

Программирование в среде RobotC

Занятие 7: Арифметические действия
(продолжение)

Основные арифметические операторы Си

- На прошлом занятии мы познакомились с основными арифметическими операторами языка Си и разобрали несколько примеров.
- Сегодня для закрепления материалов вам надо будет написать несколько программ на пройденную тему.

Задача 1.

- Напишите программу переводящую угол из градусной меры в радианную.
- **Формат входных данных:** вначале программы задаем одно целое число a — угол, измеренный в градусах.
- **Формат выходных данных:** выводим на дисплей вещественное число R — угол, измеренный в радианах. Точность — два знака после запятой.
- **Подсказка.** Для перевода из градусов в радианы можно использовать следующую формулу:

$$R = a\pi/180,$$

где a — значение угла в градусах, R — значение угла в радианах и $\pi \approx 3.1415926$.

Задача 2.

- Напишите программу, определяющую нечётные числа.
- Задаём одно целое неотрицательное число. Программа должна вывести 1, если число нечётное, и 0, если число чётное.

Задача 3.

- Напишите программу, определяющую чётные числа.
- Задаём одно целое неотрицательное число. Программа должна вывести 1, если число чётное, и 0, если число нечётное.

Задача 4.

- **Белочки и орешки - 1**

n белочек нашли k орешков и решили разделить их поровну. Напишите программу, которая определяет, сколько орешков достанется каждой белочке.

- **Формат входных данных**

Задаём два целых положительных числа n и k , каждое из которых не превышает 10000.

- **Формат выходных данных**

Выводим одно целое число – ответ на задачу.

Задача 5.

- **Белочки и орешки - 2**

n белочек нашли k орешков и решили разделить их поровну. Напишите программу, которая определяет, сколько орешков останется после того, как все белочки возьмут себе равное количество орешков.

Формат входных данных

Задаём два целых положительных числа N и K , каждое из которых не превышает 10000.

- **Формат выходных данных**

Выводим одно целое число - ответ на задачу.

Задача 6.

- **Последняя цифра**
- Напишите программу, которая определяет по заданному натуральному числу его последнюю цифру.
- **Формат входных данных**
Задаём натуральное число n , не превосходящее 10000.
- **Формат выходных данных**
Выводим одно целое число – ответ на задачу.

Задача 7.

- **Число десятков**
- Напишите программу, которая определяет по заданному целому неотрицательному числу n число десятков в нем (предпоследнюю цифру числа). Если предпоследней цифры нет, то можно считать, что число десятков равно нулю.
- **Формат входных данных**
Задаём целое положительное число n ($0 \leq n \leq 1000000$).
- **Формат выходных данных**
Выводим одно целое число - ответ на задачу.

Задача 8.

- **Следующее четное**

- Напишите программу, которая определяет следующее за данным четное число.

- **Формат входных данных**

Задаём целое положительное число n , не превышающее 10000.

- **Формат выходных данных**

Выводим одно целое число – ответ на задачу.

Задача 9.

- **Электронные часы**

Электронные часы показывают время в формате h:mm:ss (от 0:00:00 до 23:59:59), то есть сначала записывается количество часов, потом обязательно двузначное количество минут, затем обязательно двузначное количество секунд. Количество минут и секунд при необходимости дополняются до двузначного числа нулями.

С начала суток прошло n секунд. Напишите программу, которая определяет, что покажут часы.

- **Формат входных данных**

Задаём натуральное число n , не превосходящее 100000000.

- **Формат выходных данных**

Выведем ответ на задачу.

Задача 11.

- **Разность времен**

Даны значения двух моментов времени, принадлежащих одним и тем же суткам: часы, минуты и секунды для каждого из моментов времени. Известно, что второй момент времени наступил не раньше первого. Напишите программу, которая определяет, сколько секунд прошло между двумя моментами времени.

- **Формат входных данных**

Задаём три целых числа: часы, минуты, секунды, задающие первый момент времени и три целых числа, задающих второй момент времени.

- **Формат выходных данных**

Выводим число секунд между этими моментами времени.

Задача 12.

- **Улитка**

Улитка ползет по вертикальному шесту высотой h метров, поднимаясь за день на a метров, а за ночь спускаясь на b метров. Напишите программу, которая определяет, на какой день улитка доползет до вершины шеста?

- **Формат входных данных**

Задаём три целых неотрицательных числа h , a , b , причем $a > b$. Числа не превосходят 100.

- **Формат выходных данных**

Программа должна вывести одно натуральное число – ответ на задачу.

ОТВЕТЫ К ЗАДАЧАМ

1.

```
task main(){
    int a=90;
    float r, p=3.1415926;
    r=a*p/180;
    nxtDisplayTextLine(1,"%0.2f",r);
}
```
2.

```
task main(){
    int a=13, b=24, s;
    s=a%2;
    nxtDisplayTextLine(1,"a=%d s=%d",a, s);
    s=b%2;
    nxtDisplayTextLine(3,"b=%d s=%d",b,s);
}
```
3.

```
task main(){
    int a=13, b=24, s;
    s=(a+1)%2;
    nxtDisplayTextLine(1,"a=%d s=%d",a, s);
    s=(b+1)%2;
    nxtDisplayTextLine(3,"b=%d s=%d",b,s);
}
```

ОТВЕТЫ К ЗАДАЧАМ

4. `task main(){
 int n=5, k=42;
 nxtDisplayTextLine(1,"%d",k/n);
}`
5. `task main(){
 int n=5, k=42;
 nxtDisplayTextLine(1,"%d",k%n);
}`
6. `task main(){
 int n=245;
 nxtDisplayTextLine(1,"%d",n/10);
}`

Завершение 7 занятия

В ответах написаны программы не на все предложенные задачи. Свои программы на задачи без ответов вы можете выслать на электронную почту.

На этом занятие завершено.