Программирование в среде RobotC

Занятие 11: Операции отношения и логические операции

Операции отношения

- Мы уже говорили про арифметические операции и математические функции. Сегодня поговорим об операциях отношения.
- Самое простое, что умеет делать компьютер, – это сравнивать между собой числа и значения числовых выражений. Для этого и предназначены операции отношения.
- > больше
- >= больше либо равно
- < меньше
- <= меньше либо равно
- == равно
- != не равно

Операции отношения

- Каких-то пояснений здесь, видимо, не требуется. Обратите только особое внимание на операцию сравнения (равно). Поскольку один знак равно = используется для оператора присваивания, то для операции сравнения используется два знака равенства ==. Будьте внимательны и не забывайте об этом. Это частая ошибка, которую совершают начинающие программисты.
- Теперь рассмотрим пример. Давайте напишем программу и запустим её.

Операции отношения

```
task main(){
  int a=10,b=5,c=2;
  char ch='a';
  nxtDisplayTextLine(0,"%d",b>c);
  nxtDisplayTextLine(1,"%d",a==b*c);
  nxtDisplayTextLine(2,"%d",a!=b*c);
  nxtDisplayTextLine(3,"%d",a<c);
  nxtDisplayTextLine(4,"%d",b>=c+3);
  nxtDisplayTextLine(5,"%d",-1<=b);
  nxtDisplayTextLine(6,"%d",ch=='a');
  nxtDisplayTextLine(7,"%d",ch!='a');
}
```

В результате выполнения программы мы видим нули и единицы.

Единица – это истина, нуль – ложь!

Вообще, *любое число, кроме нуля, интерпретируется как истина*.

Логические операции

- Кроме операций отношения есть ещё и логические операции. Они используются для того, чтобы из простых условий создавать сложные. Но, на самом деле ничего сложного нет. Сейчас разберёмся.
- Логических операций всего три штуки:

  ! - логическое НЕ.

  && - логическое И.

  || - логическое ИЛИ.

  Разберёмся как они работают.

Логические операции

Оператор ***логическое НЕ***

- Записывается: ! (условие)
- отрицает выражение, к которому он был применён. Если условие было истиной, оно становится ложью, а если было ложью, то становится истиной. Вот пример:

```
task main(){
 int a=10,b=5,c=2;
 nxtDisplayTextLine(0,"%d",1&&1);
 nxtDisplayTextLine(1,"%d", 1&&1);
 nxtDisplayTextLine(2,"%d",a!=b*c);
 nxtDisplayTextLine(3,"%d",!(a!=b*c));
}
```

Напишите программу, запустите её. Подумайте над результатом работы программы.

Логические операции

Оператор ***логическое И***

- Записывается: (условие1) && (условие2)
- Возвращает истину, если оба условия истинны. Во всех остальных случаях возвращает ложь. Вот пример:

```
task main(){
  int a=10,b=5,c=2;
nxtDisplayTextLine(0,"%d",(b>c)&&(a==b*c));
nxtDisplayTextLine(1,"%d", (b<c)&&(a==b*c));
nxtDisplayTextLine(2,"%d", (b>c)&&(a!=b*c));
nxtDisplayTextLine(3,"%d", (b<c)&&(a!=b*c));
nxtDisplayTextLine(4,"%d", !(b>c)&&(a==b*c));
nxtDisplayTextLine(5,"%d", !(b<c)&&!(a!=b*c));
nxtDisplayTextLine(6,"%d", !((b>c)&&(a==b*c)));
}
```

Напишите программу, запустите её. Подумайте над результатом работы программы.

Логические операции

Оператор ***логическое ИЛИ***

- Записывается: (условие1) || (условие2)
- Возвращает истину, если хотя бы одно из условий истина или оба условия сразу истинны. Иначе возвращает ложь. Пример:

```
task main(){
  int a=10,b=5,c=2;
nxtDisplayTextLine(0,"%d",(b>c)||(a==b*c));
nxtDisplayTextLine(1,"%d", (b<c) ||(a==b*c));
nxtDisplayTextLine(2,"%d", (b>c) ||(a!=b*c));
nxtDisplayTextLine(3,"%d", (b<c) ||(a!=b*c));
nxtDisplayTextLine(4,"%d", !(b>c) ||(a==b*c));
nxtDisplayTextLine(5,"%d", !(b<c) ||!(a!=b*c));
nxtDisplayTextLine(6,"%d", !((b>c) ||(a==b*c)));
}
```

Напишите программу, запустите её. Подумайте над результатом работы программы.

Приоритет логических операций

- Для логических операций тоже есть определённая очерёдность выполнения.

1. Операции в скобках
2. Логическая операция НЕ
3. Логическая операция И
4. Логическая операция ИЛИ

- Включим логические операции в общий список приоритетов операций, который мы составляли для арифметических операций и математических функций.

1. Операции в скобках
2. вычисляются функции (например, sqrt(), cos() и др.)
3. Логическая операция НЕ
4. умножение, деление, остаток от деления (слева направо)
5. сложение, вычитание
6. Логическая операция И
7. Логическая операция ИЛИ
8. выполняется присваивание

Задача 1.

- Отметьте все логические выражения, значения которых истина:

1. 2<33
2. 8%2 != 4
3. 8 > 3
4. 1 == 1
5. -1
6. 1
7. 0

Задача 2.

- Отметьте все логические выражения, значения которых ложь.

1. 0
2. 10 != 10
3. 8%2 == 0
4. 2 > 3
5. 1 == 0
6. 2< 3
7. 1

Задача 3.

- Отметьте все логические выражения, значения которых истина:

1. 0 && 0
2. 1 && 0
3. 1 && 1
4. 0 || 1
5. 1 || 0
6. 0 || 0
7. 1 || 1
8. 0 && 1

Задача 4.

- Отметьте все логические выражения, значения которых ложь:

1. 0 && 0
2. 0 && 1
3. 0 || 0
4. 1 && 0

Задача 5.

- Дано следующее логическое отношение: !(a&&b)||(!a&&b). Выберите вариант(ы), в котором(ых) значения переменных a и b такие, что значение данного логического выражения истина.

1. a = 1, b = 1
2. a = 0, b = -2
3. a = 1, b = 0
4. a = 0, b = 0
5. a = 0, b = 1

Задача 6.

- Задана следующая таблица истинности:

  | a | b | c | X |

  | 1 | 0 | 0 | 1 |

  | 0 | 0 | 0 | 0 |

  | 1 | 1 | 1 | 0 |

  Выберите логическое выражение, которое удовлетворяет данной таблице истинности.

1. X = a && !b && !c
2. X = a || !b || !c
3. X = a && b && c
4. X = !a && !b && !c

Задача 7.

- Напишите программу сравнивающие два целых числа. А именно, для заданных двух целых чисел x, y программа выдаёт:
  1 -- если x=y
  0 -- если x≠y

Задача 8*

- На координатной плоскости OXY задана точка A(x,y). Необходимо указать квадрант, в котором она расположена. Номера квадрантов:

  I – $x>0$, $y>0$;

  II – $x<0$, $y>0$;

  III – $x<0$, $y<0$;

  IV – $x>0$, $y<0$;

  Для двух вещественных чисел x и y, которые не равны нулю, написать программу, которая вычисляет одно целое число – номер квадранта.
- *Замечание*. Для решения этой задачи достаточно тех знаний, которые были изучены на этом уроке.
- *Подсказка.* Выражение $(x>0)*14$ при $x>0$ даёт значение 14, при $x\leq 0$ -- значение 0.

Ответы к задачам

1. 1, 2, 3, 4, 5, 6
2. 1, 2, 4, 5
3. 3, 4, 5, 7
4. 1, 2, 3, 4
5. 2, 3, 4, 5
6. 1
7.
```
task main(){
  int x=3, y=2;// x=4, y=4
  nxtDisplayTextLine(1,"%d",x==y);
 }
```
8.
```
task main(){
  int n;
  float  x=4, y=-9;
  n=((x>0)&&(y>0))+2*((x<0)&&(y>0))+3*((x<0)&&(y<0))+4*((x>0)&&(y<0));
  nxtDisplayTextLine(1,"%d",n);
 }
```

Завершение занятия

На этом занятие завершено.