



UNIVERSIDAD AUTONOMA DE GUADALAJARA

Practica #02 Microcomputadoras “Funciones en C”

Implemente las siguientes funciones en C:

Memory Management:

Funciones de soporte de manejo dinámico de memoria (Heap)

- *void* vfnMalloc(unsigned short wSize)* – regresa un apuntador a la posición en memoria reservada para el bloque de tamaño *wSize* solicitado. Si no hay memoria, se regresa un apuntador null.
- *void vfnFree(void* vpPtr)* – libera el espacio de memoria apuntado por *vpPtr*
- *void* realloc(void* vpPtr, unsigned short wSize)* – Se cambia de posición el buffer apuntado por *vpPtr* para modificar el tamaño al especificado por *wSize*. Si no hay espacio para el nuevo buffer, se regresa el apuntador inicial.
- *void* calloc(unsigned short wSize)* – regresa un apuntador de a un buffer lleno de 0 de tamaño *wSize*. Si no hay espacio para crear el buffer, regresa un apuntador null.

Queue Management:

Funciones de manejo de estructuras de datos tipo Queue

- *void vfnQueueInit(void)* – Inicializa los apuntadores, contadores, etc de manejo del Queue a sus valores iniciales.
- *unsigned char bfnEnqueue(char* cpData, unsigned short wSize)* – mete el total de *wSize* datos apuntados por *cpData* en el Queue y regresa la cantidad de datos que se pudieron ingresar. Si el queue está lleno, entonces se regresa 0. Si solo se puede meter una parte, entonces se regresa la cantidad que si entro.
- *unsigned char bfnDequeue(char* cpData, unsigned short wSize)* – extrae a *cpData* la cantidad de datos indicadas por *wSize* del Queue. Se regresa la cantidad de datos extraídos. Si el queue está vacío, se regresa 0.

Stack Management:

Funciones de manejo de estructuras de datos tipo Stack

- *void vfnStackInit(void)* – inicializa los apuntadores, contadores, etc de manejo del Stack a sus valores iniciales
- *unsigned char bfnStackPush(char* cpData, unsigned short wSize)* – mete el total de *wSize* datos apuntados por *cpData* en el Stack y regresa la cantidad de datos que se pudieron ingresar. Si el Stack está lleno, entonces regresa 0. Si solo se puede meter una parte, entonces regresa la cantidad de datos que si entro.
- *unsigned char bfnStackPop(char* cpData, unsigned short wSize)* – extrae a *cpData* la cantidad de datos indicada por *wSize* del Stack. Se regresa la cantidad de datos extraídos. Si el Stack está vacío, se regresa 0.

General Functions:

Funciones genericas

- *unsigned char* bfnCmdLine(*char** cpCmd, *char** cpCmdList) – Recibe en cpCmdList un apuntador a la lista de comandos válidos y en cpCmd el comando a verificar. Si el comando a verificar se encuentra en la lista de comandos válidos, se regresa el índice correspondiente a la posición de la lista en la que se encontró el comando
- *unsigned char* bfnLog2 (*unsigned long* dwNum) – regresa la cantidad mínima de bits necesarios para representar el numero recibido en dwNum.

Data Conversion

Funciones de conversión de datos

- *unsigned long* dwAToUL (*char** cpPtr) – regresa el numero de 32 bits sin signo que resulta de convertir el numero representado en la cadena terminada en 0 apuntada por cpPtr Si se encuentra un carácter invalido, entonces se regresa 0
- *unsigned short* wAToW (*char** cpPtr) – regresa el numero de 16 bits sin signo que resulta de convertir el numero representado en la cadena terminada en 0 apuntada por cpPtr Si se encuentra un carácter invalido, entonces se regresa 0
- *unsigned char* bAtoB(*char** cpPtr) – regresa el numero de 8 bits sin signo que resulta de convertir el numero representado en la cadena terminada en 0 apuntada por cpPtr Si se encuentra un carácter invalido, entonces se regresa 0
- *unsigned char* bfnBToA(*unsigned char* bData, *char** cpPtr) – regresa el total de caracteres agregados a la cadena apuntada por cpPtr resultantes de convertir bData a su representación en dígitos ASCII.
- *unsigned char* bfnWToA(*unsigned short* wData, *char** cpPtr) – regresa el total de caracteres agregados a la cadena apuntada por cpPtr resultantes de convertir wData a su representación en dígitos ASCII.
- *unsigned char* bfnDwToA(*unsigned long* dwData, *char** cpPtr) – regresa el total de caracteres agregados a la cadena apuntada por cpPtr resultantes de convertir dwData a su representación en dígitos ASCII.

Data Management

- *unsigned char* bfnByteFlip(*unsigned char* bData) – regresa el numero de 8 bits resultante de reflejar cada bit de bData
- *unsigned short* wfnWordFlip(*unsigned short* wData) – regresa el numero de 16 bits resultante de reflejar cada bit de wData
- *unsigned long* dwfnDWordFlip(*unsigned long* dwData) – regresa el numero de 32 bits resultante de reflejar cada bit de dwData
- *unsigned char* bfnNibbleSwap(*unsigned char* bData) – regresa el numero de 8 bits resultante de intercambiar los nibbles de bData
- *unsigned short* wfnByteSwap(*unsigned short* wData) – regresa el numero de 16 bits resultante de intercambiar los bytes de wData
- *unsigned long* dwfnWordSwap(*unsigned long* dwData) – regresa el numero de 32 bits resultante de intercambiar los words de dwData

Como entregables se esperan:

- 1) Demostración de los códigos funcionando
- 2) Reporte de la practica

Una vez que termine su práctica, suba en un “.zip” al Moodle los archivos y el reporte con sus conclusiones en ingles y español. (Como se especifica en el documento de Especificación del Formato de Practica)