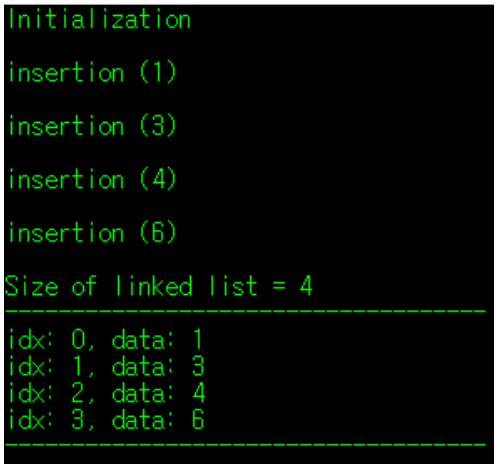
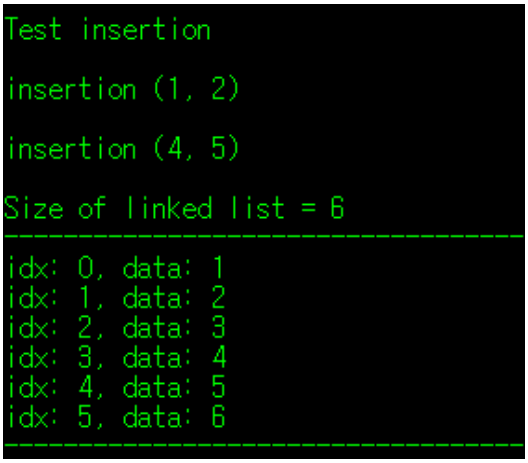
Question 1

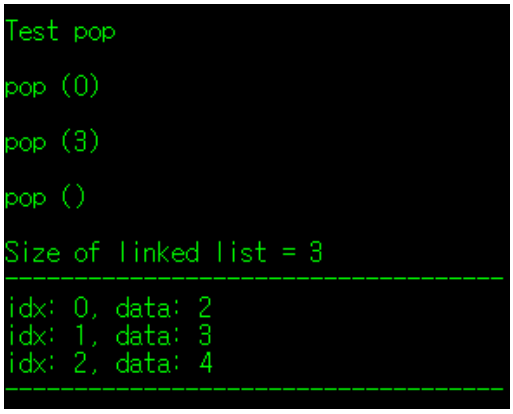
1. Explanation

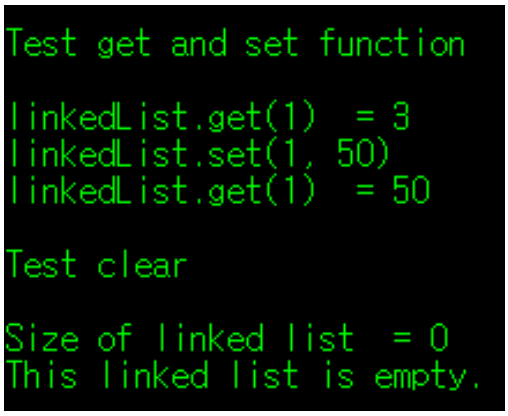
This program can store data in linked list type. Linked list has nodes which is composed data and next data's location, so all data is linked one head node. User can insert, pop, clear, set and get data.

2. Result screen









3. Study

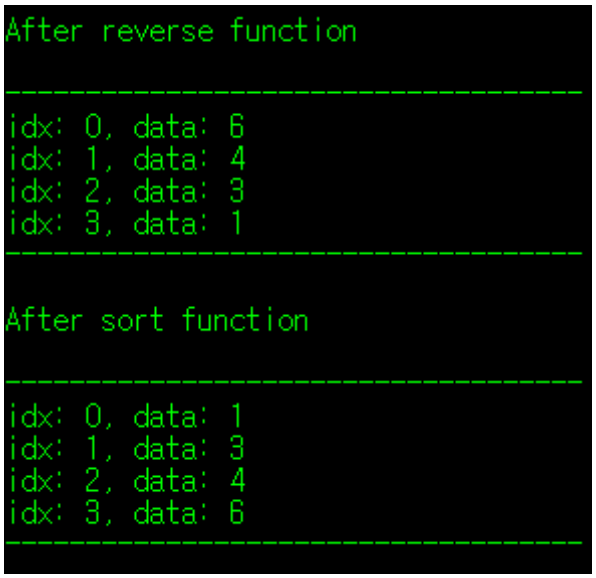
A common Linked List's head doesn't have data. Head don't have data, it is useful to controll all list.

Question 2

1. Explanation

In this program, user can reverse or sort the data of linked list. When reverse or sort method works, data must be relocated on a per-node basis.

2. Result screen



3. Study

My sorting method uses too many elements to sort. Maybe there are more useful sorting method, but I have no idea.

Question 3

1. Explanation

This program reads a list of words from a file(input.txt) and stores them as 2D-linked list format. Words must be sorted alphabetically, and program can print all arranged words.

2. Result screen

3. Study

Using 2D-LinkedList can store data effectively. 2D-LinkedList can label list.

Question 4

1. Explanation

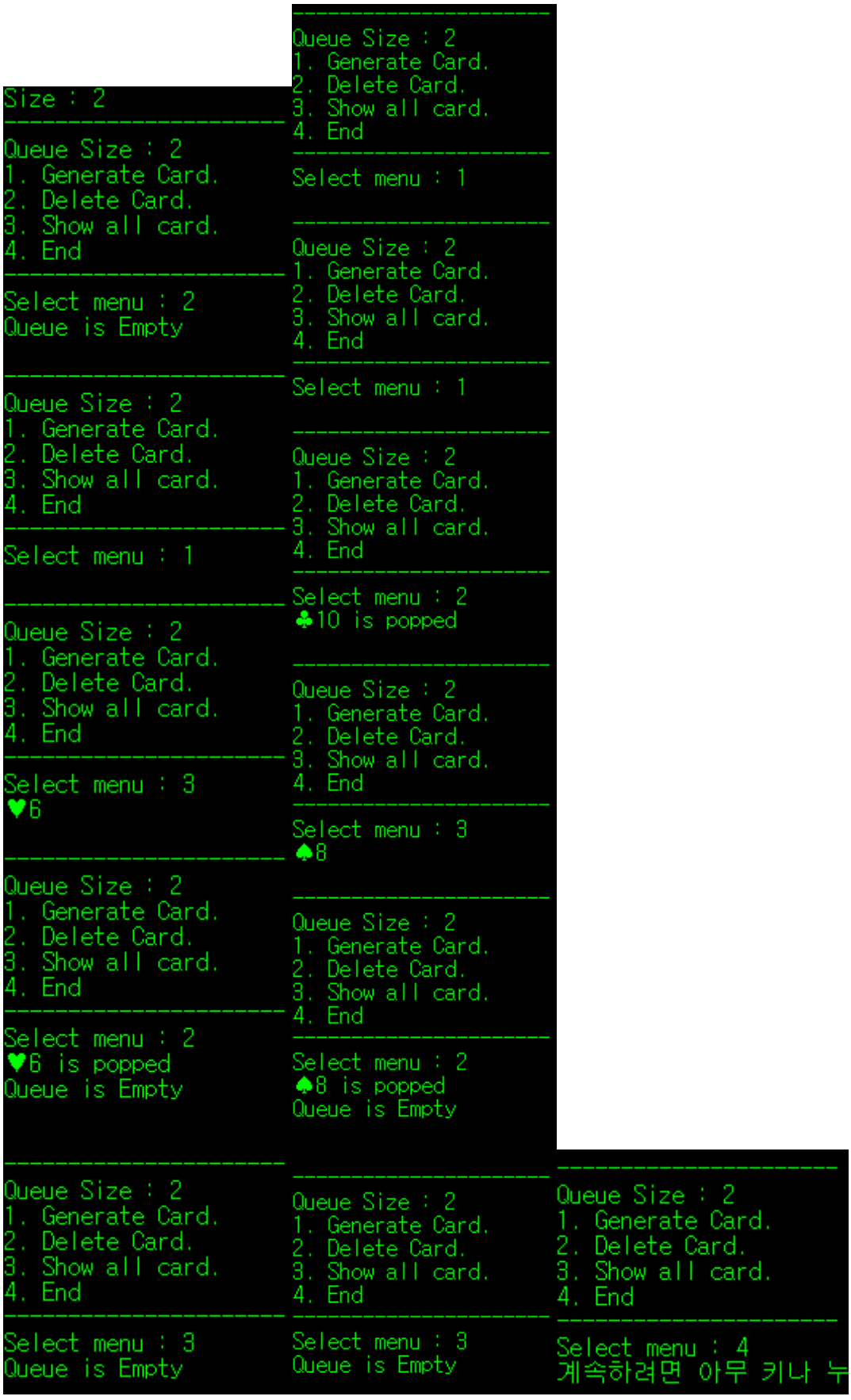
This program has 4 options, Generate Card, Delete Card, Show all card, End. User chooses Gererate Card, card is generated randomly and put the card into the Queue. All cards must be different. Delete Card is deleting card which is previously generated card. Show all card shows all card stored in Queue. If stored card does not exist, print massage. User selects End, the program ends

2. Study

Since unicord is 2-byte data, char type could not be used. But string type can take unicord letter. Generating random card, it is difficult to ideate generating card method. All card must be different, so my algorithm compares new card with all pregenerated cards. It was my best.

3. Result screen

Next Page.


Size : 2
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 2
Queue is Empty
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 1
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 3
♥6
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 2
♥6 is popped
Queue is Empty
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 3
Queue is Empty
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 1
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 1
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 2
♣10 is popped
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 3
♠8
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 2
♠8 is popped
Queue is Empty
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 3
Queue is Empty
Queue Size : 2
1. Generate Card.
2. Delete Card.
3. Show all card.
4. End
Select menu : 4
계속하려면 아무 키나 누

Question 5

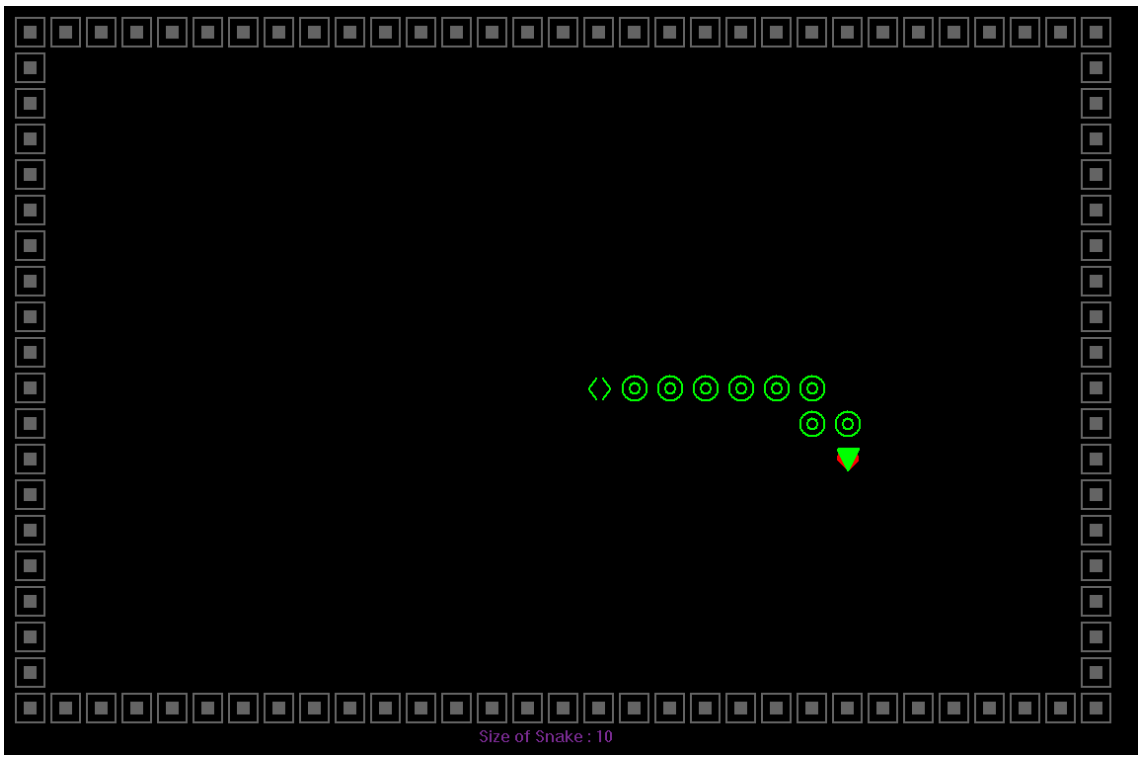
1. Explanation

This program is snake game. The 3-size snake crawls playground, gets items, and grow up. When snake gets out of prayground or snake's head over its body, the game is over. If user want to play one more, press y. The snake has to be composed by using linked list.

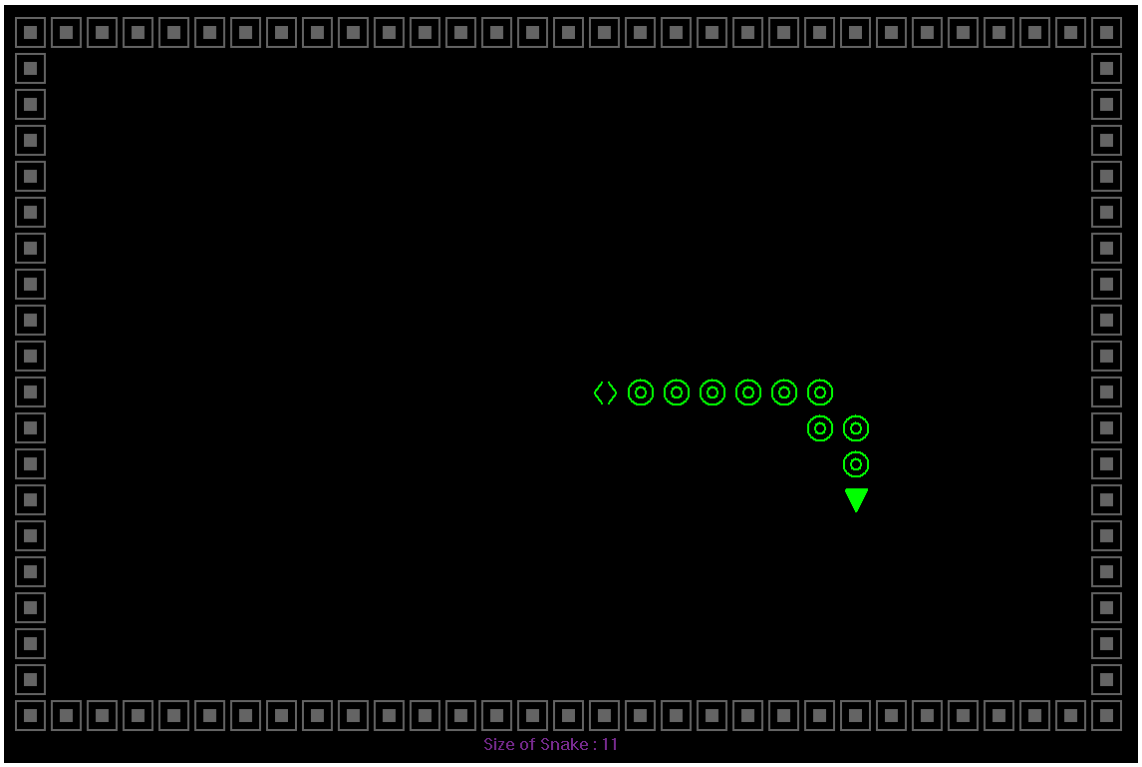
2. Study

Using MFC, it is difficult where I insert my cords. What I find way to turn a infinite loop is Using Thread, but I don't know how use it, eventually I couldn't infinite loop. As a result, the program is incomplete. If I have more time, maybe I can complete it.

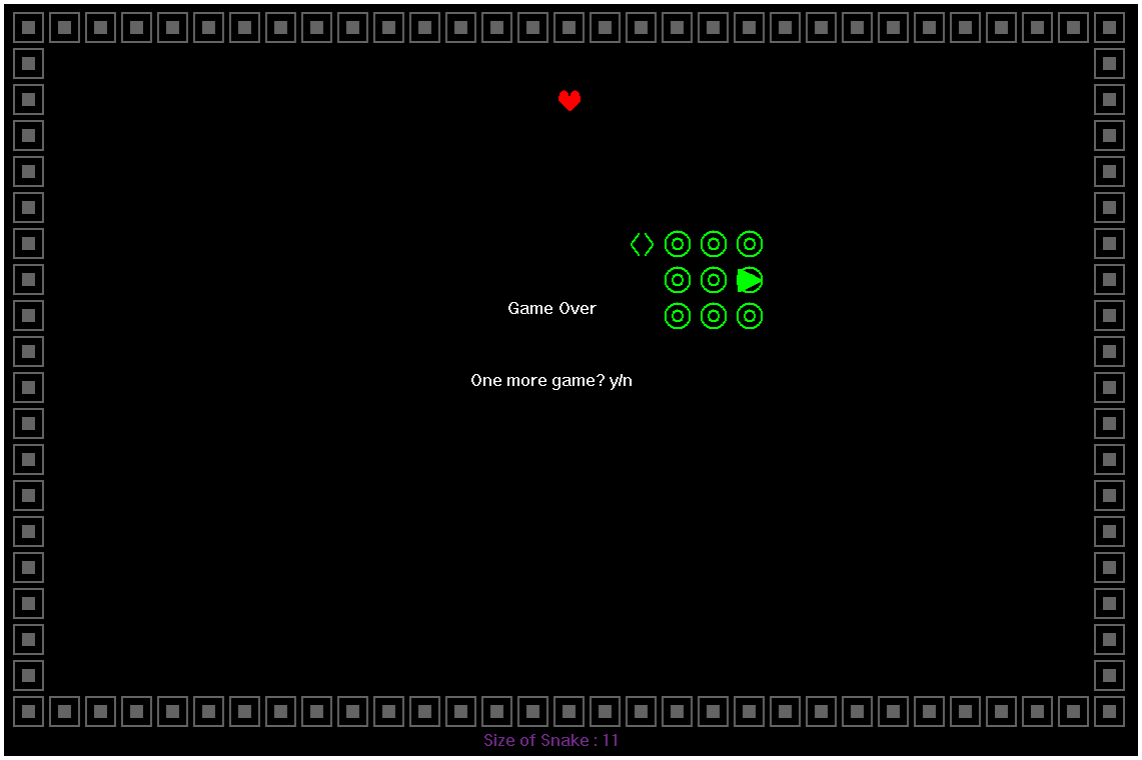
3. Result screen



- Getting item -



- Grow up 1 size -



- When snake catch it’s body -