

**Test the program** by running it and issuing three newanimal commands followed by three query commands. Each new animal should involve a different animal type (cow, bird, snake), each with a different name. Each query should involve a different animal and a different property of the animal (eat, move, speak). **Give 2 points** for each query for which the program provides the correct response.

**Examine the code.** If the code contains an interface type called **Animal**, which is a struct containing three fields, all of which are strings, then **give another 2 points**. If the program contains three types – Cow, Bird, and Snake – which all satisfy the Animal interface, **give another 2 points**.

Протестируйте программу, запустив ее и выдав три команды newanimal, а затем три команды запроса. Каждое новое животное должно быть разного типа (корова, птица, змея), каждое с разным именем. Каждый запрос должен включать разное животное и разное свойство животного (есть, двигаться, говорить). Дайте 2 балла за каждый запрос, на который программа дает правильный ответ.

Изучите код. Если программа содержит интерфейсный тип Animal, который представляет собой структуру, содержащую три поля, все из которых являются строками, то поставьте еще 2 балла. Если программа содержит три типа - корова, птица и змея - которые все удовлетворяют интерфейсу Animal, поставьте еще 2 балла.

**Write a program** which allows the user to create a set of animals and to get information about those animals. Each animal has a name and can be either a cow, bird, or snake. With each command, the user can either create a new animal of one of the three types, or the user can request information about an animal that he/she has already created. Each animal has a unique name, defined by the user. Note that the user can define animals of a chosen type, but the types of animals are restricted to either cow, bird, or snake. The following table contains the three types of animals and their associated data.

Animal	Food eaten	Locomtion method	Spoken sound
cow	grass	walk	moo
bird	worms	fly	peep
snake	mice	slither	hsss

Your program should present the user with a prompt, “>”, to indicate that the user can type a request. Your program should accept one command at a time from the user, print out a response, and print out a new prompt on a new line. Your program should continue in this loop forever. Every command from the user must be either a “newanimal” command or a “query” command.

Each “newanimal” command must be a single line containing three strings. The first string is “newanimal”. The second string is an arbitrary string which will be the name of the new animal. The third string is the type of the new animal, either “cow”, “bird”, or “snake”. Your program should process each newanimal command by creating the new animal and printing “Created it!” on the screen.

Each “query” command must be a single line containing 3 strings. The first string is “query”. The second string is the name of the animal. The third string is the name of the information requested about the animal, either “eat”, “move”, or “speak”. Your program should process each query command by printing out the requested data.

Define an interface type called **Animal** which describes the methods of an animal. Specifically, the **Animal** interface should contain the methods **Eat()**, **Move()**, and **Speak()**, which take no arguments and return no values. The **Eat()** method should print the animal's food, the **Move()** method should print the animal's locomotion, and the **Speak()** method should print the animal's spoken sound. Define three types **Cow**, **Bird**, and **Snake**. For each of these three types, define methods **Eat()**, **Move()**, and **Speak()** so that the types **Cow**, **Bird**, and **Snake** all satisfy the **Animal** interface. When the user creates an animal, create an object of the appropriate type. Your program should call the appropriate method when the user issues a query command.

Напишите программу, которая позволяет пользователю создать набор животных и получить информацию об этих животных. Каждое животное имеет имя и может быть либо коровой, либо птицей, либо змеей. С помощью каждой команды пользователь может либо создать новое животное одного из трех типов, либо запросить информацию о животном, которое он уже создал. Каждое животное имеет уникальное имя, определяемое пользователем. Обратите внимание, что пользователь может определить животных выбранного типа, но типы животных ограничены либо коровой, либо птицей, либо змеей. В следующей таблице представлены три типа животных и связанные с ними данные.

Ваша программа должна выдать пользователю приглашение ">", чтобы показать, что пользователь может ввести запрос. Ваша программа должна принимать от пользователя по одной команде за раз, печатать ответ и выводить новый запрос на новой строке. Ваша программа должна продолжать этот цикл бесконечно. Каждая команда от пользователя должна быть либо командой "newanimal", либо командой "query".

Каждая команда "newanimal" должна быть одной строкой, содержащей три строки. Первая строка - это "newanimal". Вторая строка - это произвольная строка, которая будет именем нового животного. Третья строка - это тип нового животного, либо "корова", либо "птица", либо "змея". Ваша программа должна обрабатывать каждую команду newanimal, создавая новое животное и выводя на экран сообщение "Created it!".

Каждая команда "query" должна быть одной строкой, содержащей 3 строки. Первая строка - это "query". Вторая строка - имя животного. Третья строка - название запрашиваемой информации о животном: "есть", "двигаться" или "говорить". Ваша программа должна обрабатывать каждую команду запроса, распечатывая запрашиваемые данные.

Определите тип интерфейса Animal, который описывает методы животного. В частности, интерфейс Animal должен содержать методы Eat(), Move() и Speak(), которые не принимают никаких аргументов и не возвращают никаких значений. Метод Eat() должен печатать пищу животного, метод Move() должен печатать локомоцию животного, а метод Speak() должен печатать звук, произносимый животным. Определите три типа Корова, Птица и Змея. Для каждого из этих трех типов определите методы Eat(), Move() и Speak() так, чтобы типы Cow, Bird и Snake удовлетворяли интерфейсу Animal. Когда пользователь создает животное, создайте объект соответствующего типа. Ваша программа должна вызывать соответствующий метод, когда пользователь выдает команду запроса.