

System Programming Project 3

담당 교수 : 김영재

이름 : 정지석

학번 : 20181687

1. Depiction of an architecture

2. Description of subroutines, structs, any global variables

```
#define WSIZE 4
```

Word의 size이다.

```
#define DSIZE 8
```

Double word의 size이다.

```
#define CHUNKSIZE 16
```

Initial heap의 size이다.

```
#define OVERHEAD 24
```

minimum block size이다.

```
#define PACK(size, alloc) ((size) | (alloc))
```

Size와 allocated 된 byte를 한 word에 집어넣는다.

```
define GET_SIZE(p) ((* (size_t *) (p)) & ~0x7)
```

header or footer로부터 size를 받아온다.

```
#define GET_ALLOC(p) ((* (size_t *) (p)) & 0x1)
```

header or footer로부터 allocated 된 bit를 받아온다.

```
#define HDRP(ptr) ((void *) (ptr) - WSIZE)
```

Block의 header로부터 주소를 받아온다.

```
#define FTRP(ptr) ((void *) (ptr) + GET_SIZE(HDRP(ptr)) - DSIZE)
```

Block의 footer로부터 주소를 받아온다.

```
#define NEXT_BLKP(ptr) ((void *) (ptr) + GET_SIZE(HDRP(ptr)))
```

Next block 으로부터 주소를 받아온다.

```
#define PREV_BLKP(ptr) ((void *) (ptr) - GET_SIZE(HDRP(ptr)) - WSIZE)
```

previous block 으로부터 주소를 받아온다.

```
#define NEXT_FREEP(ptr) (*(void **) (ptr + DSIZE))
```

Next free block 으로부터 주소를 받아온다.

```
#define PREV_FREEP(ptr) (*(void **) (ptr))
```

Previous free block 으로부터 주소를 받아온다.

```
static char *heap_listp;
```

첫번째 block 의 pointer

```
static char *free_listp;
```

첫번째 free block 의 pointer

```
static void *extend_heap(size_t words);
```

heap size를 늘려주는 역할을 하는 함수이다.

```
static void place(void *ptr, size_t size);
```

free block의 시작 부분에 block을 place 하는 함수이다.

```
static void *coalesce(void *ptr);
```

새롭게 free 된 block을 이전의 free block들과 연결해주는 함수이다.

```
static void remove_block(void *ptr);
```

free list에서 block을 제거해 준다.