

Midterm Exam Review

프로그래밍 입문(2)

After Midterm

- 앞으로는 본격적으로 C++에서의 객체지향 프로그래밍에 대해 학습하게 됩니다.
- 이제부터는 단순히 코드를 실행해보는 실습이 아닌 본격적으로 코드를 작성하는 실습을 진행할 것입니다.
- 반복해서 코드 작성하는 연습을 최대한 할 수 있고, 작은 규모의 프로그램을 완성하는 실습이 진행됩니다.
- 강의 시간에 배운 내용을 토대로, 주어진 코드를 완성하는 실습이며, 각 실습마다 출석체크 + 10점의 점수가 부여됩니다.

코딩 테스트 방식 평가

- 매 실습에는 여러분이 작성해야할 코드의 요구사항과 함께 테스트 케이스를 실행할 수 있는 test.cpp라는 파일이 주어집니다. test.cpp를 컴파일하고 실행하면 (통과한 테스트 수/전체 테스트 수)가 출력됩니다.
- 제출시 기본점수 0점 (출석체크만), 코드 컴파일이 가능하고 test.cpp의 모든 테스트를 통과하면 5점이 주어집니다.
- 나머지 5점은 주어진 요구사항을 잘 구현하였는지 확인하는 공개되지 않은 추가 테스트 케이스를 실행하여 이를 모두 통과하였을 경우에만 부여됩니다.
- 즉, 총 5개의 테스트가 있다고 하면, 이 중 3개만 test.cpp로 공개가 되고, 나머지는 제출 후에 검사에서만 사용됩니다. 따라서 요구사항을 잘 읽고 코드를 작성하셔야 합니다.

제출 방식 엄수

- 객체지향 프로그래밍 관련 실습을 진행하면, 다수의 파일을 이용해 코드를 작성하게 되므로 제대로 된 실행을 위해 반드시 제출 방식을 엄수해 주셔야 합니다.
- 작성해야할 코드는 따로 설명하지 않는한 classes내에 파일이 저장됩니다.
- 이 classes 폴더에서 *.hh와 *.cpp 파일만을 남기고 classes 폴더 자체를 zip, 또는 tar파일로 묶어 하나의 파일로 만들어 제출합니다.
 - 압축을 해제하면 classes 폴더가 나와야 합니다.
 - 이상한 다른 파일이 들어 있어 제대로 테스트가 실행되지 않으면 제출만 한 것으로 간주하여 0점이 부여됩니다.
 - rar, alz 같이 다른 포맷의 파일로 묶어 정상적으로 해제가 되지 않는 경우도 마찬가 지입니다.

Topics

- **Midterm Exam Review**
- OOP Basics
- Encapsulation

Q1 (10점). 다음의 빈 칸에 알맞은 단어를 쓰거나 질문에 답하세요.

- (a) (2점) 모든 C++프로그램의 시작지점은 main함수 이다.
- (b) (2점) 여러 사람이 코드를 보는 경우에는 Code Style 을 일치시키는 것이 가독성 향상을 위해 매우 중요하다.
- (c) (2점) C++에서의 함수는 프로그래밍 언어의 제어 추상화 를 위해 도입한 서브프로그램을 일컫는 말이다.
- (d) (2점) 효율적인 실행을 위해 논리 연산자로 연결된 앞의 표현식 값에 따라 뒤의 표현식 평가를 생략하는 방법을 Short Circuiting 이라고 부른다.
- (e) (2점) 값을 저장하기 위한 공간을 마련하고 이름을 붙인 것 중, 값이 계속 변할 수 있는 것을 변수 , 초기화 후 값을 바꿀 수 없는 것을 상수 라고 한다.

Q2 (10점). 다음의 코드를 보고 질문에 답하세요.

```
1  #include<iostream>
2  #include<cstdio>
3  using namespace std;
4  int main() {
5      cout << FLT_DIG << endl;
6      float x = 123.00000001;
7      float y = 123.0;
8      if(x == y)
9          cout << "Same";
10     else
11         cout << "Diff";
12     return 0;
13 }
```

(a) 5번 라인의 코드가 6을 출력했다면, 8-11번 라인의 코드에서는 Same과 Diff 중 무엇이 출력되나요?

Same

(b) 만약 8-11번 라인을 바꾸지 않고 (a)에서와 다른 메시지를 출력하고 싶으면 어떻게 코드를 수정하면 되나요?

6,7번 라인의 float를 double로 수정

Q3 (20점). 다음 프로그램이 “We need guns. Lots of guns.”를 출력하도록 프로그램을 완성하세요. (따옴표안의 내용만 출력. *hint*:`replace()`, `find()`, `size()` 함수 이용)

```
1  #include<iostream>
2  #include<string>
3  using namespace std;
4  int main() {
5      string str = "We need toys. Lots of toys.";
6      string q = "toy";
7      int idx = str.find(q);
8      while(idx >= 0) {
9          str = str.replace(idx, q.size(), "gun") ;
10         idx = str.find(q,idx) ;
11     }
12     cout << str;
13 }
```


Q4 (20점). 다음 main() 함수 내의 코드를 보고 질문에 답하세요.

```
1  const char SPACE = ' ', CROSS='x';
2  char mark;
3  for(int i=0; i<3; i++) {
4      for(int j=0; j<3; j++) {
5          if(i > j)
6              mark = SPACE;
7          else
8              mark = CROSS;
9          cout << mark;
10     }
11     cout << endl;
12 }
```

(a) 이 코드가 화면에 출력하는 모양은 무엇인가요?
(빈 칸은 밑줄_로 표시)

```
x x x
_ x x
_ _ x
```

(b) 5-8번째 줄의 if문을 삼항 조건식으로 바꾸어 쓰세요.

```
mark = i > j ?
SPACE : CROSS;
```

Q5 (20점). 다음의 코드는 최대공약수 (Greatest Common Divisor, GCD) 를 구하는 유클리드 호제법 (Euclidean Algorithm) 을 반복적으로 구현한 함수 입니다. 이 gcd() 함수의 내부 구현을 재귀함수로 바꾸어 다시 작성하세요. ¹

```
1 int gcd(int a, int b) {  
2     int r;  
3     while(b != 0){  
4         r = a % b;  
5         a = b;  
6         b = r;  
7     }  
8     return a;  
9 }
```

¹유클리드 호제법에 대한 정보는 다음과 같습니다. 두 자연수 a, b ($a > b$)에 대하여 $a \div b$ 의 나머지를 r 이라하면, a, b 의 최대공약수 $\gcd(a, b)$ 는 $\gcd(b, r)$ 과 같다. 이 성질을 이용해 나머지가 0이 될 때까지 반복적으로 계산하여 최대공약수를 구하는 것이 유클리드 호제법입니다.

```
int gcd(int a, int b) {
    int r;
    while(b != 0){
        r = a % b;
        a = b;
        b = r;
    }
    return a;
}
```

```
int gcd(int a, int b) {
    if(b == 0)
        return a;
    else
        return gcd(b, a % b);
}

int gcd(int a, int b) {
    if(a % b == 0)
        return b;
    else
        return gcd(b, a % b);
}
```

Q6 (20점). 주어진 코드는 클래스 Product를 정의하는 부분과, 값을 받아 이 클래스의 객체를 선언, 동적할당하고 내용을 출력하는 print_product() 함수로 이루어져 있습니다.

```
1 class Product {
2     private:
3         string name;
4         int price;
5     public:
6         Product(string n, int p) {
7             name = n;
8             price = p;
9         }
10        void print() {
11            cout << "Name: " << name << endl;
12            cout << "Price: " << price << endl;
13        }
14 };
15
16 void print_product(string name, int price) {
17     Product *p = new Product(name, price);
18     p.print();
19 }
```

```
16 void print_product(string name, int price) {  
17     Product *p = new Product(name, price);  
18     p.print();  
19 }
```

이 `print_product()` 함수는 2가지 문제점을 가지고 있습니다. 문제점을 수정하기 위해 코드를 어떻게 고쳐야 하나요? 몇번째 줄을 어떻게 바꾸거나, 몇번째 줄의 뒤에 어떤 코드를 추가하거나, 몇번째 줄을 지우면 된다는 식으로 간단히 설명하면 됩니다.

(1): 18번째줄의 코드를 `p->print();`로 수정

(2): 18번째줄 다음에 `delete p;`를 추가

Summary

- Midterm Exam Review