

Структури

Завдання

1. Номер телефону, наприклад (212) 767-8900, можна умовно розділити на три частини: код міста (212), номер телефонної станції (767) і номер абонента (8900). Напишіть програму з використанням структури `phone`, за допомогою якої можна окремо зберігати ці три частини телефонного номера. Створіть дві структурні змінні типу `phone`. Ініціалізацію однієї з них зробіть статичною, а значення для іншої запросіть з клавіатури. Потім виведіть вміст обох змінних на екран.

2. Сума двох точок визначається як точка, яка має координати, рівні сумі відповідних координат доданків. Напишіть програму, що використовує для інтерпретації точки на площині структуру з назвою `point`. Визначте три змінні типу `point`, і дві з них ініціалізуються за допомогою значень, що вводяться з клавіатури. Потім надайте третій змінній значення суми перших двох змінних і виведіть результат на екран.

3. Створіть структуру з ім'ям `Volume`, що містить три поля типу `Distance` з лекції, для зберігання трьох вимірів приміщення. Визначте змінну типу `Volume`, ініціалізуйте її, обчисліть об'єм приміщення, і виведіть результат на екран.

4. Створіть структуру з ім'ям `employee`, що містить два поля: номер співробітника типу `int` і величину його заробітної плати типу `double`. Запросіть з клавіатури дані про трьох співробітників, збережіть їх у трьох структурних змінних типу `employee` і виведіть дані про кожного із співробітників на екран.

5. Створіть структуру типу `date`, яка містить три поля типу `int`: місяць, день і рік. Попросіть користувача ввести день, місяць і рік у форматі 31/12/2002, збережіть введені значення в структурній змінній, а потім отримайте дані з цієї змінної і виведіть їх на екран у тому ж форматі, в якому вони вводилися.

6. Визначте перелічуваний тип даних `etype`, що відображає посаду співробітника:

```
enum etype {laborer, secretary, manager, accountant, executive, researcher};
```

Напишіть програму, яка спочатку по першій букві посади, яку вводить користувач, визначає відповідне значення змінної, та записує це значення в змінну типу `etype`, а потім виводить повністю назву посади, першу букву якої ввів користувач.

7. Додайте поля типу `enum etype` (див. завд. 6) і `struct date`

(див. завд. 5) у структуру `employee` з завдання 4. Організуйте програму таким чином, щоб користувач вводив 4 пункти даних про кожного з трьох працівників: його номер, величину зарплати, його посаду та дату прийняття на роботу. Програма повинна зберігати введені значення у трьох змінних типу `employee` і виводити їх вміст на екран.

8. Написати програму для виконання арифметичних дій з дробами. Значення дробів потрібно зберігати в структурі `fraction`, що складається з двох полів типу `int`, для зберігання чисельника і знаменника. Всі значення дробів та результати обчислень повинні зберігатися в змінних типу `fraction`.

9. Створіть структуру з ім'ям `time`. Три її поля, що мають тип `int`, повинні називатись `hours`, `minutes` та `seconds`. Напишіть програму, яка просить користувача ввести час у форматі години, хвилини, секунди. Програма повинна зберігати час у структурній змінній типу `time` і виводити загальну кількість секунд.

10. Створіть структуру з ім'ям `sterling`, що зберігає грошові суми в старій англійській системі. Поля структури можуть бути названі `pounds`, `shillings` і `pence` і мати тип `int`. Описати основні операції для роботи з грошовими одиницями, знаходження суми різниці, множення на число, знаходження відсотків від вказаної суми.

11. Використовуючи структуру `time` завдання 9, напишіть програму, яка отримує від користувача два значення часу в форматі 12:59:59, зберігає їх у змінних типу `struct time`, потім переводить обидва значення в секунди, складає їх, переводить суму в вихідний формат, зберігає його в змінній типу `time` і виводить отриманий результат на екран у форматі 12:59:59.

12. Створіть програму зі структурою `player`. Структура повинна містити поля `level` – для збереження рівня гравця (початкове значення 1), `XP` – кількість досвіду (початкове значення 0), `HP` – кількість очок життя (початкове значення 50). Програма повинна циклічно обирати довільне число з діапазону $[1; 3]$ і виводити запит про введення числа користувачем. Якщо число, яке ввів користувач дорівнює числу згенерованому програмою то кількість досвіду зростає на 5, в іншому випадку значення змінної `XP` зменшується на абсолютне значення різниці згенерованого і введенного числа. Кількість досвіду яка потрібна для переходу на наступний рівень дорівнює $5+n^2$ де n – поточне значення поля `level`. Гра закінчується коли значення поля `HP` стане 0, або користувач введе 0. Після кожного введення користувачу повинно виводитись дані про поточний рівень, кількість досвіду яка потрібна для переходу на наступний рівень.

Приклад роботи програми:

Let's play:

Початкове значення кількості очок HP дорівнює 50 #

HP: #####

Level#1; XP: 0/6; ~Enter number (1..3): 3

Користувач вгадав задумане число 3

Кількість досвіду збільшилась на 5 (5/6)

HP: #####

Level#1; XP: 5/6; ~Enter number (1..3): 1

Користувач не вгадав задумане число 3, а ввів 1. Кількість очок зменшилась на (3-1)=2

HP: #####

Level#1; XP: 5/6; ~Enter number (1..3): 1

Користувач вгадав задумане число 1.

Кількість досвіду збільшилась на 5 (5+5=10, 10>=6 відбувся перехід на наступний рівень, 10-6=4)

Level UP!

HP: #####

Level#2; XP: 4/9; ~Enter number (1..3): 2

Користувач не вгадав задумане число 1, а ввів 2. Кількість очок зменшилась на (2-1)=1

HP: #####

Level#2; XP: 4/9; ~Enter number (1..3): 1

Користувач вгадав задумане число 1.

Кількість досвіду збільшилась на 5 (4+5=9, 9>9 відбувся перехід на наступний рівень, 9-9=0)

Level UP!

HP: #####

Level#3; XP: 0/14; ~Enter number (1..3): 3

Користувач не вгадав задумане число 1, а ввів 3. Кількість очок зменшилась на (3-1)=2

HP: #####

Level#3; XP: 0/14; ~Enter number (1..3): 2

Користувач вгадав задумане число 2

Кількість досвіду збільшилась на 5 (5/14)

HP: #####

Level#3; XP: 5/14; ~Enter number (1..3): 1

Користувач вгадав задумане число 1

Кількість досвіду збільшилась на 5 (10/14)

HP: #####

Level#3; XP: 5/14; ~Enter number (1..3): 2

Користувач вгадав задумане число 2

Кількість досвіду збільшилась на 5 (5+10=15, 15>=14 відбувся перехід на наступний рівень, 15-14=1)

Level UP!

HP: #####

Level#4; XP: 1/21; ~Enter number (1..3): _