

객체지향프로그래밍

2-4

1. 아래의 main 함수가 동작해서 실행 예제와 같이 결과가 출력되도록 Overloading Class를 완성하고 아래 main함수를 실행하는 프로그램을 구현하시오. main함수는 아래의 코드에서 수정하지 않는다.

```
int main()
{
    class Overloading A;
    int int_a = 5, int_b = -10;
    float float_a = 5.1, float_b = 5.3;
    unsigned int unsigned_int_a = 5, unsigned_int_b = 5;

    A.add(int_a, int_b);
    A.add(float_a, float_b);
    A.add(unsigned_int_a, unsigned_int_b);
    return 0;
}
```

<Main 함수>

```
Int type add : -5
Float type add : 10.4
Unsigned int type add : 10
```

<실행 예제>

2. 아래의 변수들을 가지는 Student class를 이용해 학생의 이름과 점수를 관리, 출력하는 프로그램을 구현하시오. 프로그램은 4개의 command(1 : insert, 2 : sort by name, 3 : sort by score, 4. quit)을 가지며 아래 예시와 같이 insert는 이름, 점수를 받아 저장하고 sort by name은 이름 순으로 출력, sort by score는 점수 순으로 출력, quit는 프로그램을 종료한다. Student class의 멤버 변수는 추가할 수 없으며 멤버함수는 자유롭게 추가할 수 있다.

```
class Student
{
private:
    char name[20];
    int Score;
};
```

<Student class>

```

Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :1
Name :박우혁
Score :40
Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :1
Name :안재원
Score :65
Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :1
Name :홍길동
Score :15
Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :2
박우혁 40
안재원 65
홍길동 15
Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :3
홍길동 15
박우혁 40
안재원 65
Please Enter Command(1:insert,2:sort by name,3:sort by score,4.quit) :4

```

<실행 예제>

3. Linked List를 이용해 값을 저장, 탐색, 삭제, 출력하는 프로그램을 구현하시오. 프로그램은 5개의 command(insert, find, delete, print, quit)를 가지고 있으며 동작은 아래 표와 같다. 프로그램은 실행 예제와 같이 insert를 이용해 값을 저장, delete를 이용해 삭제하며 find를 이용해 값을 출력, quit를 이용해서 제거한다.

Command	Format	설명
insert	Insert 숫자	수를 Linked List에 저장
find	find 숫자	Linked List에 입력받은 수가 있을 경우 저장되어있는 Index를 모두 출력
delete	Delete 숫자	Linked List에 입력받은 수가 있을 경우 모두 제거
print	print	모든 data를 출력
quit	quit	프로그램 종료

<Command List>

```

Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :insert 5
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :insert 13
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :insert 5
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :insert 8
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :print
5
13
5
8
Number of nodes in the list: 4
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :find 5
Found Index:1
Found Index:3
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :delete 5
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :print
13
8
Number of nodes in the list: 2
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :find 5
Please Enter Command(1:insert,2:find,3:delete,4.print,5.quit) :quit

```

<실행 예제>

4. Linked List를 이용해 끝말잇기 프로그램을 구현하시오. 프로그램은 사용자로부터 단어를 입력 받아 끝말이 이어지는지 아닌지, 이전에 입력을 받았는지 받지 않았는지 검사해서 조건에 만족할 경우 단어를 저장한다. 단 단어의 대,소문자는 구별하지 않고 같은 알파벳으로 취급한다.

```
CMD(Word/exit)>> Print
Print->
CMD(Word/exit)>> truck
Print->truck->
CMD(Word/exit)>> full
Not Chained
Print->truck->
CMD(Word/exit)>> key
Print->truck->key->
CMD(Word/exit)>> yep
Print->truck->key->yep->
CMD(Word/exit)>> print
Already Exists
Print->truck->key->yep->
CMD(Word/exit)>> exit
```

<실행 예시>