Universidad Autónoma de Guadalajara

**Programación Avanzada 2**

**LAB01: Funciones en C**

* *void GENFUN\_vCapsOn (uint8 \*pu8Src, uint8 u8SizeOfList)*

Inicio

au8array[]="helloworld"

Imprimir, “Función 1 \*CapsOn\*

Imprimir “Original array”: **Imprime el au8array**

Imprimir “After CapsOn”

While (u8SizeOFList !=0)

If(\*pu8Src>ASCII\_LOW\_THRESHOLD\_OFF\_ON && \*pu8Src<ASCII\_HIGH\_THRESHOLD\_OFF\_ON)

*False True*

\*pu8Src -= ASCII\_CONVERTION\_FACTOR;

//Nothing to do//

pu8Src++;

u8SizeOFList--;

printf("%s\n au8array);

END

* *void GENFUN\_vCapsOff (uint8 \*pu8Src, uint8 u8SizeOfList)*

Inicio

au8array[]="helloworld"

Imprimir, “Función 1 \*CapsOff\*

Imprimir “Original array”: **Imprime el au8array**

Imprimir “After CapsOff”

While (u8SizeOFList !=0)

If(\*pu8Src>ASCII\_LOW\_THRESHOLD\_OFF\_OFF && \*pu8Src<ASCII\_HIGH\_THRESHOLD\_OFF\_OFF)

*True*

*False*

\*pu8Src += ASCII\_CONVERTION\_FACTOR;

//Nothing to do//

pu8Src++;

u8SizeOFList--;

printf("%s\n au8array);

END

* *uint8 GENFUN\_u8GetOccurence (uint8 \*p u8Src, uint8 u8Target, uint8 u8SizeOfList)*

Inicio

au8array="helloworld"

au8array=\*pu8Src

printf("función 3 \*GetOccurence\*\n");

printf("Cadena a buscar : %s\n",au8array);

printf("Letra a buscar : %c \n",u8Target);

While (u8SizeOFList !=0)

If(\*pu8Src==u8Target)

*False True*

u8Elements++;

//Nothing to do//

pu8Src++;

u8SizeOFList--;

printf("La letra se encontró %d veces\n\n", u8VALOR);

END

* *uint8 GENFUN\_u8MemSet (uint8 \*pu8Src, uint8 u8SizeOfList)*

Inicio

au8array="helloworld"

printf("función 4 \*MemSet\*\n");

printf("Cadena a utilizar: %s\n",au8array);

printf("Ingrese el carácter a utilizar: ");

scanf("%c",&u8Char2Set);

While (u8SizeOFList !=0)

\*pu8Src=u8Char2Set;

pu8Src++;

u8SizeOFList--;

printf("\n La nueva cadena es %s",au8array);u8VALOR);

END

• void GENFUN\_u8GetAverage (uint8 \*pu8Src, uint8 u8Char2Set, uint8 u8SizeOfList)

Inicio

uint32 Sumadevalores=0;

uint8 div=u8SizeOfList;

printf("La funcion Average dara el promedio obtenido de la cadena siguiente:");

for(i=0;i<7;i++)

printf(" %d ",au8arraynum[i]);

printf("\n");

uint32 Sumadevalores=0;

uint8 div=u8SizeOfList;

While (u8SizeOFList !=0)

Sumadevalores+= \*pu8Src;

pu8Src++;

u8SizeOfList--;

Sumadevalores/=div;

printf("\nEl promedio obtenido es : %d",saveoaverage);

END

• void GENFUN\_u8MemCopy (uint8 \*pu8Src, uint8 \*pu8Dest, uint8 u8SizeOfList)

Inicio

printf("La función MemCopy copiara los valores de la cadena siguiente y estos serán asignados a una cadena diferente:");

for(i=0;i<7;i++)

printf(" %d ",au8arraynum[i])

printf("\nLa cadena vacia es la siguinete: ");

for(i=0;i<7;i++)

printf(" %d ",au8arraynum2[i]);

printf("\n");

while ( u8SizeOfList != 0 )

\*pu8Dest= \*pu8Src;

pu8Src++;

pu8Dest++;

u8SizeOfList--;

printf("\n Ya se realizo la función MemCopy y el resultado es el siguiente : ");

for(i=0;i<7;i++)

printf(" %d ",au8arraynum2[i]);

END

• void GENFUN\_vSortList (uint8 \*pu8Src, uint8 \*pu8Dest, uint8 u8SizeOfList)

Inicio

for(i2=0;i2<10;i2++)

printf(" %d ", au8Aux[i2]);

uint8 u8SizeOfList2 = u8SizeOfList;

uint8 Aux[10], TempAux;

while(indice!= 0)

pu8Dest = pu8Src;

pu8Dest ++;

while ( u8SizeOfList2 != 0 )

if ( \*pu8Src < \*Aux )

Else if

TempAux = \*pu8Src;

\*pu8Src = \*pu8Dest;

\*pu8Dest = TempAux;

pu8Dest ++;

u8SizeOfList2++;

u8SizeOfList--;

pu8Src ++;

printf("Afteer Sort %d\n", au8Disorder[i2]);

END

• void GENFUN\_vSoftSignal (uint8 \*pu8Src, uint8 \*pu8Dest)

Inicio

printf("\n\n Funcion SoftSignal \n");

uint8 indice = END\_SOFT;

uint8 indice2 = END\_SOFT\_2;

uint8 prom=0,num=0,sum=0;

pu32Src=pu32Src-255;

\*pu32Src = rand() % 10;

while(indice!= 0)

printf("Se mostrara los valores almacenados aleatoriamente en el primer areglo: \n");

printf(" %d ",\*pu32Src);

printf("\n El resultado de la segunda cadena con los promedios es : \n");

indice--;

pu32Src++;

while(indice!= 0)

num = \*pu32Src;

pu32Src++;

sum=(num+\*pu32Src);

prom=(sum/2);

\*pu32Dest=prom;

END

printf(" %d ",\*pu32Dest );

indice2--;

pu32Dest++;

•void GENFUN\_vFilterSignal (uint8 \*pu8Src, uint8 \*pu8Dest, uint8 u8MaxVal, uint8 u8MinVal)

Inicio

printf("\n\Función FilterSignal \n");

uint8 indice = END\_SOFT;

uint8 indice2 = END\_SOFT;

uint8 o=0;

u32MaxVal--;

printf(" %d ",\*pu32Src);

pu32Src=pu32Src-255;

pu32Dest=pu32Dest-1;

indice--;

pu32Src++;

pu32Dest++;

\*pu32Src = (u32MinVal) + (rand() % u32MaxVal);

\*pu32Dest=o;

while(indice!= 0)

printf("Se mostrara los valores almacenados aleatoriamente en el primer areglo con el limite menor de %d y el limite mayor de %d\n",u32MinVal,u32MaxVal);

printf(" %d ",\*pu32Src);

while(indice!= 0)

END

printf(" %d ",\*pu32Src);

indice2--;

o = \*pu32Src;

\*pu32Dest=o;

pu32Src++;

pu32Dest++;