

**ТЕМА: ВВЕДЕНИЕ В ЯЗЫК
ПРОГРАММИРОВАНИЯ «C++»****Домашнее задание 2****ЗАДАНИЕ 1**

Пользователь вводит с клавиатуры время в секундах. Необходимо написать программу, которая переведет введенные пользователем секунды в часы, минуты, секунды и выводит их на экран.

Подсказка 1

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 1**ЗАДАНИЕ 2**

Написать программу, которая преобразует введенное с клавиатуры дробное число в денежный формат. Например, число 12,5 должно быть преобразовано к виду — 12 долларов и 50 центов.

Подсказка 2

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 2**ЗАДАНИЕ 3**

Написать программу, которая преобразует введенное пользователем количество дней в количество полных недель и оставшихся дней. Например, пользователь ввел 17 дней, программа должна вывести на экран 2 недели и 3 дня.

Подсказка 3

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 3

ЗАДАНИЕ 4

Написать программу, вычисляющую, с какой скоростью бегун пробежал дистанцию. Рекомендуемый вид экрана во время выполнения программы приведен ниже:

```
Calculating running speed.
Enter the length of distance(meters) = 1000
Enter time(min.sec) = 3.25
Distance: 1000 м
Time: 3 min 25 sec = 205 seconds
You were running at speed 17.56 km/h
```

ИЛИ

```
Вычисление скорости бега
Введите длину дистанции (метров) = 1000
Введите время (мин.сек) = 3.25
Дистанция: 1000 м
Время: 3 мин 25 сек = 205 сек
Вы бежали со скоростью 17.56 км/ч
```

Подсказка 4

Если у Вас возникли сложности с выполнением данного задания, нажмите кнопку «Подсказка». С полным решением задания вы сможете ознакомиться, нажав на кнопку «Решение»

Решение 4

ЗАДАНИЕ 5

Написать программу, которая предлагает пользователю ввести сумму денежного вклада в евро, а также процент годовых, которые выплачивает банк. Определить сумму денег, выплачиваемых банком вкладчику каждый месяц.

Финальное задание рассчитано на самостоятельное решение. Здесь не будет подсказок и готового алгоритма. Вам необходимо применить все практические навыки, полученные из предыдущих заданий.

ПОДСКАЗКА К ЗАДАНИЮ 1

1. Сколько переменных нам понадобится для этого задания?
2. Какой тип данных лучше подойдет для сохранения секунд, минут и часов? Сколько секунд в 1 минуте? Сколько минут в 1 часу?
3. Какой командой можно осуществить ввод данных? Какой командой можно осуществить вывод данных?

ПОДСКАЗКА К ЗАДАНИЮ 2

1. Сколько переменных нам понадобится для этого задания?
2. Какой тип данных способен сохранить дробную часть?
3. Какой командой можно осуществить ввод данных? Какой командой можно осуществить вывод данных?

ПОДСКАЗКА К ЗАДАНИЮ 3

1. Сколько переменных нам понадобится для этого задания?
2. Какой тип данных лучше подойдет для хранения количества дней? Количество дней может быть не целым значением?
3. Сколько дней в одной неделе? Как узнать остаток от деления?
4. Какой командой можно осуществить ввод данных? Какой командой можно осуществить вывод данных?

ПОДСКАЗКА К ЗАДАНИЮ 4

1. Сколько переменных нам понадобится для этого задания?
2. Какую информацию вводит пользователь, а какую мы должны вычислить самостоятельно? Зная дистанцию и время, как вычислить скорость бегуна?
3. Как перевести м/с в км/ч?
4. Какой командой можно осуществить ввод данных? Какой командой можно осуществить вывод данных?

РЕШЕНИЕ ЗАДАНИЯ 1

Описание решения

Нам необходимо заранее создать несколько переменных для хранения времени. Отдельно секунды, минуты и часы. После этого предложить пользователю ввести секунды.

Для ввода данных необходимо использовать команду `cin`, которая достает данные из буфера консоли и помещает их в указанную переменную.

Синтаксис:

```
cin >> название_переменной;
```

Однако при этом не стоит забывать, что перед вводом данных лучше всего показать на экран информацию о том, что необходимо пользователю вводить. Для этого можно использовать команда `cout`, которая позволяет показать на экран любые данные.

Синтаксис:

```
cout << данные;
```

Если данные у нас находятся в текстовом виде, то текст всегда необходимо помещать внутрь двойных кавычек «текст».

После того как мы получили количество секунд их необходимо перевести в минуты (`minutes = seconds / 60`) и как только мы узнаем количество минут их необходимо перевести в часы (`hours = minutes / 60`). Также не забываем вычислить остаток из наших переменных при переводе.

Решение

1. Необходимо создать переменные для секунд, минут и часов.

```
#include <iostream>
using namespace std;

int main()
{
    int seconds, minutes, hours;

    return 0;
}
```

2. Предоставляем возможность пользователю ввести данные.

```
#include <iostream>
using namespace std;

int main()
{
    int seconds, minutes, hours;

    cout << "seconds: ";
    cin >> seconds;

    return 0;
}
```

3. Высчитываем количество минут из введенных данных.

```
#include <iostream>
using namespace std;
```



```
int main()
{
    int seconds, minutes, hours;
    cout << "seconds : ";
    cin >> seconds;

    minutes = seconds / 60;
    return 0;
}
```

4. После перевода секунд в минуты, высчитываем остаток секунд, который не перевелся в минуты.

```
#include <iostream>
using namespace std;

int main()
{
    int seconds, minutes, hours;
    cout << "seconds : ";
    cin >> seconds;

    minutes = seconds / 60;
    seconds = seconds % 60;
    return 0;
}
```

5. По принципу предыдущих шагов также переводим минуты в часы и находим остаток.

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int seconds, minutes, hours;

    cout << "seconds : ";
    cin >> seconds;

    minutes = seconds / 60;
    seconds = seconds % 60;
    hours = minutes / 60;
    minutes = minutes % 60;

    return 0;
}
```

6. Отображаем результат.

```
#include <iostream>
using namespace std;

int main()
{
    int seconds, minutes, hours;
    cout << "seconds : ";
    cin >> seconds;

    minutes = seconds / 60;
    seconds = seconds % 60;
    hours = minutes / 60;
    minutes = minutes % 60;

    cout << "hours    : " << hours << "\n";
}
```

```
cout << "minutes : " << minutes << "\n";  
cout << "seconds : " << seconds << "\n";  
  
return 0;  
}
```

РЕШЕНИЕ ЗАДАНИЯ 2

Описание решения

Нам необходимо заранее создать несколько переменных для хранения самого числа, долларов и центов. После этого предлагаем пользователю ввести число.

Для ввода данных необходимо использовать команду `cin`, которая достает данные из буфера консоли и помещает их в указанную переменную.

Синтаксис:

```
cin >> название_переменной;
```

Однако при этом не стоит забывать, что перед вводом данных лучше всего показать на экран информацию о том, что необходимо пользователю вводить. Для этого можно использовать команда `cout`, которая позволяет показать на экран любые данные.

Синтаксис:

```
cout << данные;
```

Если данные у нас находятся в текстовом виде, то текст всегда необходимо помещать внутрь двойных кавычек «текст».

После того как мы получили число — мы выделяем из него целую часть и таким образом получаем целое количество долларов, которые нам необходимо было вычислить. Далее проводим операцию вычитания между числом и долларами, получая в качестве результата дробную часть, которая будет является в последствии количеством центов. Так как нам необходимо отобразить центы в виде целого числа, а не

дробного — умножаем это значение на 100 ($\text{cent} = (\text{number} - \text{dollar}) * 100$).

Решение

1. Необходимо создать переменные для числа, которое введет пользователь, а также для долларов и центов.

```
#include <iostream>
using namespace std;

int main()
{
    double number;
    int dollar, cent;

    return 0;
}
```

2. Предоставляем возможность пользователю ввести данные.

```
#include <iostream>
using namespace std;

int main()
{
    double number;
    int dollar, cent;
    cout << "number : ";
    cin >> number;

    return 0;
}
```

3. Узнаем количество гривен путем присвоения дробного числа в переменную целого типа.

```
#include <iostream>
using namespace std;

int main()
{
    double number;
    int dollar, cent;

    cout << "number : ";
    cin >> number;

    dollar = number;
    return 0;
}
```

4. Высчитываем количество копеек и переводим их в нужный нам формат путем умножения на 100.

```
#include <iostream>
using namespace std;

int main()
{
    double number;
    int dollar, cent;

    cout << "number : ";
    cin >> number;

    dollar = number;
```

```
cent = (number - dollar) * 100;

return 0;
}
```

5. Отображаем результат.

```
#include <iostream>
using namespace std;

int main()
{
    double number;
    int dollar, cent;

    cout << "number : ";
    cin >> number;

    dollar = number;
    cent = (number - dollar) * 100;

    return 0;
}
```

РЕШЕНИЕ ЗАДАНИЯ 3

Описание решения

Нам необходимо заранее создать переменные для хранения общего количества дней, а также недель. Предоставляем пользователю возможность ввести количество дней.

Для ввода данных необходимо использовать команду `cin`, которая достает данные из буфера консоли и помещает их в указанную переменную.

Синтаксис:

```
cin >> название_переменной;
```

Однако при этом не стоит забывать, что перед вводом данных лучше всего показать на экран информацию о том, что необходимо пользователю вводить. Для этого можно использовать команда `cout`, которая позволяет показать на экран любые данные.

Синтаксис:

```
cout << данные;
```

Если данные у нас находятся в текстовом виде, то текст всегда необходимо помещать внутрь двойных кавычек «текст».

После того как мы узнали количество дней — нам необходимо вычислить целое количество недель. Для этого мы можем поделить дни на 7 (количество дней в неделе) (`weeks = days / 7`). После этого не забываем вычислить остаток при переводе дней в недели (`days = days % 7`).

Решение

1. Создаем переменные для дней и недель.

```
#include <iostream>
using namespace std;

int main()
{
    int days, weeks;

    return 0;
}
```

2. Предоставляем пользователю возможность ввести количество дней.

```
#include <iostream>
using namespace std;

int main()
{
    int days, weeks;
    cout << "days : ";
    cin >> days;

    return 0;
}
```

3. Вычисляем количество недель.

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int days, weeks;
    cout << "days : ";
    cin >> days;

    weeks = days / 7;

    return 0;
}
```

4. Вычисляем остаток дней.

```
#include <iostream>
using namespace std;

int main()
{
    int days, weeks;
    cout << "days : ";
    cin >> days;

    weeks = days / 7;
    days = days % 7;

    return 0;
}
```

5. Отображаем результат.

```
#include <iostream>
using namespace std;
```

```
int main()
{
    int days, weeks;

    cout << "days : ";
    cin >> days;

    weeks = days / 7;
    days = days % 7;

    cout << "weeks : " << weeks << "\n";
    cout << "days : " << days << "\n";

    return 0;
}
```

РЕШЕНИЕ ЗАДАНИЯ 4

Описание решения

Нам необходимо заранее создать несколько переменных для хранения дистанции, времени, скорости. А также отдельно создать переменные целого типа для хранения минут и секунд. Предоставить возможность пользователю ввести дистанцию и общее время.

Для ввода данных необходимо использовать команду `cin`, которая достает данные из буфера консоли и помещает их в указанную переменную. Синтаксис:

```
cin >> название_переменной;
```

Однако при этом не стоит забывать, что перед вводом данных лучше всего показать на экран информацию о том, что необходимо пользователю вводить. Для этого можно использовать команду `cout`, которая позволяет показать на экран любые данные. Синтаксис:

```
cout << данные;
```

Из введенного дробного времени необходимо вычислить количество минут и секунд. Из получившихся значений мы легко узнаем общее количество секунд.

Далее узнаем скорость бегуна, для этого разделим дистанцию на время ($\text{speed} = \text{distance} / \text{time}$) и получим скорость в м/с. Переводим скорость в км/ч, как требуется в задании, для этого умножаем скорость на 3.6 ($\text{speed} = \text{speed} * 3.6$). Последним этапом необходимо отобразить результат по всем требованиям задания.

Решение

1. Создаем переменные для скорости, дистанции и времени, а также создадим для удобства переменные для минут и отдельно секунд.

```
#include <iostream>
using namespace std;

int main()
{
    double speed, distance, time;
    int minutes, seconds;

    return 0;
}
```

2. Предоставляем возможность пользователю ввести дистанцию и время, а также делаем вывод на экране как показано в самом задании

```
#include <iostream>
using namespace std;

int main()
{
    double speed, distance, time;
    int minutes, seconds;

    cout << "Calculating running speed.";

    cout << " Enter the length of distance(meters) = ";
    cin >> distance;
```

```
cout << " Enter time(min.sec) = ";
cin >> time;

return 0;
}
```

3. Из введенного времени нам необходимо узнать общее количество секунд. Сначала выделяем целую часть из дроби — узнаем минуты. Далее узнаем количество секунд. Для этого переводим дробную часть из времени в целое значение путем умножение на 100. Теперь, когда у нас есть отдельно минуты и секунды — вычисляем общее количество секунд.

```
#include <iostream>
using namespace std;

int main()
{
    double speed, distance, time;
    int minutes,seconds;

    cout <<"Calculating running speed.";
    cout << " Enter the length of distance(meters) = ";
    cin >> distance;
    cout << " Enter time(min.sec) = ";
    cin >> time;

    minutes = time;
    seconds = (time - minutes) * 100;
    time = (minutes * 60) + seconds;

    return 0;
}
```

4. Узнаем скорость (м/с) по формуле $S = D / T$. Переводим м/с в км/ч умножив на 3.6.

```
#include <iostream>
using namespace std;

int main()
{
    double speed, distance, time;
    int minutes, seconds;

    cout << "Calculating running speed.";
    cout << " Enter the length of distance(meters) = ";
    cin >> distance;
    cout << " Enter time(min.sec) = ";
    cin >> time;

    minutes = time;
    seconds = (time - minutes) * 100;
    time = (minutes * 60) + seconds;
    speed = distance / time * 3.6;

    return 0;
}
```

5. Отображаем результат исходя из правил отображения указанным в самом задании.

```
#include <iostream>
using namespace std;

int main()
{
```

```
double speed, distance, time;
int minutes, seconds;

cout << "Calculating running speed.";
cout << " Enter the length of distance(meters) = ";
cin >> distance;
cout << " Enter time(min.sec) = ";
cin >> time;

minutes = time;
seconds = (time - minutes) * 100;
time = (minutes * 60) + seconds;
speed = distance / time * 3.6;

cout << "Distance: " << distance << "m\n";
cout << "Time: " << minutes << " min " <<
    seconds << " sec = " << time <<
    " seconds\n";
cout << "You were running at speed " <<
    speed << "km/h\n";

return 0;
}
```