PLP3: Data Types and Naming Conventions

**Write a piece of code that creates variable of each of these common data types and follows the naming conventions:**

• int

• string

• floating-point number

• boolean

• array/list

• hash/dictionary

1. (attached code for this part of the homework) Build/run and it will print results.

**In your code, experiment with doing different things with the data types: can you add ints and floats? If you do, is the resulting variable an int (narrowing conversion) or a float (widening conversion)? Can you put different data types in the same array or list? Can one data type be converted to another (int to float, string to int, etc)?**

1. You can add ints and floats and it will return an int with the value rounded down. In the attached code I summed age(20) and inches(9.5) with the int 29 being printed out as a result which is a narrowing conversion. You can put in different data types by creating structs in the program. One data type can be converted to another such as short to int, int to float, double to int...

Source: <http://www.cplusplus.com/doc/tutorial/typecasting/>

**Discussion questions:**

1. **What are the naming requirements for variables in your language?**

**What about naming conventions? Are they enforced by the compiler/interpreter, or are**

**they just standards in the community?**

1. The names can range from 1-255 characters. The name must be between 1 to 31 characters in order to be portable to other environments. All variable names must begin with a letter of the alphabet or an underscore( \_ ).
2. After the first letter in the variable name, both letters and numbers are allowed for naming conventions. No spaces or special characters are allowed. Uppercase characters are distinct from lowercase. Using all capital letters are used to identify constant variables and are reserved. C++ keywords cannot be used as a variable name.
3. Depending on the compiler, some enforces some C++ rules more strictly than others.

Sources: <https://mathbits.com/MathBits/CompSci/DataBasics/naming.htm>

<https://docs.oracle.com/cd/E37069_01/html/E37075/gnyjb.html>

2. **Is your language statically or dynamically typed?**

1. C++ is a statically typed language.

Source: <https://www.sitepoint.com/typing-versus-dynamic-typing/>

3. **Strongly typed or weakly typed?**

1. C++ is a weakly typed language

Source:<http://dynamicsofprogramming.blogspot.com/2014/05/strong-typing-vs-weak-typing-vs-static.html>

4. **If you put this line (or something similar) in a program and try to print x, what does it do? If it doesn't compile, why? Is there something you can do to make it compile?**

**(x = "5" + 6)**

1. After declaring x as an int and typing in the expression above the program returns “error: conversion from ‘const char\*’ to ‘int’.”
2. It is possible to make this compile by creating 5 as a separate string and using the stringstream class to convert the string to a number.

Source: <https://www.geeksforgeeks.org/converting-strings-numbers-cc/>

5. **Describe the limitations (or lack thereof) of your programming language as they relate to the coding portion of the assignment (adding ints and floats, storing different types in lists, etc). Are there other restrictions or pitfalls that the documentation mentions that you need to be aware of ?**

1. - When data points to the same thing from two different starting points, this causes a major problem and the program will have mixed up problems within the code

- C++ is used for platform specific application, for the particular operating system or platform, the library set is usually chosen that locks, when C++ program used for web applications complex and difficult to debug

- can’t support garbage collection, it does not support Dynamic Memory Allocation, it is not secure because it has pointer, friend function and global variable and it has no support for threads built in

- allows classes and thus the functions with the same name ( and overloaded functions ) thus the symbol mangling system must be used

Source:<https://www.online-sciences.com/programming/c-programming-language-advantages-and-disadvantages/>

6. **How do type conversions work in your language? Are the conversions narrowing or widening, and do they work by default or do they have to be declared by the programmer?**

1. *Implicit type conversion:* Type conversions are done by the compiler without any trigger from the user. This usually takes place when in an expression, more than one data type is present. All the data types of the variables are upgraded to the data type of the variable with the largest data type.

*Explicit type conversion:* This type conversion is called type casting and it is user defined. It can be done in two ways:

1. Converting by assignment(forceful casting) → (type) expression
2. Conversion using cast operator → unary operator forcing conversion

Source: <https://www.geeksforgeeks.org/type-conversion-in-c/>