**Pintos Project 0-2: Pintos Data Structure**

| 담당 교수 : | 문의현 |
| --- | --- |
| 학번 : | 20161250 |
| 이름 : | 이지호 |
|  |  |

**반드시 아래의 양식과 순서를 따라서 작성하기 바랍니다.**

1. **Additional Implementation**

| **Prototype** | struct list\_item \*get\_list\_item(const struct list\_elem \*e) |
| --- | --- |
| **Parameter** | 반환하고자 하는 list\_item에 들어있는 list\_elem 포인터  const struct list\_elem \*e |
| **Return** | 파라미터로 입력된 list\_elem을 포함하는 list\_item 포인터 |
| **Function** | struct list\_elem \*e를 가지고 있는 struct list\_item \*it를 return  내부적으로 list\_entry를 호출함 |

| **Prototype** | struct hash\_item \*get\_hash\_item(const struct hash\_elem \*e) |
| --- | --- |
| **Parameter** | 반환하고자 하는 hash\_item에 들어있는 hash\_elem 포인터  const struct hash\_elem \*e |
| **Return** | 파라미터로 입력된 hash\_elem을 포함하는 hash\_item 포인터 |
| **Function** | hash\_elem 포인터를 포함하는 hash\_item 포인터를 return  내부적으로 hash\_entry를 호출함 |

1. **List**

| **Prototype** | void list\_swap(struct list\_elem \*a, struct list\_elem \*b) |
| --- | --- |
| **Parameter** | (순서를 바꾸고자 하는) list 내의 두 list\_elem 포인터  struct list\_elem \*a, \*b |
| **Return** | X (void) |
| **Function** | 두 list\_elem의 위치를 바꾼다(swap).  아무 값도 return하지 않는다. |

| **Prototype** | void list\_shuffle(struct list \*list) |
| --- | --- |
| **Parameter** | Shuffle할 list 포인터  struct list \*list |
| **Return** | X (void) |
| **Function** | list 내의 모든 element들을 랜덤하게 swap  내부적으로 list\_swap()을 호출함 |

1. **Hash Table**

| **Prototype** | unsigned hash\_int\_2(int num) |
| --- | --- |
| **Parameter** | Hash할 int형 변수  int num |
| **Return** | Hash된 결과를 unsigned로 반환 |
| **Function** | 입력받은 int num의 bit를 모두 flip한 결과를 unsigned로 반환  모든 int형 변수 num에 대해, hash 결과인 ~num과 1:1 대응 관계이므로 bucket size가 무한하다면 hash conflict가 발생하지 않는다. |

1. **Bitmap**

| **Prototype** | struct bitmap \*bitmap\_expand(struct bitmap \*b, size\_t size) |
| --- | --- |
| **Parameter** | Expand할 bitmap 포인터, 추가할 bit size  struct bitmap \*b, size\_t size |
| **Return** | Expand된 bitmap 포인터 |
| **Function** | 기존 bitmap의 뒷부분에 size개의 0 bit를 expand(pad)  성공 시 bitmap pointer를, 실패 시 NULL을 반환 |