

Sorteo de puntos

- 1: BENITEZ GUADALUPE CECILIA
- 2: MALASPINA LOPEZ CARLOS MARIANO
- 3: BORQUEZ CIOLFI FEDERICO
- 4: CHAPUR JUAN AGUSTÍN
- 5: MURILLO PEREZ ALEXIS EZEQUIEL
- 6: ABITU VICTOR ANDRES
- 7: BARRIA REYES JOSE DANIEL
- 8: SANTILLAN ANDRES FEDERICO
- 9: MAYOBRE PABLO ARIEL
- 10: SEIDNITZER AGUSTINA MICAELA
- 11: GOMEZ FACUNDO NICOLAS
- 12: DI MARTINO FRANCO GABRIEL
- 13: IRIGOYEN CARLOS DAMIAN
- 14: LUCIA NINA

Trabajo práctico para programar

FECHA DE ENTREGA: Jueves 22/10

Todos los ejercicios deberán utilizar expresiones regulares y deben ser realizados en un lenguaje interpretado.

1. Programe un verificador de formularios (al menos 5 campos). Ejemplo:
 - Username (4-15 caracteres, no permite espacios, ni caracteres especiales)
 - Password (5-15 caracteres, deberá tener al menos un carácter especial, al menos un número y una letra en mayúscula)
2. Programe un extractor y verificador de números de tarjetas de crédito (como mínimo cuatro diferentes tarjetas). Ejemplo:
 - Las tarjetas Visa comienzan con 4, las nuevas tarjetas tienen 16 dígitos y las antiguas 13.
 - Las tarjetas MasterCard comienzan con valores en el rango de 51 a 55 y todas cuentan con 16 dígitos.
3. Programe un filtro de texto el cual bloquee ciertas combinaciones de letras pasadas por parámetro (pueden ser más de una en cuyo caso estarán separadas por coma). Una vez ingresada la combinación de letras a bloquear, se debe programar un funcionamiento al estilo chat o echo para simular una conversación en la cual las oraciones ingresadas serán filtradas reemplazando las ocurrencias de la combinación a bloquear por tantos guiones como letras queden bloqueadas.
 - Ej 1: Combinación bloqueada "ava","te"
Oración ingresada: pava de mate
Respuesta esperada: p--- de ma--
 - Ej 2: Combinación bloqueada "man","s", "tu"
Oración ingresada: semana del estudiante
Respuesta esperada: -e---a del e-tudiante
4. Programar un script que permita convertir un texto con formato A a formato B. Donde el cambio de formato está dado por las siguientes reglas.
 - HolaQueTal >>>>> Hola que tal
 - Hola,QueTal >>>> Hola, que tal
 - Hola;QueTal >>>> Hola; que tal
 - Hola.QueTal >>>> Hola. Que tal
 - EstamosBienLos33,Saludos. >>>> Estamos bien los 33, saludos.

Nota: El formato A corresponde a un texto en notación Camel sin espacios, mientras que el formato B es un texto estándar.

<p>UnOrdenadorHaceMuchoMásQueOrdenar.UsamosEsaPalabraPrestadaDelFrancésOrdinateur,QueEsLaTraducciónDelOriginalInglésComputer.ÉseEraElNombreDeLasPersonasEspecialistasEnHacerCálculosNuméricosYEnElSigloXIXEseTérminoComenzóAAplicarseAMáquinas,QueComenzaronASustituirALosHumanosEnEsasTareas.CalculabanCadaVezMásRápido,SobreTodoCuandoEnElSigloXXSusEngranajesSeSustituyeronPorComponentesElectrónicos.</p> <p>SinEmbargo,LosPrimitivosOrdenadoresTeníanElInconvenienteDeQueSeConstruíanParaResolverUnProblemaDeterminadoYSiSeQueríaUsarlosParaOtroFin,HabíaQueCambiarLosCircuitos.EsoFueAsíHastaQueEn1936UnEstudianteInglés,AlanTuring,PensóEnUnaComputadoraQueResolveríaCualquierProblema;SiempreQueEseProblemaPudieraTraducirseAExpresionesMatemáticasYLuegoReducirseAUnaCadenaDeOperacionesLógicasConNúmeroBinarios,EnLasQueSóloCabíanDosDecisiones:VerdaderoOFalso.LaIdeaEraReducirTodo(Cifras,Letras,Imágenes,Sonidos)ARistrasDeUnosYCerosYUsarUnaReceta,UnPrograma,ParaResolverLosProblemasEnPasosMuySimples.HabíaNacidoElOrdenadorDigital,PeroDeMomentoSóloEraUnaMáquinaImaginaria.</p>	<p>Un ordenador hace mucho más que ordenar. Usamos esa palabra prestada del francés ordinateur, que es la traducción del original inglés computer. Ése era el nombre de las personas especialistas en hacer cálculos numéricos y en el siglo XIX ese término comenzó a aplicarse a máquinas, que comenzaron a sustituir a los humanos en esas tareas. Calculaban cada vez más rápido, sobre todo cuando en el siglo XX sus engranajes se sustituyeron por componentes electrónicos.</p> <p>Sin embargo, los primitivos ordenadores tenían el inconveniente de que se construían para resolver un problema determinado y si se quería usarlos para otro fin, había que cambiar los circuitos. Eso fue así hasta que en 1936 un estudiante inglés, Alan Turing, pensó en una computadora que resolvería cualquier problema; siempre que ese problema pudiera traducirse a expresiones matemáticas y luego reducirse a una cadena de operaciones lógicas con números binarios, en las que sólo cabían dos decisiones: verdadero o falso. La idea era reducir todo (cifras, letras, imágenes, sonidos) a ristras de unos y ceros y usar una receta, un programa, para resolver los problemas en pasos muy simples. Había nacido el ordenador digital, pero de momento sólo era una máquina imaginaria.</p>
---	--

5. Programe un Script que lleve los valores contenidos en un archivo html a la siguiente estructura. Nota: El archivo html ejemplo está en el aula virtual.

- Nro de puntos x y z
- 1 5.7 -1.2 0.8

6. Realice un programa que permita saber si una cadena es aceptada o no por un AFD. Consejo: Utilice la tabla transición para su implementación.

Estados	a	b
s0	s0	s1
s1	s2	s1
s2	s2	s3
s3 -> ACEPTADOR	B	B

7. Programe un anonimizador de documentos que busque y remplace por guiones, apellido y nombre de personas en un texto dado. Nota: consideraremos un nombre y apellido como dos palabras consecutivas que comiencen con mayúscula.
8. Programe un filtro de texto el cual elimine palabras que contienen ciertas combinaciones de letras pasadas por parámetro (pueden ser más de una en cuyo caso estarán separadas por coma). Una vez ingresada la combinación de letras a bloquear, se debe abrir un archivo con el texto para filtrar y se debe guardar el resultado en otro archivo.
- Ej 1: Combinación bloqueada "blo", "el"
Oración ingresada: **el** **establo** está cerrado
Respuesta esperada: está cerrado.
 - Ej 2: Combinación bloqueada "man", "s", "tu"
Oración ingresada: **semana** del **estudiante**
Respuesta esperada: del
9. Programe un software que dado un archivo de texto y un comando retorne palabras que cumplan con una condición dada, por ejemplo:
- Empiecen con 'b', terminen con 'o' y tengan al menos una 'a'.
 - Empiecen 'a' y terminen con 'o'.
 - Empiecen y terminen con 'a'.

El parámetro a recibir será un comando que debe responder al siguiente comportamiento:

- _a-b_ (guión bajo letra "a" guion medio letra "b" guión bajo): se resaltan palabras que empiecen con "a" y terminen con "b".
- _a-b-c_: Las palabras deben empezar con "a", terminar con "c" y contener un b en cualquier parte intermedia.
- _a-_: Las palabras deben comenzar con "a".
- _-b_: Las palabras deben terminar con "b".

Ejemplos:

Contenido archivo:

“Cielo (del latín caelum; de caelum tangi: ser –tocado– herido por el rayo) se define a menudo como el espacio en el que se mueven los astros y por efecto visual parece rodear la Tierra.”

nombre_programa archivo.txt **_c-o_**

respuesta esperada: cielo, como

nombre_programa archivo.txt **_-s_**

respuesta esperada: astros

nombre_programa archivo.txt **_e-_**

respuesta esperada: espacio, en, el, efecto

nombre_programa archivo.txt **_e-e-o_**

respuesta esperada: efecto

10. Programe un aplicativo que dado un archivo de texto y una cadena, elimine del mismo todas las ocurrencias de dicha cadena cuando esté utilizada como nombre de dominio.

Ej: Pensemos en el texto: ‘Tal es el repositorio de wikipedia, el cual puede ubicarse en la siguiente dirección: www.wikipedia.com’ si llamamos a nuestro script con la palabra ‘wikipedia’, el resultado sería el siguiente: “Tal es el repositorio de wikipedia, el cual puede ubicarse en la siguiente dirección:’.

Nota: Tener en cuenta que los demonios pueden presentarse de varias maneras:

<http://www.lalala.com/sdsds>

<https://www.lalala.es/sociales>

www.lalalalall.com.ar/sadasdasdsa

- 11. mismo que el ejercicio 9.
- 12. mismo que el ejercicio 4.
- 13. mismo que el ejercicio 6.
- 14. mismo que el ejercicio 3.