

Assignment #1

Classes**Due date: ~10/20(Fri), 23:59**

이번 과제에서는 “피자가게 메뉴 관리 및 주문 시스템”을 구현한다. 이를 통하여 클래스를 이용한 객체 지향 프로그래밍의 기초를 연습한다.

1. 클래스 설명

피자는 토핑 종류, 도우 종류, 에지 종류에 대해 각각 1개씩, 개별 선택하여 주문할 수 있다. 이 세 종류의 옵션을 서로 다른 클래스로 구현한다. 동일한 클래스의 인스턴스들은 서로 링크드 리스트 형태로 연결된다. 각 종류는 아래와 같이 구성되어야 하며, 명시된 멤버 변수 및 멤버 함수는 동일한 이름으로 반드시 포함되어야 한다. 링크드 리스트 사용을 위해서, 연결 노드를 가리키는 포인터 변수와, 그 변수를 다루는 함수 추가를 허용한다(자유롭게 추가하여 사용하는 것을 허용. STL, Boost와 같은 라이브러리의 구현된 리스트를 사용하는 것이 금지됨). 그 외 필요에 의한 멤버 변수나 멤버 함수 추가도 허용하지만, 이 경우 반드시 보고서에 그 이유를 명시하여야 한다.

Topping	Dough	Edge
<ul style="list-style-type: none"> - name - ingredient - price - sell_count - comment 	<ul style="list-style-type: none"> - name - thickness - extra_cost - sell_count - comment 	<ul style="list-style-type: none"> - name - mousse - extra_cost - sell_count - comment
<ul style="list-style-type: none"> +set_name(string) +add_ingredient(string) ... +get_name() +get_ingredient() ... 	<ul style="list-style-type: none"> +set_name(string) +set_thickness(int) ... +get_name() +get_thickness() ... 	<ul style="list-style-type: none"> +set_name(string) +add_mousse(string) ... +get_name() +get_mousse() ...

- 각 멤버 변수들은 private으로 선언하도록 하며, 변수로의 접근은 반드시 public 멤버 함수로 이루어져야 한다.
- 이름(name)은 string을 사용한다.
- 판매가격(price), 추가비용(extra_cost)은 int을 사용한다.

- 판매횟수(sell_count)는 int를 사용한다.
- 소개말(comment)은 string을 사용한다.
- Topping 클래스의 재료(ingredient)는 string[10]을 사용한다.
- Dough 클래스의 두께(thickness)는 float을 사용한다.
- Edge 클래스의 무스(mousse)는 string[3]을 사용한다.
- 멤버 함수는 모든 멤버 변수에 대한 값을 넣고, 얻는 것이 제공되어야 한다. 해당 변수가 배열인 경우 함수명의 접두사로 add_를 사용하고(이 경우 하나씩 입력 받는다), 단일 변수라면 set_을 사용하며, sell_count에 대응되는 함수는 inc_sellcount()를 사용하여 기존의 sell_count에 1을 더하는 기능을 한다. 값을 얻는 것은 get_을 사용한다.

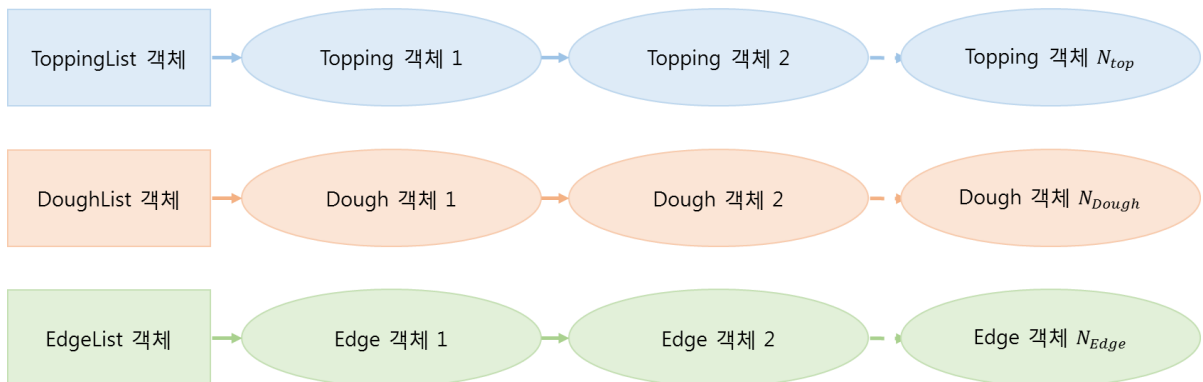
링크드 리스트로 관리하기 위해서, 각 클래스에 대응되는 리스트 클래스를 추가로 정의한다. 또한, 아래의 멤버 변수 및 멤버 함수는 동일한 이름으로 반드시 포함하여야 한다. 필요에 따라 멤버 변수와 함수를 추가하는 것을 허용하지만, 이 경우 반드시 보고서에 그 이유를 명시하여야 한다.

ToppingList	DoughList	EdgeList
- head - size	- head - size	- head - size
+add(*Topping) +del(string) +pick(string) +show_all_menu() +show_most_popular() +search_by_ingred(string) +search_by_price(int, int)	+add(*Dough) +del(string) +pick(string) +show_all_menu() +show_most_popular() +search_by_thick(int, int)	+add(*Edge) +del(string) +pick(string) +show_all_menu() +show_most_popular() +search_by_mousse(string)

- 멤버 변수 head는 포인터 변수로 리스트의 첫 번째 인스턴스의 주소를 저장한다.
- 멤버 변수 size는 int를 사용하며, 현재 리스트의 크기(총 인스턴스의 수)를 저장한다.
- 멤버 함수 add()는 리스트에 해당 메뉴를 추가하기 위해 사용되는 함수이다. 메뉴를 추가하기 위해서, 해당 인스턴스의 주소를 받아 추가한다.
- 멤버 함수 del()은 string 타입의 메뉴 이름을 입력 받아, 해당 메뉴를 리스트로부터 제거한다.
- 멤버 함수 pick()은 string 타입의 메뉴 이름을 입력 받아, 해당 메뉴의 주소를 돌려준다. 메인 프로그램의 "주문하기" 메뉴 등에서 사용될 수 있다.

- 멤버 함수 `show_all_menu()`는 `sell_count`를 제외한 모든 메뉴의 내용을 출력한다.
- 멤버 함수 `show_most_popular()`는 가장 많이 팔린 메뉴의 `sell_count`를 제외한 내용을 출력한다.
- `ToppingList` 클래스의 멤버 함수 `search_by_ingred()`는 `string` 타입의 재료명을 입력 받아, 해당 재료를 사용하는 모든 메뉴들의 내용을 출력한다.
- `ToppingList` 클래스의 멤버 함수 `search_by_price()`는 `int` 타입의 가격 범위(낮은 가격, 높은 가격)를 입력 받아, 해당 가격 범위 내에 속하는 메뉴들의 내용을 출력한다.
- `DoughList` 클래스의 멤버 함수 `search_by_thick()`는 `float` 타입의 두께 범위(낮은 값, 높은 값)를 입력 받아, 해당 두께 범위 내에 속하는 메뉴들의 내용을 출력한다.
- `EdgeList` 클래스의 멤버 함수 `search_by_mousse()`는 `string` 타입의 무스 이름을 입력 받아, 해당 무스를 사용하는 메뉴들의 내용을 출력한다.

앞에서 정의한 각 메뉴 클래스와 그에 대응되는 리스트 클래스는 메인 프로그램에서 인스턴스들을 통하여 다음 그림과 같은 구조를 가지게 된다.



2. 프로그램 설명

메인 프로그램에서는 토핑, 도우, 에지를 각각 담당하는 3개의 리스트(ToppingList, DoughList, EdgeList) 변수(인스턴스)를 선언하고 이를 통해 메뉴들을 관리한다. 각 메뉴의 초기 정보는 "init_menu.txt" 파일로부터 정보를 얻는다. 해당 파일은 토핑, 도우, 에지에 대한 정보가 적혀있는 파일로, 다음과 같이 구성되어있다.

```
init_menu.txt x
-토핑↓
슈퍼슈프림      피망,양파,파프리카,버섯 23500  가장 평범한 피자↓
불고기 불고기,피망,양파,파프리카,버섯 24000 불고기 향이 강한 피자↓
베이컨포테이토 웨지감자,베이컨,피망,양파,버섯 25000 베이컨과 감자 맛 피자↓
-도우↓
썬      0.5      0      토핑 맛을 살려줌↓
팬      2        0      배부름↓
프리미엄      1      5000  비쌘↓
-에지↓
리치골드      고구마 1000  고구마와 스트링 치즈맛↓
리치골드멜로  고구마,감자 1000  감자와 스트링 치즈맛↓
치즈크러스트      500  스트링 치즈맛↓
치즈바이트      고구마 1000  고구마 맛 약간과 스트링 치즈맛←
```

init_menu.txt 내용

레이블이 아닌 각 행은 하나의 메뉴를 의미하며, 각 메뉴의 서로 다른 정보는 탭(\t)으로 구분되어 있다. Topping 클래스의 재료(ingredient)와 Edge 클래스의 무스(mousse)는 둘 이상일 경우, 반점(.)으로 구분되어있다. 각 종류별 정보 순서는 다음과 같다.

- 토핑: <이름>\t<재료>\t<가격>\t<소개말>
- 도우: <이름>\t<두께>\t<추가비용>\t<소개말>
- 에지: <이름>\t<무스>\t<추가비용>\t<소개말>

해당 파일로부터 정보를 읽은 이후에, 사용자는 다음과 같은 명령 중 하나를 수행할 수 있다.

0) 모든 메뉴 보기

- A. 현재 등록되어있는 토핑, 도우, 에지에 대한 모든 메뉴들의 내용을 보여준다.

1) 주문하기

- A. 토핑, 도우, 에지 순으로 선택 가능한 항목을 차례로 보여주며, 메뉴 이름을 입력받

는다. 입력을 다 받고 난 이후에는 주문한 피자명("<도우> <예지> <토핑>")을 보여 주며, 해당 피자의 가격을 보여준다. 또한, 내부적으로는 주문된 종류들에 대해서 sell_count 값을 1 증가시켜 "인기메뉴 추천"에서 해당 값을 이용할 수 있도록 한다.

2) 메뉴 추가

- A. 메뉴의 종류를 먼저 입력 받은 후, 해당 메뉴에 필요한 정보들을 차례로 입력 받아 메뉴를 추가시킨다.

3) 메뉴 삭제

- A. 메뉴의 종류를 먼저 입력 받은 후, 삭제하고자 하는 메뉴의 이름을 입력 받아 해당 메뉴를 삭제시킨다. 입력 받은 메뉴 이름이 없다면, 삭제에 실패했다고 알리고 종료 된다.

4) 메뉴 검색

- A. 메뉴의 종류를 먼저 입력 받은 후, 검색 기준을 선택하게 한다. 검색 기준은, 토핑의 경우 재료이름 또는 가격범위, 도우의 경우 두께범위, 예지의 경우 무스이름이 된다. 해당 검색 기준에 따라 적절한 입력을 추가로 받아, 그에 해당하는 메뉴들을 모두 등록된 순으로 보여준다.

5) 인기메뉴 추천

- A. 각 종류 별로 현재까지 가장 많이 주문된 메뉴의 정보를 보여준다. "주문하기"를 통해 계산된 sell_count 값들에서 가장 큰 값을 가지는 메뉴들을 보여준다. 가장 큰 값을 가지는 메뉴가 둘 이상인 경우, 먼저 등록된 메뉴를 보여준다.

6) 프로그램 종료

- A. 프로그램을 종료시킨다. 편의상 다른 기능은 추가하지 않는 것으로 한다.

3. 프로그램 예시 (빨강: 사용자 입력, 파랑: 주요 결과)

```
=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 0

-TOPPING <3>
[이름:슈퍼슈프림][재료:피망,양파,파프리카,버섯][가격:23500원][소개말:가장 평범한
피자]
[이름:불고기][재료:불고기,피망,양파,파프리카,버섯][가격:24000원][소개말:불고기
향이 강한 피자]
[이름:베이컨포테이토][재료:웨이감자,베이컨,피망,양파,버섯][가격:25000원][소개말:
베이컨과 감자 맛 피자]

-DOUGH <3>
[이름:썸][두께:0.5mm][추가비용:0원][소개말:토핑 맛을 살려줌]
[이름:팬][두께:2mm][추가비용:0원][소개말:배우름]
[이름:프리미엄][두께:1mm][추가비용:5000원][소개말:비쌈]

-EDGE <4>
[이름:리치골드][무스:고구마][추가비용:1000원][소개말:고구마와 스트링 치즈맛]
[이름:리치골드옐로][무스:고구마,감자][추가비용:1000원][소개말:감자와 스트링 치즈
맛]
[이름:치즈크러스트][무스: ][추가비용:500원][소개말:스트링 치즈맛]
[이름:치즈바이트][무스:고구마][추가비용:1000원][소개말:고구마 맛 약간과 스트링
치즈맛]
```

0. 모든 메뉴 보기

```
=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 1

-TOPPING <3>
[이름:슈퍼슈프림][재료:피망,양파,파프리카,버섯][가격:23500원][소개말:가장 평범한
피자]
[이름:불고기][재료:불고기,피망,양파,파프리카,버섯][가격:24000원][소개말:불고기
향이 강한 피자]
[이름:베이컨포테이토][재료:웨이감자,베이컨,피망,양파,버섯][가격:25000원][소개말:
베이컨과 감자 맛 피자]

원하시는 토핑 이름을 입력하세요.
>> 베이컨포테이토

-DOUGH <3>
[이름:썸][두께:0.5mm][추가비용:0원][소개말:토핑 맛을 살려줌]
[이름:팬][두께:2mm][추가비용:0원][소개말:배우름]
[이름:프리미엄][두께:1mm][추가비용:5000원][소개말:비쌈]

원하시는 도우 이름을 입력하세요.
>> 썸

-EDGE <4>
[이름:리치골드][무스:고구마][추가비용:1000원][소개말:고구마와 스트링 치즈맛]
[이름:리치골드옐로][무스:고구마,감자][추가비용:1000원][소개말:감자와 스트링 치즈
맛]
[이름:치즈크러스트][무스: ][추가비용:500원][소개말:스트링 치즈맛]
[이름:치즈바이트][무스:고구마][추가비용:1000원][소개말:고구마 맛 약간과 스트링
치즈맛]

원하시는 에지 이름을 입력하세요.
>> 리치골드

주문하신 피자는 "썸 리치골드 베이컨포테이토" 입니다.
가격은 "26000"원 입니다.
```

1. 주문하기

```

주문하신 피자는 "썸 리치골드 베이컨포테이토" 입니다.
가격은 "26000"원 입니다.
=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 2
추가하고자 하는 메뉴의 종류를 선택하세요.
1: 토핑
2: 도우
3: 에지
>> 2
이름
>> 찰도우
두께
>> 0.2
추가비용
>> 100
소개말
>> 쫄깃한 식감
메뉴 추가가 완료되었습니다.

-DOUGH <4>
[이름:썸][두께:0.5mm][추가비용:0원][소개말:토핑 맛을 살려줌]
[이름:팬][두께:2mm][추가비용:0원][소개말:배무름]
[이름:프리미엄][두께:1mm][추가비용:5000원][소개말:비쌌]
[이름:찰도우][두께:0.2mm][추가비용:100원][소개말:쫄깃한 식감]

```

2. 메뉴 추가

```

=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 3
삭제하고자 하는 메뉴의 종류를 선택하세요.
1: 토핑
2: 도우
3: 에지
>> 3
-EDGE <4>
[이름:리치골드][무스:고구마][추가비용:1000원][소개말:고구마와 스트링 치즈맛]
[이름:리치골드엘로][무스:고구마,감자][추가비용:1000원][소개말:감자와 스트링 치즈
맛]
[이름:치즈크러스트][무스: ][추가비용:500원][소개말:스트링 치즈맛]
[이름:치즈바이트][무스:고구마][추가비용:1000원][소개말:고구마 맛 약간과 스트링
치즈맛]

삭제하고자 하는 메뉴 이름을 입력하세요.
>> 리치골드엘로
메뉴 삭제가 완료되었습니다.

-EDGE <3>
[이름:리치골드][무스:고구마][추가비용:1000원][소개말:고구마와 스트링 치즈맛]
[이름:치즈크러스트][무스: ][추가비용:500원][소개말:스트링 치즈맛]
[이름:치즈바이트][무스:고구마][추가비용:1000원][소개말:고구마 맛 약간과 스트링
치즈맛]

```

3. 메뉴 삭제

```

=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 4
검색하고자 하는 메뉴의 종류를 선택하세요.
1: 토핑
2: 도우
3: 예지
>> 1
검색 기준을 선택하세요.
1: 재료로 검색
2: 가격으로 검색
>> 1
재료 이름을 입력해주세요.
>> 불고기
[이름: 불고기][재료: 불고기, 피망, 양파, 파프리카, 버섯][가격: 24000원][소개말: 불고기
향이 강한 피자]
이상 총 1개가 검색되었습니다.

```

4-1. 메뉴 검색 (재료)

```

=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 4
검색하고자 하는 메뉴의 종류를 선택하세요.
1: 토핑
2: 도우
3: 예지
>> 2
검색 기준을 선택하세요.
1: 두께로 검색
>> 1
두께 범위 중 하한을 입력해주세요.
>> 1.0
두께 범위 중 상한을 입력해주세요.
>> 2.2
[이름: 팬][두께: 2mm][추가비용: 0원][소개말: 배무름]
[이름: 프리미엄][두께: 1mm][추가비용: 5000원][소개말: 비쌘]
이상 총 2개가 검색되었습니다.

```

4-2. 메뉴 검색 (두께)


```

주문하신 피자는 "썸 리치골드 베이컨포테이토" 입니다.
가격은 "26000"원 입니다.
=====
0: 모든 메뉴 보기
1: 주문하기
2: 메뉴 추가
3: 메뉴 삭제
4: 메뉴 검색
5: 인기메뉴 추천
6: 프로그램 종료
=====
>> 5
-TOPPING 인기메뉴
[이름: 베이컨 포테이토][재료: 웨지감자, 베이컨, 피망, 양파, 버섯][가격: 25000원][소개말: 베이컨과 감자 맛 피자]
-DOUGH 인기메뉴
[이름: 썸][두께: 0.5mm][추가비용: 0원][소개말: 토핑 맛을 살려줌]
-EDGE 인기메뉴
[이름: 리치골드][무스: 고구마][추가비용: 1000원][소개말: 고구마와 스트링 치즈맛]

```

5. 인기메뉴 추천

Requirements

- 이번 과제의 프로그램은 프로그래밍 실습서버의 리눅스 환경 및 MinGW(Windows)에서 컴파일 및 구동 가능하도록 작성한다.
- 본 과제의 소스코드는 다음과 같이 13개의 파일로 구성하기를 권장한다. 다른 형태의 파일 구성도 허용하지만, 그에 대한 이유를 보고서에 명시하여야 한다.
 - 메인함수를 포함하는 .cpp 파일
 - 각 메뉴 클래스(Topping, Dough, Edge)의 선언 및 정의에 대한 .cpp 및 .h 파일
 - 각 메뉴 리스트 클래스(ToppingList, DoughList, EdgeList)의 선언 및 정의를 위한 .cpp 및 .h 파일
- 반드시 Makefile을 작성하여 컴파일이 가능하도록 해야 하며, 작성된 Makefile 역시 제출해야 한다. (make를 통한 컴파일 실패 시, 컴파일이 안되는 것으로 간주)
- 실습서버의 접속 방법은 첨부된 [프로그래밍 실습환경 사용가이드]의 '실습 시스템 접속 방법'을 참조한다.
- 상기한 두 환경(Linux 및 Windows)에서의 프로그램 컴파일 및 실행 과정에 대한 설명과 이미지 모두 보고서에 포함해야 한다.
- 각 메뉴와 리스트는 모두 클래스로 구현하여야 한다. (클래스 미사용 시 0점)

- 상속(Inheritance)은 사용하지 않는다. (사용할 경우 0점)
- STL이나 Boost 등의 라이브러리는 사용하지 않는다. (사용할 경우 0점)
- 구성원들의 모든 멤버 변수는 'private'이어야 하며, 이들 멤버 변수에 대한 접근은 모두 'public' 멤버 함수들을 통해서만 이루어져야 한다.
- 위에 설명된 프로그램의 형식을 모두 만족하여야 한다.
- 프로그램은 사용자가 종료를 선택하기 전까지 반복 실행되어야 한다.
- 모든 예외 상황(잘못된 명령, 잘못된 메뉴 선택, 파일 열기 실패, 조회나 수정 또는 삭제 시 메뉴가 존재하지 않는 경우 등)에 대해 적절한 에러 메시지를 출력해야 한다.
- 보고서는 사용된 클래스의 멤버 변수/함수에 대한 설명을 포함하여야 한다.
- 보고서는 위의 모든 요구조건을 확인할 수 있는 시나리오와 그 결과를 캡처한 그림을 포함하여야 한다.
- 보고서에 위 요구사항 중 만족한 것과 만족하지 못한 것을 정확하게 명시하여야 하며, 정확하게 명시하지 않을 경우 '프로그램 설계 및 구현', '보고서 구성 및 내용, 양식' 점수가 감점될 수 있다.
- 보고서 및 소스코드는 LMS의 과제 디렉토리에 업로드한다.
- 채점 기준은 AssnReadMe_Fall_2017.pdf 파일을 참고한다.