



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* Julio Cesar Saynez Fabian

*Asignatura:* EDA

*Grupo:* 17

*No de Práctica(s):* 1

*Integrante(s):* JONATHAN OMAR CALZADA MARTINEZ

*Semestre:* 2018-2

*Fecha de entrega:* 26 de febrero de 2018

*Observaciones:* Faltó el ejercicio 3 y 4

CALIFICACIÓN: \_\_\_\_1\_\_\_\_

## EJERCICIO 1

Escribe un programa que lea una cadena(máx. 250 caracteres) y cambie la primera letra de cada palabra a mayúscula y el resto en minúscula

Ej : Entrada : Buen día a todos  
Salida : Buen Día A Todos

(2 puntos)

**[CÓDIGO]** #include <stdio.h>

#define s scanf

#def La función main debe  
regresar un valor entero.

char car[250];

int i;

main (void){

p("\nIngrese la cadena \n");

for (i=0;i<250;i++)

{

s("%[^\n]", car);

for (i=0;i<250;i++){

if (car[i]==car[0])

car[i]-=32;

if (car[i]==32)

car[i+1]-=32;

}

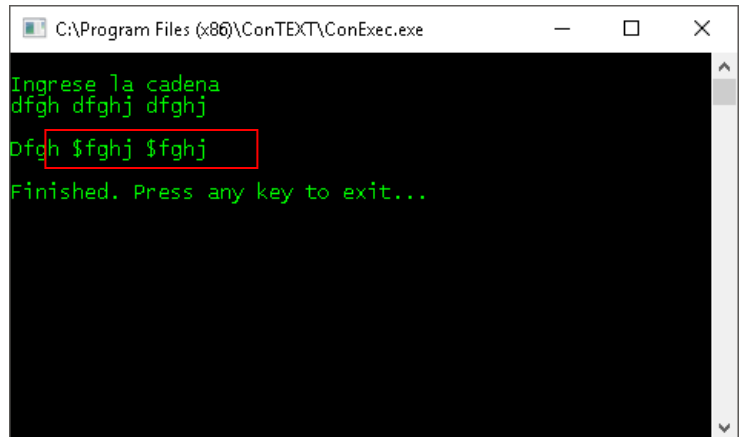
p("\n%s\n",car);

}

}

Rev. 1 punto.

- Falla al cambiar de la segunda palabra en adelante.



## EJERCICIO 2

Lee dos fechas en el formato dd-mm-año (también es valido dd/mm/año), después imprimir ambas fechas comenzando por la más reciente y la salida debe ser en un formato extendido.

(2 puntos)

Escribe dos fechas:

01/04/2017 (enter)

01/01/2018 (enter)

Salida en forma extendida:

1 de enero de 2018

1 de abril de 2017

Rev. 0 puntos

- El programa no entrega el resultado solicitado

#include <stdio.h>

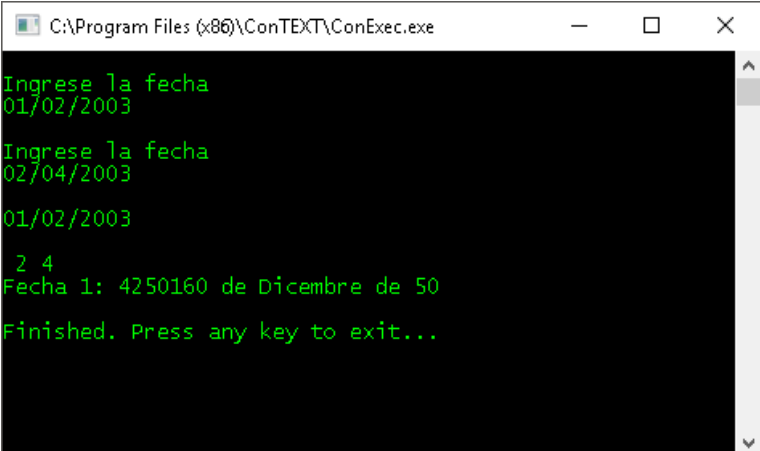
```
#include<stdlib.h>
#define s scanf
#define p printf
```

```
char vec1[11],vec2[11];
int i,sum,sum2,cont=1;
main (void){
p("\nIngrese la fecha \n");
fflush(stdin);
s("%^[^\\n]",vec1);

p("\nIngrese la fecha \n");
fflush(stdin);
s("%^[^\\n]",vec2);
```

```
sum=vec1[3]+vec1[4];
sum2=vec2[3]+vec2[4];
sum-=48;
sum2-=48;
```

```
if(sum<13){
switch (sum){
case '1':
p("Fecha 1: %i de Enero de %i\\n",vec1,sum);
break ;
case '2':
p("Fecha 1: %i de Feb de %i\\n",vec1,sum);
break;
case '3':
p("Fecha 1: %i de Marzo de %i\\n",vec1,sum);
break;
case '4':
p("Fecha 1: %i de Abril de %i\\n",vec1,sum);
break;
case '5':
p("Fecha 1: %i de Mayo de %i\\n",vec1,sum);
break;
case '6':
p("Fecha 1: %i de Junio de %i\\n",vec1,sum);
break;
case '7':
p("Fecha 1: %i de Julio de %i\\n",vec1,sum);
break;
case '8':
p("Fecha 1: %i de Agosto de %i\\n",vec1,sum);
break;
case '9':
p("Fecha 1: %i de Septiembre de %i\\n",vec1,sum);
break;
case '10':
p("Fecha 1: %i de Octubre de %i\\n",vec1,sum);
break;
case '11':
p("Fecha 1: %i de Noviembre de %i\\n",vec1,sum);
break;
case 12:
p("Fecha 1: %i de Diciembre de %i\\n",vec1,sum);
break;
}}
if(sum2<13){
switch (sum2){
case '1':
p("Fecha 1: %i de Enero de %i\\n",vec2,sum);
```



```
C:\Program Files (x86)\ConTEXT\ConExec.exe

Ingrese la fecha
01/02/2003

Ingrese la fecha
02/04/2003

01/02/2003

2 4
Fecha 1: 4250160 de Diciembre de 50

Finished. Press any key to exit...
```

```

break ;
case '2':
    p("Fecha 1: %i de Feb de %i\n",vec2,sum);
break;
case '3':
    p("Fecha 1: %i de Marzo de %i\n",vec2,sum);
break;
case '4':
    p("Fecha 1: %i de Abril de %i\n",vec2,sum);
break;
case '5':
    p("Fecha 1: %i de Mayo de %i\n",vec2,sum);
break;
case '6':
    p("Fecha 1: %i de Junio de %i\n",vec2,sum);
break;
case '7':
    p("Fecha 1: %i de Julio de %i\n",vec2,sum);
break;
case '8':
    p("Fecha 1: %i de Agosto de %i\n",vec2,sum);
break;
case '9':
    p("Fecha 1: %i de Septiembre de %i\n",vec2,sum);
break;
case '10':
    p("Fecha 1: %i de Octubre de %i\n",vec2,sum);
break;
case '11':
    p("Fecha 1: %i de Noviembre de %i\n",vec2,sum);
break;
case 12:
    p("Fecha 1: %i de Diciembre de %i\n",vec2,sum);
break;

}}
p("\n%s\n",vec1);
p("\n %c %c\n",sum,sum2);
p("Fecha 1: %i de Diciembre de %i\n",vec2,sum);

}

```

### EJERCICIO 3

Implementar el algoritmo que permita leer una matriz de dos dimensiones, como una cadena utilizando los símbolos [ ], para indicar el inicio y terminación de la matriz y el de cada línea. Considere valores enteros con signo.

(3 puntos)

Ejemplo

Estribe la matriz: [ [ 1 , 2] , [ 3, 4] ]

[CÓDIGO]

**EJERCICIO 4**

Escribir un programa que lea las dimensiones de una matriz ( $n \times m$ ) y la llene la matriz con números enteros desde 0, con un incremento de 3, después calcular la transpuesta de la matriz.

Sea  $A$  una matriz con  $m$  filas y  $n$  columnas. La matriz transpuesta, denotada con  $A^t$

Está dada por:  $(A^t)_{ij} = A_{ji}, 1 \leq i \leq n, 1 \leq j \leq m$

En donde el elemento  $a_{ji}$  de la matriz original  $A$  se convertirá en el elemento  $a_{ij}$  de la matriz transpuesta  $A^t$

Nota : Validar para matrices rectangulares.

(3 puntos)

[CÓDIGO]

**Conclusiones:**

~~Bueno, pues se me complico~~ la practica y aquí me doy cuenta que necesito practicar y estudiar bastante, solo pude hacer correctamente el primero, el segundo lo intenté con la instrucción switch y no me salió, y hasta el final pensé que era mejor con arreglos pero ya no me daba tiempo. Bueno no tengo nada más que decir más que ponerme a estudiar.

Estos son comentarios,  
favor de revisar como se  
redactan las conclusiones.