**CSC 1100 – Problem Solving and Programming**

**Homework 2 – rory lange**

**25 points – Due February 4, 11am**

**Late deadline is February 6, 11:59pm, but 20% off**

**a)** Save this document with your name and the homework number somewhere in the file name.

**b)** Type/paste your answers into the document.

**c)** Submit this document to the Canvas item where you downloaded this document.

**1) [5 points]** Write an algorithm to describe how to convert a user input from teaspoons to fluid ounces, cups, and quarts. Use enough detail that you have between five and ten steps. List at least one assumption.

a) Assumptions

the user enters an integer

should use a precise output because they could be cooking

b) Steps

define variables/constants

get user input

convert user input into oz

convert user input into cups

convert user input into quarts

return answers

**2) [4 points]** Using information from the Internet, list and define two terms used in Agile development. Include your sources in the form of hyperlinks.

**a)**

Term ► Agile software development

Source ► [What is Agile Software Development? | Agile Alliance](https://www.agilealliance.org/agile101/)

Definition ► an umbrella term that refers to the 12 guidelines expressed in the “Agile Manifesto”.

**b)**

Term ► Business agility

Source ► [What is Agile Software Development? | Agile Alliance](https://www.agilealliance.org/agile101/)

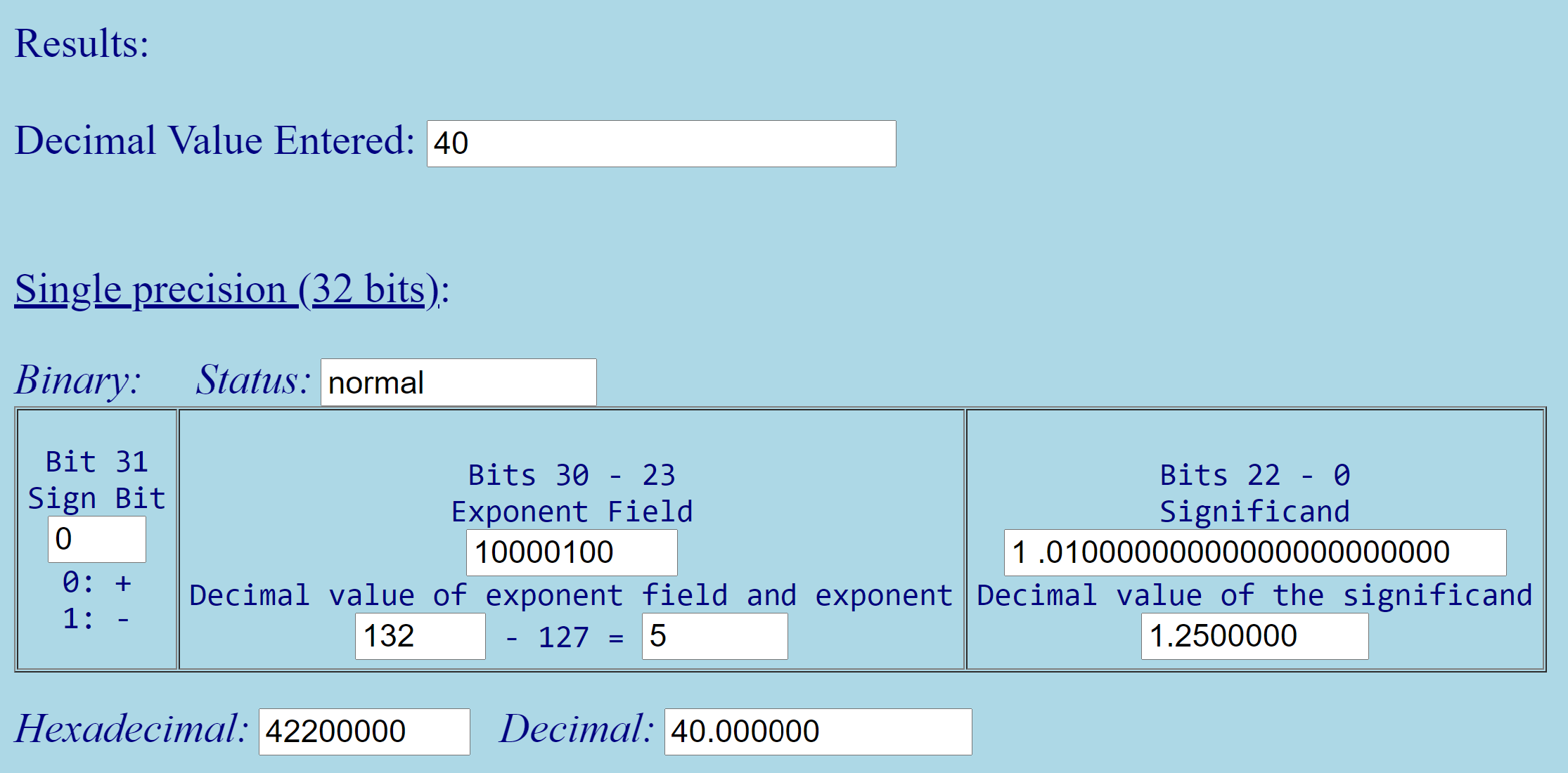
Definition ► The organized acceptance of agile software development so that businesses can react and solve problems as they come up

**3) [2 points]** Describe two techniques for making code more readable.

► Adding white space

► Adding comments

**4) [4 points]** The tool at [babbage.cs.qc.cuny.edu/IEEE-754.old/Decimal.html](https://babbage.cs.qc.cuny.edu/IEEE-754.old/Decimal.html) shows how a real number is stored in the computer. Enter four random decimal numbers less than 100 in the first column of the table below. For each number, enter the parts that represent how that number is stored as a float (32 bits) value. The first example for 40 shows that number is stored.



| Decimal value | Sign bit | Decimal exponent | Decimal significand | Hex value | Calculation |
| --- | --- | --- | --- | --- | --- |
| 40 | 0 | 5 | 1.25 | 42200000 | 2^5 \* 1.25 = 40 |
| 21 | 0 | 4 | 1.3125 | 41A80000 | 2^4 \* 1.3125 = 21 |
| 73 | 0 | 6 | 1.1406250 | 42920000 | 2^6 \* 1.1406250 = 73 |
| 3 | 0 | 1 | 1.5 | 40400000 | 2^1 \* 1.5 = 3 |
| 39 | 0 | 5 | 1.21875 | 421C0000 | 2^5 \* 1.21875 = 39 |

**5) [2 points]** Why can't many real numbers be represented precisely by a computer?

They can only be as precise as the data type allows because the computer only has limited memory

**6) [4 points]** What happens in an **assignment statement** when the given expressions are assigned to the given variables? Is casting required? Use the **assignment statement table** provided in the session notes.

**a)** A **float** expression is assigned to a **string** variable.

Implicit cast is done, it stores the float as a string instead of 6.1 its “6.1”

**b)** A **int** expression is assigned to a **double** variable.

Stores int as double implicit cast is done

**c)** A **string** expression is assigned to an **float** variable.

Value that was stored in float remains unchanged, may cause unpredictable errors.

**d)** A **char** expression is assigned to an **char** variable.

No cast is done, it stores the char

**7) [2 points]** What is the preferred method for writing multiple rows of output, like a table, to the screen? Just describe it, don't show code.

One line of code per row in the table/output.

**8) [1 point]** The most important person in the application development process is the:

► Customer

**9) [1 point]** The most important statement in any programming language is the:

► Comment