**2016, 봄 학기, 고급 객체지향 프로그래밍 시험 4**

1. 택시와 버스를 모두 운영하는 운수회사의 차량을 관리하는 SW를 개발하려고 한다. 다음의 5가지 클래스를 만들고 이들 사이의 관계를 적절히 구성하라. (40점)
   1. VehicleManager 클래스
      1. 차량 관리 클래스. 여기에는 다량의 “Bus” 차량, “Taxi” 차량에 대한 데이터를 저장할 수 있어야 함
      2. 관리하는 차량을 추가하는 기능이 있어야 함
      3. 저장된 Bus, Taxi 에 드는 비용을 계산할 수 있어야 함 (인건비 포함).
      4. 저장된 Bus, Taxi들이 벌어드리는 수익을 계산할 수 있어야 함.
   2. Bus 클래스
      1. 출발-도착지가 정해져 있고 승객은 정해진 가격의 승차권을 사서 타는 형태
      2. 이름, 버스 승차권 가격, 일간 평균 승객 수를 저장할 수 있어야 함.
      3. 한달 수익 (가격 \* 일간 평균 승객 수 \* 30) 계산할 수 있어야 함.
      4. 월별 운행 비용을 저장할 수 있어야 함
      5. 운전사를 저장할 수 있어야 함.
   3. Taxi 클래스
      1. 출발-도착지가 정해져 있지 않아 승객은 주행거리당 비용을 지불하는 형태
      2. 이름, 주행거리당 택시비, 일회 평균 주행거리, 일일 평균 주행 횟수를 저장할 수 있어야 함
      3. 한달 수익 (거리당택시비\*주행거리\*주행횟수\*30)를 계산 할 수 있어야 함.
      4. 월별 운행비용을 계산할 수 있어야 함
      5. 운전사를 저장할 수 있어야 함.
   4. Driver 클래스
      1. 이름, 월급여를 저장할 수 있어야 함.
   5. 다음 Main 이 동작하게 위 3개의 클래스의 내용을 적절히 만들어라. 필요하다면 맴버함수, 변수추가 가능
   6. Bus와 Taxi 클래스는 공통적인 맴버변수와 맴버함수를 가진다. “상속” 이라는 기능으로 코드를 좀 더 효율적으로 만들 수 있다. (힌트: 공통된 부분을 묶어주는 공통 부모 클래스인 Vehicle클래스). 이를 수행하라. 그리고, 아래 main 코드를 분석해 보면 이 상속관계는 단순 상속관계가 아니고, Vehicle는 “순수가상함수”를 가지는 “추상클래스”로 만들어야 할 것 같다. 그렇게 만들어라. (힌트: 공통으로 가지는 함수는 순수가상함수로 만들기).
   7. **또 다른 힌트: VehicleManager 에서 Bus 리스트과 Taxi 리스트를 따로 관리하지 않고 다형성을 이용해 Vehicle의 리스트로 관리하면 아래 Main 가능!! 다형성을 적용하려면 리스트는 Vehicle \*의 배열이 되어야 함 (최대 Vehicle 수는 100개라고 가정)**
   8. 다음 Main 이 동작하게 만들어라.

VehicleManager KHUTransport;

Vehicle \* temp = new Bus(“따요버스”,1500, 150,200000, Driver(“타요,3000000)); //이름, 버스비, 일별승객수, 유지비용, 운전수(이름,월급)

KHUTransport.addVehicle(temp); //temp 가 가리키는 차 추가

…//최소 4가지의 버스 혹은 택시 추가

temp = new Taxi(“폴리”,200, 20, 20, 100000, Driver(“쿠쿠”, 2000000)); //이름, 주행거리당택시비, 주행거리, 주행횟수, 운전수

KHUTransport.addVehicle(temp);

cout << “한달수입: “ << KHUTrnasport.calculateTotalIncomes() << endl; //한달 총 수입: 모든 차량의 한달 수익 합

cout << “한달지출: “ << KHUTransport.calculateTotalExpenses() << endl; //한달 총 지출: 모든 차량의 유지비용 + 모든 운전사 인건비

1. 위 운수회사가 승승장구하여 게임 SW쪽으로 사업을 확장하려 한다. 이 회사에서 첫 번째 출시할 게임은 온라인 물총 대전 게임 “KHUKEN Street Fighter”으로, 두 사용자가 아바타를 조정하면서 서로 물총을 쏘며 싸우는 게임이다 (일인칭 슈팅게임과 1:1 대전게임의 결합!!). 당신은 지금 이 회사의 새로운 SW개발 부서의 개발 책임자이고, 이 쿠켄 스트리트파이터 게임을 객체지향 개념을 기반으로 디자인하려고 한다. 이 상황에서 여러분이 필요하다고 생각되는 클래스들과 이들의 관계를 디자인 하라. (10점)
   1. 코딩으로 구현할 필요는 없고 글로 cpp 파일에 comment 로 작성
   2. 게임 SW에서 필요한 객체가 어떤것을이 있을지(예, 아바타, 사용자, 무기 등등)를 생각해 보고 이를 위주로 디자인.
   3. 게임의 세부 기능 및 세부 스펙에 대한 정의가 필요하다고 생각되면 정의해도 됨
   4. 각 클래스의 역할, 대략적인 데이터 및 함수, 각 클래스간의 관계 (상속, Aggregation 등) 에 대한 기술 필요
   5. 정답은 없지만 완성도 및 적절성으로 평가할 예정
2. class template을 이용한 외계인 클래스를 구현하고 프로그램을 완성하시오. (25점)
   1. template 타입을 지정할 때에는 NAMEK , B612 class를 만들어 사용하시오. (아래 main 함수에서의 쓰임 참조.)

NAMEK와 B612는 각각 다음과 같은 멤버를 가지고 있다.

NAMEK : 전투력을 저장할 int형 변수

B612 : 고항별에 돌아가는 데까지 남은 시간을 저장할 long long int 형 변수

그리고 공통으로 고향별의 이름을 저장할 문자열 변수

멤버 변수에 값을 저장하는 setData() 함수와 저장된 데이터를 화면에 출력하는 printData()함수를 가지고 있다.

* 1. class template을 사용하여 CAlien 클래스를 구현하시오.
  2. CAlien 클래스의 멤버 변수.

문자열에 외계인 이름을 저장

template type 변수

* 1. public 메소드

생성자들, 소멸자 및void store() 와 void printInfo() 함수를 각 template type에 맞춰 모든 멤버 변수들을 저장 및 출력 가능하도록 구현하시오.

* 1. 다음 main 코드가 잘 동작하여 오른쪽 결과가 나오도록 CAlien클래스를 정의하시오.(특히 생성자) 그리고 data encapsulation 을 잘 하시오.

|  |  |
| --- | --- |
| #include "Alien.h"  int main()  {  CAlien<NAMEK> picolo("피콜로",1000);  CAlien<NAMEK> nail("네일",42000);  CAlien<B612> littlePrince("어린왕자", 300);  picolo.printInfo();  nail.printInfo();  littlePrince.printInfo();    return 0;  } | 피콜로  NAMEK  전투력 : 1000  네일  NAMEK  전투력 : 42000  어린왕자  B612  남은 시간 : 300 |

1. 위 CAlien 클래스를 이용하여 여러 외계인의 정보를 파일에 보관 관리 하는 프로그램을 만드시오. (30점)
   1. CAlien 클래스에 멤버 변수를 각각 반환하는 getName(),getData() 함수를 추가하시오. 그리고 다음의 함수를 추가하시오.

NAMEK class : 전투력을 반환하는 int getPower();

B612 class : 고항별에 돌아가는 데까지 남은 시간을 반환하는 long long int getRemainDate();

그리고 두 class에 공통으로 고향별의 이름을 반환하는 getHome()함수.

* 1. switch-case문으로 type을 선택하여 0이 입력될 때까지 CAlien 클래스의 객체를 생성하시오.

이 때 데이터를 각 타입의 멤버 변수에 맞도록 입력 받은 후 자료구조(STL 사용 가능, 사용하지 않아도 됨)에 저장하시오.

* 1. 생성된 객체의 정보를 모두 binary 파일 aliendata.bin에 저장하시오.

객체의 정보를 저장할 때 객체의 타입을 char형으로 NAMEK면 1, B612면 2로 먼저 저장하여 나중에 구분할 수 있도록 할 것.

그리고 CAlien 클래스의 멤버 변수 중 문자열들은 100byte 고정 길이로 저장하시오.

이 때 클래스 타입이 다르면 사이즈가 차이 나는 부분을 고려해서 객체 단위 random access 가능하도록 잘 파일에 저장할 것.

* 1. aliendata.bin에 저장된 CAlien 객체 중 사용자에게 입력 받은 인덱스에 해당하는 객체의 정보를 printInfo 함수를 이용하여 -1이 입력될 때까지 출력하시오.

|  |  |
| --- | --- |
| 1  피콜로  1000  2  어린왕자  300  1  네일  42000  0  2  1  -1 | 외계인 타입을 입력해주세요. 1. 나메크 2. B612 0. 종료 : 1  이름 : 피콜로  전투력 : 1000  외계인 타입을 입력해주세요. 1. 나메크 2. B612 0. 종료 : 2  이름 : 어린왕자  남은 기간 : 300  외계인 타입을 입력해주세요. 1. 나메크 2. B612 0. 종료 : 1  이름 : 네일  전투력 : 42000  외계인 타입을 입력해주세요. 1. 나메크 2. B612 0. 종료 : 0  출력할 외계인의 인덱스를 입력해주세요. (-1은 종료) : 2  네일  NAMEK  전투력 : 42000  출력할 외계인의 인덱스를 입력해주세요. (-1은 종료) : 1  어린왕자  B612  남은 시간 : 300  출력할 외계인의 인덱스를 입력해주세요. (-1은 종료) : -1 |

* 1. 좌측과 같은 입력에 대해 우측과 같이 화면에 실행되도록 프로그램을 잘 구현하시오.