

Introduction to computer systems CSC 115 Primitive Data Types and Variables How Computing Works? Console.WriteLine("a+b={0} sum={1} equal={2}", a+b, sum, equal); 41 Decimal Floating-Point Types ? There is a special decimal floating-point real number type in C#: ? decimal (+-1,0 x 10-28 to +-7,9 x 1028): 128-bits, precision of 28-29 digits ? Used for financial calculations ? No round-off errors ? Almost no loss of precision ? The default value of decimal type is: ? 0.0M (M is the suffix for decimal numbers) 42 The Boolean Data Type ? The Boolean data type: ? Is declared by the bool keyword ? Has two possible values: true and false ? Is useful in logical expressions ? The default value is false 43 Boolean Values – Example ? Example of boolean variables taking values of true or false: int a = 1; int b = 2; bool greaterAB = (a > b); Console.WriteLine(greaterAB); // False bool equalA1 = (a == 1); Console.WriteLine(equalA1); // True 44 The Character Data Type ? The character data type: ? Represents symbolic information ? Is declared by the char keyword ? Gives each symbol a corresponding integer code ? Has a '\0' default value ? Takes 16 bits of memory (from U+0000 to U+FFFF) 45 Characters and Codes ? The example below shows that every symbol has an its unique Unicode code: char symbol = 'a'; Console.WriteLine("The code of '{0}' is: {1}", symbol, (int) symbol); symbol = 'b'; Console.WriteLine("The code of '{0}' is: {1}", symbol, (int) symbol); symbol = 'A'; Console.WriteLine("The code of '{0}' is: {1}", symbol, (int) symbol); 46 The String Data Type ? class \_ "A" ? point9 ? Declaring Variables ? When declaring a variable we: ? Specify its type ? Specify its name (called identifier) ? May give it an initial value ? The syntax is the following: ? Example: [= ]; int height = 200; 53 Initializing Variables ? Initializing ? Is assigning of initial value ? Must be done before the variable is used! ? Several ways of initializing: ? By using the new keyword ? By using a literal expression ? By referring to an already initialized variable 54 Initialization – Examples ? Example of some initializations: // The following would assign the default // value of the int type to num: int num = new int(); // num = 0 // This is how we use a literal expression: float heightInMeters = 1.74f; // Here we use an already initialized variable: string greeting = "Hello World!"; string message = greeting; 55 Escaping Sequences ? Using a Structure Type Employee companyEmployee; companyEmployee.firstName = "Joe"; companyEmployee.age = 23; public struct Employee { public string firstName; public int age; } Thinking Corner: Ants ? An ant starts walking on a bright whose length is m meters. The ant walks with velocity v meters/second. After arriving at one end of the bridge, it walks back with the same speed. It keeps walking for t minutes. How many rounds that the ant completely crosses the bridge and gets back to the starting point? Write a program that takes m, v, and t , as double, and compute how many rounds that the ant walks over the bridge. Ideas ? From the duration and speed, we can calculate the total distance the ant walks. ? From the distance and the length of the bridge, we can calculate the number of rounds ? The problem we have to solve is that how we manage to removes all the fractions in the result by the division. ? We will start with a program that compute the number of rounds, in real, and we will fix this problem later. Partial solution ? method Main static void Main() { Console.Write("Enter m (m): "); double m = double.Parse(Console.ReadLine()); Console.Write("Enter v (m/s): "); double v = double.Parse(Console.ReadLine()); Console.Write("Enter t (min): "); double t = double.Parse(Console.ReadLine()); double dist = v \* t \* 60; (click to show) Can You Spot the Disallowed Variable Names? int.Parse(string) – parses a string to int string str =

Console.ReadLine() int number = int.Parse(str); Console.WriteLine("You entered: ", number); 15

Converting Strings to Numbers ? Numeral types have a method Parse(...) for extracting the numeral value from a string ? int.Parse(string) – string → int ? long.Parse(string) – string → long ? float.Parse(string) – string → float ? Causes FormatException in case of error string s = "123"; int i = int.Parse(s); // i = 123 long l = long.Parse(s); // l = 123L string invalid = "xxx1845"; int value = int.Parse(invalid); // FormatException 16 Reading Numbers from the Console – Example 17 static void Main() { int a = int.Parse(Console.ReadLine()); int b = int.Parse(Console.ReadLine()); Console.WriteLine(a, b, a+b); Console.WriteLine(a, b, a\*b); float f = float.Parse(Console.ReadLine()); Console.WriteLine(a, b, f, a\*b/f); } Declaring Local Variables ? Depending on the unit of measure we may use different data types: byte centuries = 20; // Usually a small number ushort years = 2000; uint days = 730480; ulong hours = 17531520; // May be a very big number Console.WriteLine("centuries is years, or days, or hours.", centuries, years, days, hours); 13 Console.ReadLine() ? Gets a line of characters ? Returns a string value ? Returns null if the end of the input is reached Console.Write("Please enter your first name: "); string firstName = Console.ReadLine(); Console.Write("Please enter your last name: "); string lastName = Console.ReadLine(); Console.WriteLine("Hello, ", firstName, lastName); 14 Reading Numeral Types ? Specifying extra formatting: Console.WriteLine("Size {0}x{1}", width, height); double salary=12000; Console.WriteLine("My salary is {0:f2}.", salary); Console.WriteLine("Hello"); Console.WriteLine(area); 25 Exercise : Find the output of the following Code ? using System; class Binary { public static void Main() { int x, y, result; float floatresult; x = 7; y = 5; result = x+y; Console.WriteLine("x+y: {0}", result); result = x-y; Console.WriteLine("x-y: {0}", result); result = x\*y; Console.WriteLine("x\*y: {0}", result); result = x/y; Console.WriteLine("x/y: {0}", result); floatresult = (float)x/(float)y; Console.WriteLine("x/y: {0}", floatresult); result = x%y; Console.WriteLine("x%y: {0}", result); result += x; Console.WriteLine("result+=x: {0}", result); } } 26 And here's the output: x+y: 12 x-y: 2 x\*y: 35 x/y: 1 x/y: 1.4 x%y: 2 ? Using the + operator string s = "Microsoft .NET Framework"; 47 Saying Hello – Example ? Concatenating the two names of a person to obtain his full name: ? NOTE: a space is missing between the two names! We have to add it manually string firstName = "Ivan"; string lastName = "Ivanov"; Console.WriteLine("Hello, {0}!\n", firstName); string fullName = firstName + " " + lastName; Console.WriteLine("Your full name is {0}.", fullName); 48 What Is a Variable? \ for single quote \ for double quote ? \ for backslash \n for new line ? \uXXXX for denoting any other Unicode symbol 56 Character Literals – Example ? Examples of different character literals: char symbol = 'a'; // An ordinary symbol symbol = '\u006F'; // Unicode symbol code in // a hexadecimal format symbol = '\u8449'; // ? (Leaf in Traditional Chinese) symbol = '\'; // Assigning the single quote symbol symbol = '\\'; // Assigning the backslash symbol symbol = '\n'; // Assigning new line symbol symbol = '\t'; // Assigning TAB symbol symbol = "a"; // Incorrect: use single quotes 57 Structure Types ? May lose precision, but not magnitude using System; class Test { static void Main( ) { int intValue = 123; long longValue = intValue; Console.WriteLine("(long) {0} = {1}", intValue, ?longValue); } } Explicit Data Type Conversion ? To do explicit conversions, use a cast expression: using System; class Test { static void Main( ) { long longValue = Int64.MaxValue; int intValue = (int) longValue; Console.WriteLine("(int) {0} = {1}", longValue, ?intValue); } } What are Floating-Point Types? struct

| User-Defined Value | Type   | Built-in Type                                  | Data Types | 9                            | Type                | Size   | Description      | Range   |
|--------------------|--|--|------------|------------------------------|---------------------|--|------------------|---------|
| bool               | 1 byte   | Store truth value true / false                 | char       | 1 byte                       | Store one character | character code                                   | 0 – 255          | byte    |
| byte               | 1 byte   | Store positive integer                         | 0 – 255    | short                        | 2 byte              | Store integer                                    | –32,768 – 32,767 | int     |
| 4 byte             | Store integer  | –2.1 x 10 <sup>9</sup> – 2.1 x 10 <sup>9</sup> | long       | 8 byte                       | Store integer       | –9.2 x 10 <sup>18</sup> – 9.2 x 10 <sup>18</sup> | double           | 16 byte |
| Store real number  | +– 5.0x10 <sup>–324</sup> – +– 1.7x10 <sup>308</sup> | string   | N/A        | Store sequence of characters | N/A                 |  |                  |         |

What are Integer Types? NOTE: The "f" suffix in the first statement! ? Real numbers are by default interpreted as double! ? One should explicitly convert them to float

```
float floatPI = 3.141592653589793238f; double doublePI = 3.141592653589793238;
```

```
Console.WriteLine("Float PI is: {0}", floatPI); Console.WriteLine("Double PI is: {0}", doublePI); 40
```

Floating-Point Calculations ? Declare five variables choosing for each of them the most appropriate of the types byte, sbyte, short, ushort, int, uint, long, ulong to represent the following values: 52130, –115, .4825932, 97, –10000.????????????????2.3.5.6.8