

Sabanci University

Faculty of Engineering and Natural Sciences

CS204 Advanced Programming

Fall 2019-2020

Homework 7

Inheritance and Polymorphism

Due: **29/11/2019, 11:55pm**

No late submissions.

PLEASE NOTE:

Your program should be a robust one such that you have to consider all relevant programmer mistakes and extreme cases; you are expected to take actions accordingly!

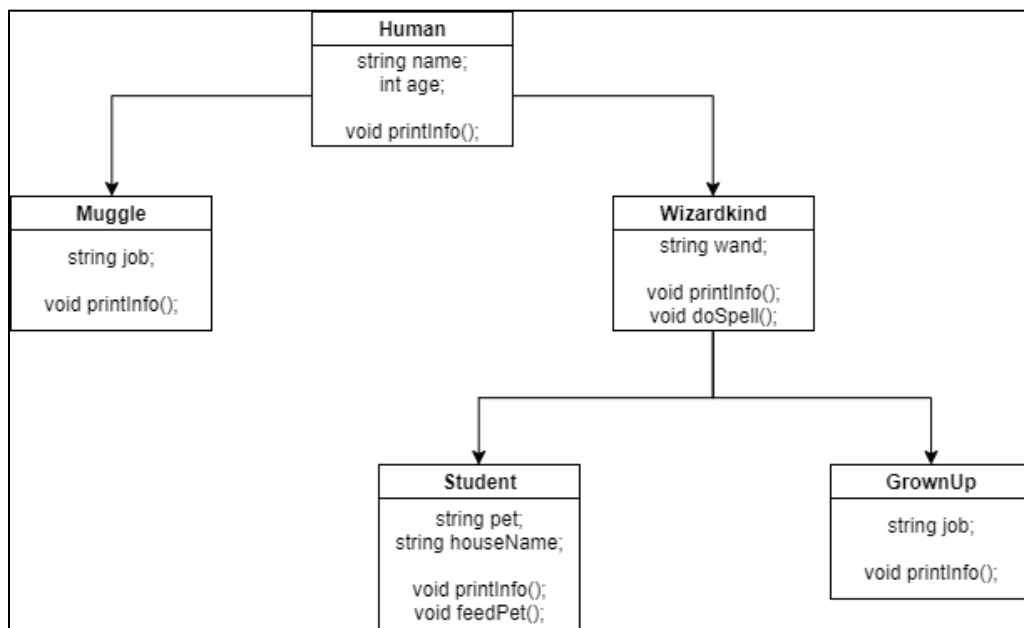
You HAVE TO write down the code on your own.

You CANNOT HELP any friend while coding.

Plagiarism will not be tolerated!

Introduction

In this homework, you are asked to implement a simple program using inheritance and polymorphism as described in the lectures. In this homework, you will be writing a base class called **Human** which is inherited by two classes: **Muggle**, **Wizardkind**. And **Wizardkind** class is also inherited by two other classes called: **Student**, **GrownUp**. The derived classes inherit the functions of their bases classes and also add new member variables and functions. The functions and variables the classes have is illustrated in the following diagram.



Class Definitions

- 1- Human
 - a. Has two member variables: **name** and **age**.
 - b. Has a function named: **printInfo**, which prints the name, age information of a human.
- 2- Muggle
 - a. Has a member variable: **job**.
 - b. Has **printInfo** function implemented to print **job** alongside name and age.
- 3- Wizardkind
 - a. Has a member variable: **wand**.
 - b. Has **printInfo** function implemented to print **wand** alongside name and age.
 - c. Has another function, called **doSpell** which takes a string parameter which is the spell to cast.
- 4- Student
 - a. Has two more member variables: **pet**, **houseName**.
 - b. Has **printInfo** function implemented to print **pet** and **houseName** alongside name, age and wand.
 - c. Has another function called **feedPet**, which prints a corresponding message.
 - i. Examlpe: "Harry Potter fed Hedwig."
- 5- GrownUp
 - a. Has one additional member variable: **job**.
 - b. Has **printInfo** function implemented to print **job** alongside name, age and wand.

Sample Main

You can use the main function provided below to test your program and compare the output of yours' with the given output.

```
int main(){
    Muggle vernon("Vernon Dursley", 50, "Director at Grunnings");
    vernon.printInfo();

    cout << endl;
    Student Harry("Harry Potter", 16, "Phoenix Feather", "Hedwig", "Gryffindor");
    Harry.printInfo();

    cout << endl;
    GrownUp Dumbledore("Albus Dumbledore", 110, "Elder Wand", "Headmaster");
    Dumbledore.printInfo();

    cout << endl;
    Dumbledore.doSpell("Expecto Patronum");

    cout << endl;
    Harry.doSpell("Expelliarmus");

    cout << endl;
    Harry.feedPet();

    cout << endl;
```

```

Student Ginny("Ginny Weasley", 15, "Yew wood", "Arnold", "Gryffindor");
GrownUp Snape("Severus Snape", 35, "Dragon Heartstring", "Potion Master");
Student Hermione("Hermione Granger", 16, "Vine", "Crookshanks", "Gryffindor");

Human hArray[6];

hArray[0] = vernon;
hArray[1] = Harry;
hArray[2] = Dumbledore;
hArray[3] = Ginny;
hArray[4] = Snape;
hArray[5] = Hermione;

Human * hPtr;
for (int i = 0; i < 6; i++) {
    cout << endl;
    hPtr = &hArray[i];
    hPtr->printInfo();
}

cout << endl;
system("pause");

return 0;
}

```

Output

Muggle Informations

Name: Vernon Dursley

Age: 50

Job: Director at Grunnings

Student Wizardkind Informations

Name: Harry Potter

Age: 16

House: Gryffindor

Wand: Phoenix Feather

Pet: Hedwig

Grownup Wizardkind Informations

Name: Albus Dumbledore

Age: 110

Wand: Elder Wand

Job: Headmaster

Albus Dumbledore used spell: Expecto Patronum

Harry Potter used spell: Expelliarmus

Harry Potter fed Hedwig

Human informations

Name: Vernon Dursley
Age: 50

Human informations
Name: Harry Potter
Age: 16

Human informations
Name: Albus Dumbledore
Age: 110

Human informations
Name: Ginny Weasley
Age: 15

Human informations
Name: Severus Snape
Age: 35

Human informations
Name: Hermione Granger
Age: 16

Press any key to continue . . .

Some Important Rules

In order to get a full credit, your programs must be efficient and well presented, presence of any redundant computation or bad indentation, or missing, irrelevant comments are going to decrease your grades. You also have to use understandable identifier names, informative introduction and prompts. Modularity is also important; you have to use functions wherever needed and appropriate.

When we grade your homeworks we pay attention to these issues. Moreover, in order to observe the real performance of your codes, we are going to run your programs in *Release* mode and **we may test your programs with very large test cases**.

What and where to submit (PLEASE READ, IMPORTANT)

You should prepare (or at least test) your program using MS Visual Studio 2012 C++. We will use the standard C++ compiler and libraries of the abovementioned platform while testing your homework. It'd be a good idea to write your name and last name in the program (as a comment line of course).

Submissions guidelines are below. Some parts of the grading process are automatic. Students are expected to strictly follow these guidelines in order to have a smooth grading process. If you do not follow these guidelines, depending on the severity of the problem created during the grading process, 5 or more penalty points are to be deducted from the grade.

Name your cpp file that contains your program as follows:

"SUCourseUserName_YourLastname_YourName_HWnumber.cpp"

Your SUCourse user name is actually your SUNet username that is used for checking sabanciuniv e-mails. Do NOT use any spaces, non-ASCII and Turkish characters in the file name. For example, if your SUCourse user name is valent, name is Valentina, and last name is Tereşkova, then the file name must be:

Valent_Tereskova_Valentina_hw1.cpp

Do not add any other character or phrase to the file name. Make sure that this file is the latest version of your homework program. Compress this cpp file using WINZIP or WINRAR programs. Please use "zip" compression. "rar" or another compression mechanism is NOT allowed. Our homework processing system works only with zip files. Therefore, make sure that the resulting compressed file has a zip extension. Check that your compressed file opens up correctly and it contains your cpp file.

You will receive no credits if your compressed zip file does not expand or it does not contain the correct file. The naming convention of the zip file is the same as the cpp file (except the extension of the file of course). The name of the zip file should be as follows:

SUCourseUserName_YourLastname_YourName_HWnumber.zip

For example zubzipler_Zipleroglu_Zubeyir_hw1.zip is a valid name, but

hw1_hoz_HasanOz.zip, HasanOzHoz.zip

are **NOT** valid names.

Submit via SUCourse ONLY! You will receive no credits if you submit by other means (e-mail, paper, etc.).

Successful submission is one of the requirements of the homework. If, for some reason, you cannot successfully submit your homework and we cannot grade it, your grade will be 0.

Good Luck!

CS204 Team (M. Yusa Erguven, Kamer Kaya)