Variables and Arrays

Chantilly Robotics (Team 612)

Variables

- Used to store data
- Have types - builtin or user-defined
- Builtin types are built into the language

Beginner Variable Types

- `int` - stores integers
- `bool` - an `int` that can be 1 (true) or 2 (false)
- `float` - stores numbers with decimal
- `double` - double the precision of floats
- `char` - stores single characters
- `std::string` - (not a builtin) stores multiple (or a *string* of) characters

Variable Type Example

Output: "05.1233.1415dnone"

```
// Example program
#include <iostream>
#include <string>

int main()
{
  std::string name = "none";
  int x = 0;
  float f = 5.123;
  double d = 3.1415;
  char c = ‘d’;
  std::cout << x << f << d << c << name;
}
```

Operators

- Operates on variables
- You can redefine existing operators

Arithmetic Operator List

Operator name	Syntax	Definition
Addition	`a + b`	Returns sum of a and b
Subtraction	`a - b`	Returns difference of a and b
Multiplication	`a * b`	Returns product of a and b
Division	`a / b`	Returns quotient of a and b
Modulus	`a % b`	Returns remainder of a and b
Increment	`++a` `a++`	Increases value of a by 1
Decrement	`--a` `a--`	Decreases value of a by 1

*variable a and variable b are both integers

Comparison Operator List

Operator name	Syntax	Definition
Is equal to	`a == b`	Returns if a is equal to b
Less than	`a < b`	Returns if a is less than b
Greater than	`a > b`	Returns if a is greater than
Less than or equal to	`a <= b`	Returns if a is less than or equal to b
Greater than or equal to	`a => b`	Returns if a is greater than or equal to b
Is not equal to	`a != b`	Returns if a is not equal to b (false if a is equal to b)

*variable a and variable b are both integers

Logical Operator List

Operator name	Syntax	Definition
Negation	`!a`	Returns opposite of a
Negation	`not a`	Alternative way to write `!a`
And	`a && b`	Returns true if a and b are both true
And	`a and b`	Alternative way to write `a && b`
Or	`a || b`	Returns true if a or b is true
Or	`a or b`	Alternative way to write `a || b`
Conditional	`a ? c1 : c2`	If `a`, then `c1`, else `c2`

*variable a and variable b are both booleans

Assignment Operator List

Operator name	Syntax	Definition
Assignment	a = b	Assigns b to a
Addition assignment	`a += b`	Stores the sum of a and b in a (a = a + b)
Subtraction assignment	`a -= b`	Stores the difference of a and b in a (a = a - b)
Multiplication assignment	`a *= b`	Stores the product of a and b in a (a = a * a)
Division assignment	`a /= b`	Stores the quotient of a and b in a (a = a / b)
Modulus assignment	`a %= b`	Stores the remainder of a / b in a (a = a % b)

*variable a and variable b are both integers

Arrays

- Lists of variables with a certain size.
- All items must be the same data type
- The place a variable is in an array is called its index
    - Indexes start at 0

Example

```
#include <iostream>
#include <string>
#include <array>

int main() {
    std::array<std::string, 3> students;
    // two ways of setting values of arrays
    students[0] = “Max”;
    students[1] = “Muhammad”;
    Students[2] = “Tiana”;
    std::array<int, 3> grades = { 70, 84, 77 };

    std::cout << “Test scores\n”
    std::cout << students[0] << “ got a “ << grades[0] << std::endl;
    std::cout << students[1] << “ got a “ << grades[1] << std::endl;
    std::cout << students[2] << “ got a “ << grades[2] << std::endl;
}
```

Homework (on repl.it)

Copy your code from last week and paste it into the new assignment.

Create an array of `int` variables called player_stats

- The first slot should be the player's level (set this to 1 for now)
- The second slot should be the player's HP (set this to 12 for now)
- The third slot should be the player's attack (set this to 0 for now)
- The fourth slot should be the player's magic stat (set this 0 for now)
- Create a variable which contains the sum of all the player's stats
- Create another variable that is half sum of the player's stats (remember how `int`s act in operations