mayor o igual a 1. |
|  |  |  | 2 | 7.Si el valor es igual a cero: |
|  |  |  | 3 | 7.1El programa termina y regresa un error. |
|  |  |  | 4 | 8.Cada vez que se llama la función auxiliar: |
| [3] |  |  | 5 | 8.1Comprueba si el valor es igual a cero. De ser así la función termina y retorna la suma de los valores. |
|  | ++ | 4.14 | 6 | 8.2Suma el valor recibido como parámetro con el anterior cada vez que se corre la función y llama a la función de forma recursiva. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteración 4** | | | |
| [] |  |  | 1 | 6.Comprobar si el valor de la variable es mayor o igual a 1. |
|  |  |  | 2 | 7.Si el valor es igual a cero: |
|  |  |  | 3 | 7.1El programa termina y regresa un error. |
|  |  |  | 4 | 8.Cada vez que se llama la función auxiliar: |
| [] |  |  | 5 | 8.1Comprueba si el valor es igual a cero. De ser así la función termina y retorna la suma de los valores. |
|  | ++ | 4.14 | 6 | 8.2Suma el valor recibido como parámetro con el anterior cada vez que se corre la función y llama a la función de forma recursiva. |

Resultado en pantalla

|  |
| --- |
| suma\_raices([1,2,3])  4.14 |