

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

INF239 SISTEMAS OPERATIVOS

Semestre 2024-1

Laboratorio 6

1) (5 puntos – nombre del archivo a entregar: *xrunNFit.c*) Copie el archivo *xrun.c* con el nombre *xrunNFit.c* y modifíquelo, de forma que verifique que se está empleando el algoritmo *next fit*. Considere emplear rutinas que impriman la lista de agujeros libres junto con los tamaños de cada agujero.

Para verificar que la asignación de memoria es *next fit*, usted previamente tiene que solicitar memoria a *xmalloc* y luego liberar algunos bloques apropiadamente invocando a la rutina *xfree*. El resultado debe llevar a conclusión inequívoca (sin lugar a dudas), que el algoritmo empleado es *next fit*.

Usted es libre de hacer cualquier otra modificación que logre su objetivo.

2) (10 puntos – nombres de los archivos a entregar: *xallocBFit.c* – *xrunBFit.c*) Copie el archivo *xalloc.c* con el nombre *xallocBFit.c* y modifíquelo, de forma que emplee el algoritmo *best fit*, para la asignación de memoria. También copie el archivo *xrun.c* con el nombre *xrunBFit.c* y modifíquelo, de forma que verifique que se está empleando el algoritmo *best fit*. Considere emplear rutinas que impriman la lista de agujeros libres junto con los tamaños de cada agujero.

Para verificar que la asignación de memoria es *best fit*, usted previamente tiene que solicitar memoria a *xmalloc* y luego liberar algunos bloques apropiadamente invocando a la rutina *xfree*. El resultado debe llevar a conclusión inequívoca (sin lugar a dudas), que el algoritmo empleado es *best fit*.

Usted es libre de hacer cualquier otra modificación que logre su objetivo.

3) (5 puntos – nombre del archivo a entregar: *xallocV2.c* – *xrunV2.c*) Implemente la función *xrealloc*

```
void *xrealloc(void * ptr, size_t size)
{
    ...
}
```

The *xrealloc()* function changes the size of the memory block pointed to by *ptr* to *size* bytes. The contents of the memory will be unchanged in the range from the start of the region up to the minimum of the old and new sizes. If the new size is larger than the old size, the added memory will not be initialized.

If *ptr* is NULL, then the call is equivalent to *malloc(size)*, for all values of *size*.

If *size* is equal to zero, and *ptr* is not NULL, then the call is equivalent to *free(ptr)*.

Unless *ptr* is NULL, it must have been returned by an earlier call to *malloc* or related functions. If the area pointed to was moved, a *free(ptr)* is done.

Modifique el programa *xrun.c* para comprobar que la implementación de *xrealloc* se ejecuta como se ha definido.

Porf. Alejandro T. Bello Ruiz