Assignment4 Report

20181016 KwonYongmin

Assignment4.c

To make heap memory allocator, I manage two variables.

One is “heap\_area” which is char array with 64 bytes, allocated in heap memory. And I make the integer variable “filled” represent the index of filled “heap\_area” continuously.

The other one is “mem\_table” which is an array consisted with 64 “mem\_table\_entry”s. “mem\_table\_entry” is a structure defined in “assignment4.h”. It contains order of data, each data’s name, and size. Similar with “heap\_area”, I also manage the integer variable “current\_table\_index” which means the number of data in “mem\_table\_entry”.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Figure 1 main function

Assignment4.h

There are two structures and one function.

**mem\_table\_entry**

As I mentioned above, it represents the name and size of the data which is allocated to “heap\_area”.

텍스트, 폰트, 스크린샷, 친필이(가) 표시된 사진

자동 생성된 설명

Figure 2 structure of “mem\_table\_entry”

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

Figure 3 usage of “mem\_table\_entry”

**struct\_table\_entry**

This structure is for supporting structure to our heap allocator. When allocating structure in “heap\_area”, I make an array of “struct\_table\_entry”. And using data in this array, it allocates the data in the structure to “heap area”.

텍스트, 폰트, 스크린샷, 그래픽이(가) 표시된 사진

자동 생성된 설명

Figure 4 structure of “struct\_table\_entry”

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

Figure 5 usage of “struct\_table\_entry”

**manage\_heap\_area**

This function is quite long, but it works easy. It conducts two processes, allocation and deallocation to “heap\_area”.

In data allocation process, it just takes the information of data from stdin, and if the information is correct, it allocates the data to “heap\_area” and saves the information of the data to “mem\_table”.

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Figure 6 allocation process

Deallocation process is simple. Using “mem\_table” it finds the offset and index of target variable in “heap\_area”. And then it clears target data from both of “heap\_area” and “mem\_table”.

텍스트, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

Figure 7 deallocation process