Data Structure LAB #4-1 API

struct dllist_item   (typedef dllitem)

Double Linked List structure

[prev_link] – previous link of item

[data] – string data (size = DATA_SIZE)

[next_link] – next link of item

dllitem* create_dllist(char *data)

Create Double Linked List.

Make head item with data.

<input>

[data] – head data

<return>

Double Linked List head item pointer

void insert_item (dllitem **head, char *data, dllitem *prev)

Insert data in Double Linked List.

If head is NULL, create Double Linked List with data.

<input>

[head] – Double Linked List head pointer what you want to input item

[data] – input string data

[prev] – Double Linked List item pointer. Previous position of target.

<return>

None

void insert_item_index(dllitem **head, char *data, int index)

Insert data in Double Linked List with index position.

If input index is larger than list count, insert data after last item.

<input>

[head] – Double Linked List head pointer what you want to input item. Do not input NULL value.

[data] – input string data

[index] – Target index. Head index is 0.

<return>

None

```
void insert_item_last(dllitem **head, char *data)
```

Insert data after last item in Double Linked List.

If head is NULL, create Double Linked List with data.

<input>

[head] – Double Linked List head pointer what you want to input item

[data] – input string data

[index] – Target index. Head index is 0.

<return>

None

void remove_item(dllitem **head, dllitem *target)

Remove item in Double Linked List.

If head is NULL, no operation.

<input>

[head] – Double Linked List head pointer what you want to remove item

[target] – Double Linked List item pointer to remove.

<return>

None

void remove_item_index(dllitem **head, int index)

Remove item in Double Linked List with index

If input index is larger than list count, remove last item.

<input>

[head] – Double Linked List head pointer what you want to remove item

[index] – Target index. Head index is 0.

<return>

None

void remove_head(dllitem **head)

Remove head & Change head with next item in Double Linked List.

<input>

[head] – Double Linked List head pointer what you want to remove head. Do not input NULL value.

<return>

None

void remove_item_last(dllitem **head)

Remove last item in Double Linked List.

If head is NULL, no operation.

<input>

[head] – Double Linked List head pointer what you want to remove last item.

<return>

None

void remove_dllist(dllitem *head)

Remove & Deallocate Double Linked List.

If head is NULL, no operation.

<input>

[head] – Double Linked List what you want to remove.

<return>

None

void print_list(dllitem *head)

Print Double Linked List items.

<input>

[head] – Double Linked List what you want to print.

<return>

None