고급프로그래밍및실습 과제 #4

214823 박종현

문제 #8

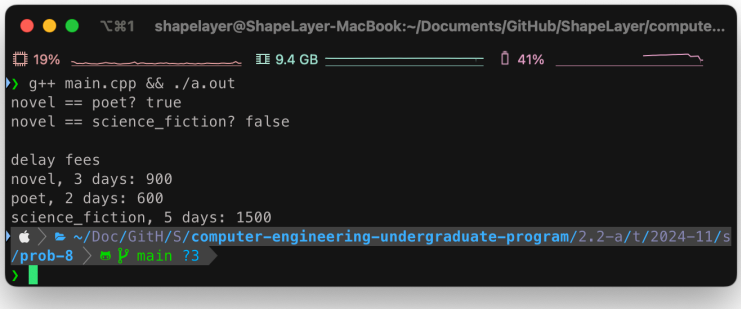
- `Book` 클래스: 관리번호(`number`), 제목(`title`), 저자(`author`)
- `Novel`, `Poet`, `ScienceFiction` 클래스: `Book` 상속
- `getLateFees(int delayed)` 구현
  - `Novel`: 300원/일, `Poet`: 200원/일, `ScienceFiction`: 600원/일

답안

```cpp
#include "bits/stdc++.h"

class Book {
private:
  int number;
  string title;
  string author;
public:
  int get_number() { return number; }
  void set_number(int value) { number = value; }

  string get_title() { return title; }
  void set_title(string value) { title = value; }

  string get_author() { return author; }
  void set_author(string value) { author = value; }

  bool operator==(const Book &other) const {
    return this->number == other.number;
  }

  virtual int getLateFees(int delayed) = 0;
  virtual ~Book() {};
};

class Novel: public Book {
public:
  int getLateFees(int delayed) { return 300 * delayed; }
};
class Poet: public Book {
public:
  int getLateFees(int delayed) { return 200 * delayed; }
};
class ScienceFiction: public Book {
public:
  int getLateFees(int delayed) { return 600 * delayed; }
};

int main() {
  Novel novel;
```

```
41    novel.set_number(1);
42    novel.set_title("Omniscient Reader");
43    novel.set_author("singNsong");
44
45    Poet poet;
46    poet.set_number(1);
47    poet.set_title("Prelude");
48    poet.set_author("Yun Dong-ju");
49
50    ScienceFiction science_fiction;
51    science_fiction.set_number(2);
52    science_fiction.set_title("Martian");
53    science_fiction.set_author("Andy Weir");
54
55    cout << "novel == poet? " << (novel == poet ? "true" : "false") << endl
56         << "novel == science_fiction? " << (novel == science_fiction ? "true" : "false") << endl;
57    cout << endl;
58    cout << "delay fees" << endl
59         << "novel, 3 days: " << novel.getLateFees(3) << endl
60         << "poet, 2 days: " << novel.getLateFees(2) << endl
61         << "science_fiction, 5 days: " << novel.getLateFees(5) << endl;
62  }
```

실행 결과



실행 1

```
1  g++ main.cpp && ./a.out
```
Shell

```
1  novel == poet? true
2  novel == science_fiction? false
3
4  delay fees
5  novel, 3 days: 900
6  poet, 2 days: 600
7  science_fiction, 5 days: 1500
```

문제 #9

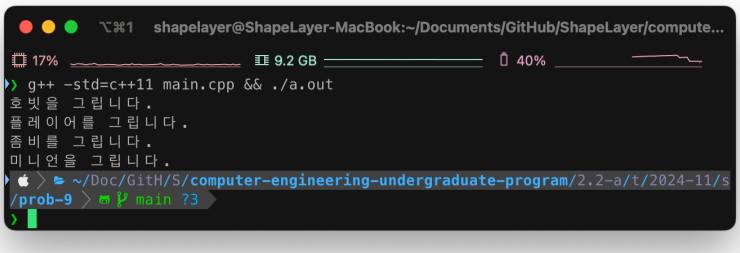
- `GameCharacter` 클래스: `draw` 가상 함수
- `GameCharacter` 클래스를 상속받는 캐릭터 클래스 정의
- `GameCharacter` 포인터를 담는 배열 구현

답안

```cpp
#include "bits/stdc++.h"
// Required C++11: Refer `call_draw_all_characters` function

class GameCharacter {
public:
  virtual void draw() = 0;
  static vector<GameCharacter*> characters;

  static vector<GameCharacter*> *getCharacters() {
    return &characters;
  }
};

vector<GameCharacter*> GameCharacter::characters;

class Player: public GameCharacter {
  void draw() {
    cout << "플레이어를 그립니다." << endl;
  }
};

class Zombie: public GameCharacter {
  void draw() {
    cout << "좀비를 그립니다." << endl;
  }
};

class Minion: public GameCharacter {
  void draw() {
    cout << "미니언을 그립니다." << endl;
  }
};

class Hobit: public GameCharacter {
  void draw() {
    cout << "호빗을 그립니다." << endl;
  }
};

void init_register_all_characters() {
  vector<GameCharacter*> *characters = GameCharacter::getCharacters();

  characters->push_back(new Hobit());
  characters->push_back(new Player());
  characters->push_back(new Zombie());
  characters->push_back(new Minion());
```

```
}

void call_draw_all_characters() {
  vector<GameCharacter*> *characters = GameCharacter::getCharacters();

  for (GameCharacter* character: *characters) {
    character->draw();
  }
}

int main() {
  init_register_all_characters();
  call_draw_all_characters();
  return 0;
}
```

실행 결과



실행 1

```shell
g++ -std=c++11 main.cpp && ./a.out
```

```
호빗을 그립니다.
플레이어를 그립니다.
좀비를 그립니다.
미니언을 그립니다.
```