|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Cadenas y funciones** | | | **No.** |  |
| **Asignatura:** | **Métodos Numéricos** | **Carrera:** | **Ingeniería en Sistemas Computacionales** | **Duración de la práctica (Hrs)** |  |

NOMBRE DEL ALUMNO: Isaac MARTINEZ MARTINEZ

GRUPO: 30402

1. **Competencia(s) específica(s):**

**Encuadre con CACEI: Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.**

|  |  |  |  |
| --- | --- | --- | --- |
| **No. atributo** | **Atributos de egreso del PE que impactan en la asignatura** | **Criterios de desempeño** | |
|  |  | **1** |  |
| **2** |  |
| **3** |  |
|  |  | **1** |  |
| **2** |  |
| **3** |  |

**II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):**

**Aula de clases y laboratorio independiente**

**III. Material empleado:**

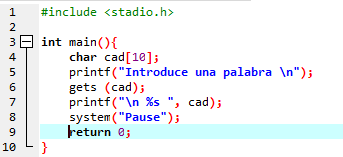
* **Equipo de computo**
* Dec-c++

**IV. Desarrollo de la práctica:**

**Pdf 12:**

**Programa 1:**

**Como declarar una cadena.**



**Línea 4: se declara la cadena como si fuera un arreglo.**

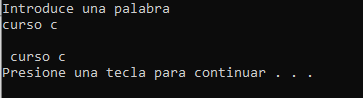
**Línea 5: se imprime el mensaje introduce una palabra.**

**Línea 6: se usa gets para guardar el valor en la cadena adentro de sus paréntesis lleva el nombre de la cadena.**

**Línea 7: se imprime el contenido de la cadena.**

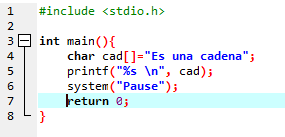
**Y se cierra el método principal.**

**Ejecución:**



**Programa 2:**

**Llenar una cadena.**



**Línea 4: se declara la cadena y se llena colocando un igual y unas comillas para guardar en la cadena el mensaje.**

**Línea 5: se imprime el contenido de la cadena.**

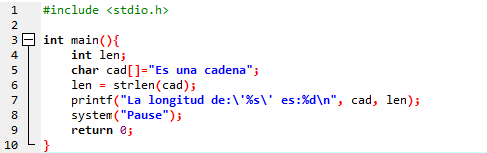
**Y se finaliza el método principal.**

**Ejecución:**



**Programa 3.**

**Uso de la función strlen de una cadena.**



**Linea 4: se declara una variable de tipo entero.**

**Linea 5: se crea la cadena ero ya llenada.**

**Linea 6: se usara la función strlen para poder contar el número de letras que contiene la cadena y se guarda el valor en la variable len.**

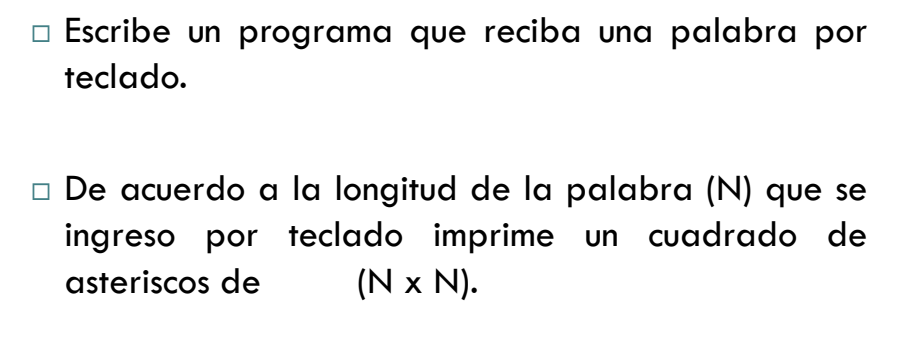
**Linea 7: se imprime la longitud de la cadena y el contenido de la misma.**

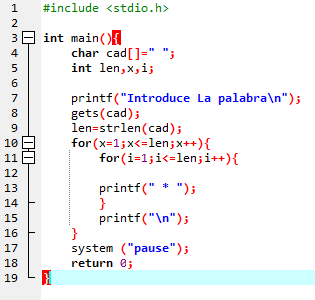
**Se cierra el método principal.**

**Funcionamiento:**



**Programa 4:**





**Linea 4: se declara la cadena pero sin nada almacenado.**

**Linea 5: se declara tres variables de tipo int que son len, x y i.**

**Linea 7: se imprime un mensaje pidiendo una palabra.**

**Linea 8: se usa el gets para guardar lo ingresaso en la cadena.**

**Linea 9: se usa la función strlen para contar los elementos de la palabra ingresada y las guarda en len.**

**Linea 10: se crea un for para llevar el control de columnas que no deben de pasar a len.**

**Linea 11: se crea un for para llevar el control de filas que no deben de pasar a len.**

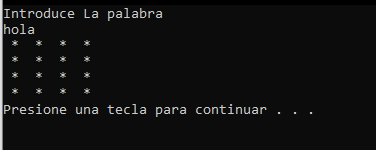
**Linea 13: se imprimen los asteriscos.**

**Linea 14: se cierra el segundo for.**

**Linea 15: se imprime un salto de línea.**

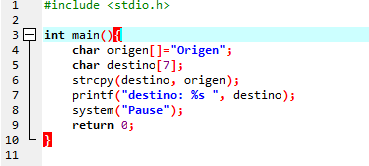
**Linea 16: se cierra el primer for.**

**Ejecución:**



**Programa 5:**

**Ejemplo de uso del strcpy**



**Linea 4: crea una cadena llamada origen que va estar inicializada con origen.**

**Linea 5: se crea una segunda cadena llamada destino de tamaño 7.**

**Linea 6: se usa la función strcpy donde se va intercambia el contenido de variable.**

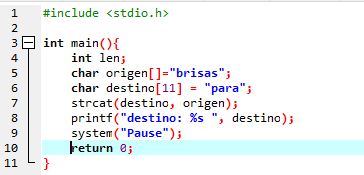
**Linea 7: imprime el contenido del variable destino.**

**Ejecución:**



**Programa 6:**

**Ejemplo de strcat**



**Linea 4: se declara una variable len.**

**Linea 5: se declara una cadena que guarda brisas.**

**Linea 6: se declara una segunda cadena que guarda para:**

**Linea 7: usamos la función strcat.**

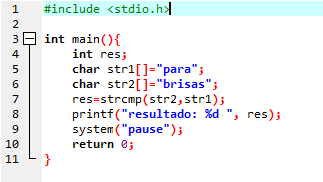
**Linea 8: imprimimos.**

**Ejecución:**



**Programa 7:**

**Uso del strcmp.**



**Linea 5: se crea una cadena guardando para**

**Linea 6: se crea una segunda cadena que guarda brisas**

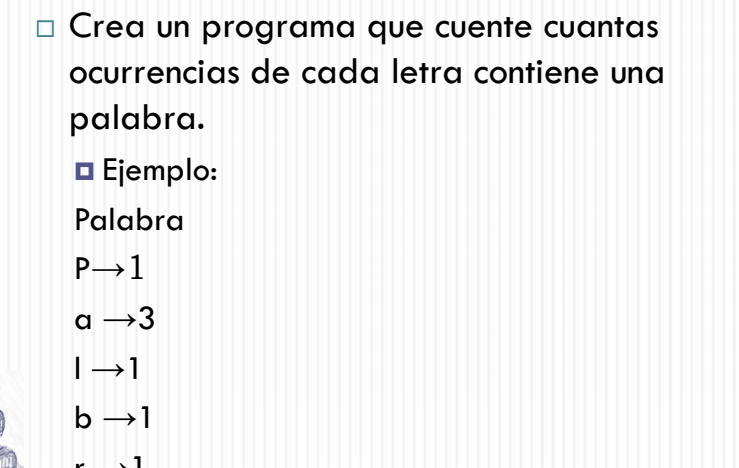
**Linea 6: se usa la función strcmp para comparar la cadena y verificar si son iguales.**

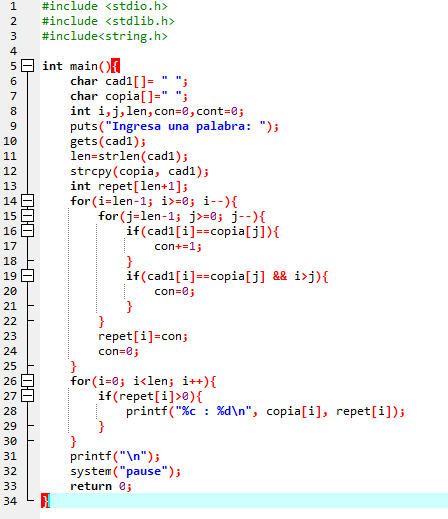
**Linea 7: imprime el resultado.**

**Ejecución:**



**Programa 8:**





**Linea 6y7: se declaran dos cadenas.**

**Linea 8: se declaran variables**

**Linea 9: se imprime un mensaje.**

**Linea 10: se guarda el valor en una cadena.**

**Linea 11: se guarda el valor de toda la palabra.**

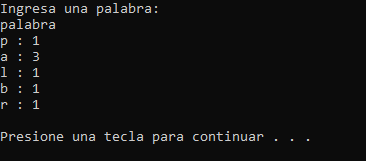
**Linea 12: se copea lo guardado en la primera cadena en la otra cadena.**

**Linea 13: se crea un arreglo.**

**Linea 14-25: se crean varios for para comparar las letras de la palabra ingresada para saber cuánto se repite la misma letra**

**Linea 26-30: se usa otro for para imprimir las letras repetidas:**

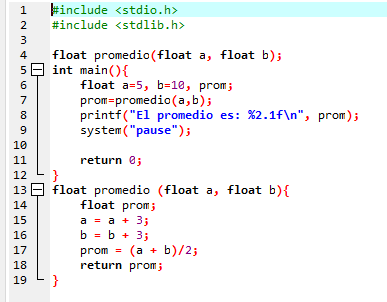
**Ejecución:**



**Pdf 13:**

**Programa 1:**

**Calcular el promedio:**



**Linea 4: se crea la función en la que se va realizar las funciones con sus parámetros.**

**Linea 5: se crea el método principal.**

**Linea 6: se declaran varias variables.**

**Linea 7: Se pasan las variables al método promedio y los resultados que se obtengan se guardan en prom.**

**Linea 8: se imprime el total del promedio.**

**Linea 9: se pausa la consola.**

**Linea 11: se retorna un valor nulo.**

**Linea 12: se cierra el método principal.**

**Linea 13: se manda a llamar la función creada para poder ser usada.**

**Linea 14: se crea una variable.**

**Linea 15: se suma lo que tenía a más 3**

**Linea 16: lo que tenía b más 3**

**Linea 17: se suma a más b y se dividen entre 2 para ser guardado en b.**

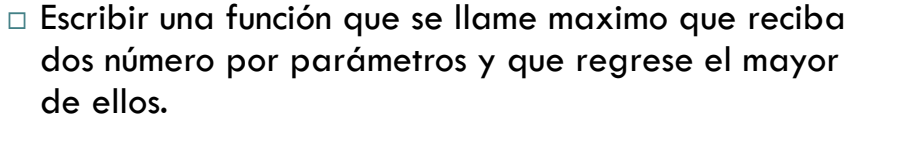
**Linea 18: se retorna el valor de prom al método principal.**

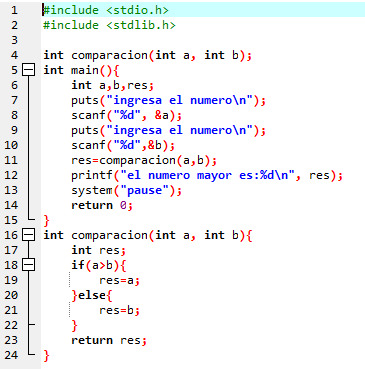
**Linea 19: se cierra la función promedio.**

**Ejecución:**



**Programa 2:**





**Linea 4: se declara la función.**

**Linea 5: se crea el método principal.**

**Linea 6: se crean variables.**

**Linea 7: se imprime un mensaje solicitando un número.**

**Linea 8: se guarda el valor en a.**

**Linea 9: se imprime un mensaje solicitando un número.**

**Linea 10: se guarda el valor en a.**

**Linea 11: se pasan los datos por parámetros y se guardan los resultados en res**

**Linea 12: se imprime res.**

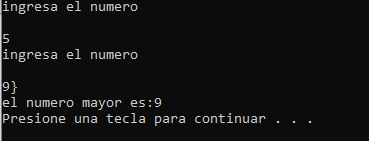
**Linea 16: se manda a llamar la función creada.**

**Linea 17: se declara una variable.**

**Linea 18 se crea un if que compara los números y si a es mayor que b entonces se guarda el valor de a en res y si no se cumple res almacena a b.**

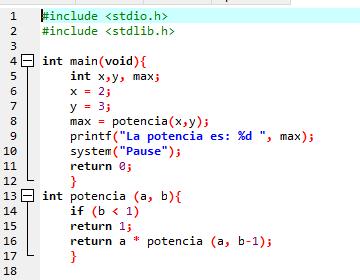
**Linea 23: se retorna el valor de res.**

**Ejecución:**



**Programa 3:**

**Utilizar funciones recursivas.**



**Linea 5-7: se crean variables a x, y se le asigna valores.**

**Linea 8: se pasan valores a la función potencia.**

**Linea 10: se imprime los valores de max.**

**Linea 13: se crea la función.**

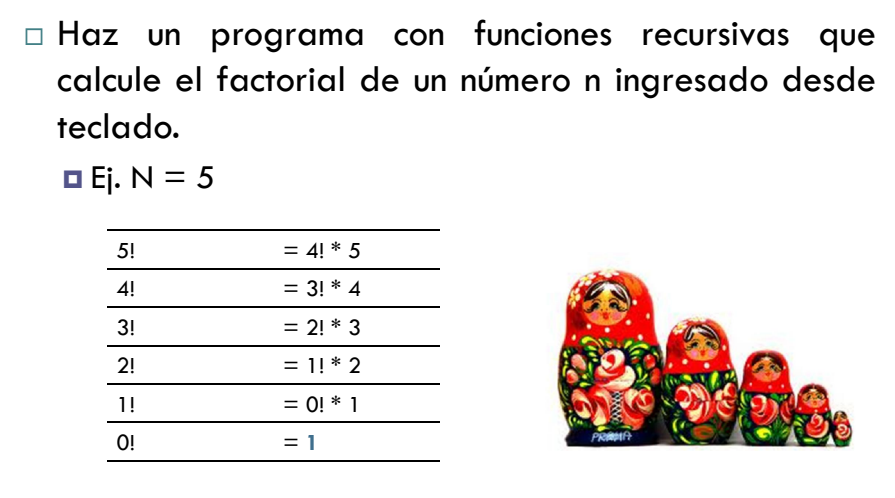
**Linea 14: se usa un if donde compara si b es menor que 1 y si se cumple retorna 1.**

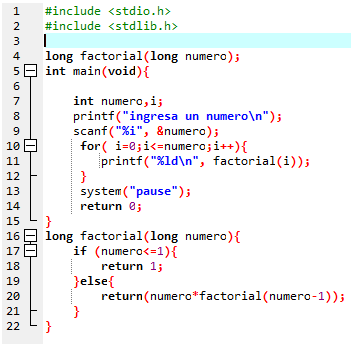
**Linea 16: retorna el valor de la operación.**

**Ejecución:**



**Programa 4:**





**Linea 4: se crea la función.**

**Linea 7: se declara una variable.**

**Linea 8: Se imprime un mensaje**

**Linea 9: se guarda el número en la variable.**

**Linea 10: se crea un for para imprimir las factoriales.**

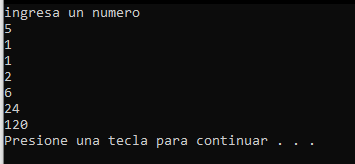
**Linea 16: se manda a llamar la función creada.**

**Linea 17 se crea un if donde compara si número es menor o igual a uno**

**Y si se cumple retorna a 1**

**Linea 20: retorna la operación realizada.**

**Ejecución:**



**V. Conclusiones:**

**El manejo de todas sus funciones de las cadenas es complejas porque te podrían facilitar el trabajo como el copiar o el pasar el valor de una cadena a otra para poder ser modificada como queramos.**

**Y las funciones es algo nuevo que podemos descomponer un programa para que todo su código no esté en un solo método lo cual es más fácil su compresión al estar desfr4agmentado dependiendo loa función que se le quiera dar.**