

Xilun Guo

Computation - Process of calculating an equation in order.

Central Processing Unit (CPU) – The hardware within a computer that carries out the instructions of a computer program.

Arithmetic Logic Unit (ALU) – A digital circuit that performs integer arithmetic and logical operations.

Registers – Small amount of storage that speeds up computations.

Memory – The storage of information.

Cache – Storage so that future operations can be performed faster.

Disk Drive – Place of storage.

Input – What you type into the machine.

Output – What the machine computes and gives you.

Compiler – A computer program that transforms source code written in a programming language into another computer language.

Interpreter – A computer program that directly executes. For example: performs, instructions written in a programming or scripting language, without previously compiling them into a machine language program.

High-level Language - Compiler that allows shortcuts and can use generic language to benefit the programmer.

Intermediate Language – Compiler that is limited in the amount of shortcuts available.

Machine Language – A set of instructions executed directly by a computer's central processing unit.

Integer - number without decimal.

Float - Number that can contain a decimal but is limited. To the 6th power.

Double - A larger float that can contain larger and more precise numbers. To the 15th power.

String - A sequence of characters. i.e. (" 123 " + " Jacky ")

Variable - A storage location that has an identifier and contains some information.

Constant - An identifier whose associated value cannot typically be altered by the programming during its execution.

Data Type - A classification of a type of data.

Expression - A combination of values, constants, variables, operators, and functions.

Assignment Statement - It copies the value into the variable.

Mathematical Operator - A variable that uses math to change a value.

Logical Operator - A variable that uses words to compare values.

Relational Operator - A variable that uses symbols to compare values.

Precedence - Defining something as being more important than something else.

Function - A named procedure that performs a distinct service.

- **Step 1: Problem Analysis.** (10 pts)
 - a. Comments about the problem to aid in understanding it.

Use Babylonian function to get the square of positive whole number, and compare the answer is calculated by computer.

b. Description of the knowledge base (this list would include what you would be expected to know to follow the solution).

1. The square rule in math.
2. Python language.
3. Babylonian function.

- **Step 2: Program Design.** (10 pts) List the specific steps needed to calculate the square root of a number using the Babylonian algorithm, where the user provides you the number as input. Remember, you have to be very explicit here to make sure the computer can accomplish the task using your directions.

1. Write the Babylonian function with code.
2. Ask a positive whole number from the user.
3. Use the Babylonian function to get the square of the positive whole number from the user give you as a guess answer.
4. Use computer to get the square of the positive whole number from the user give you as a right answer.
5. Print the guess answer and the right answer, and compare the difference between them.
6. Ask the user if want to play again?

- **Step 3: Program Implementation.** (20 pts) This is the Python code that calculates the square root of a number using the Babylonian algorithm, i.e. your .py file that gets saved before running your program.

- **Step 4: Program Testing.** (20 pts) Create a Test Plan with several test cases including the average and extreme cases.

1. Test small positive whole numbers.
2. Test big positive whole numbers.

```
import math
def sqrt(n, error): #define a function, Babylonian algorithm
    guess = n / 2.0;
    answer = n ** 0.5;
    while abs(answer - guess) > error:
        guess = (guess + (n / guess)) / 2;
    return guess;

user_input = 1;
while(user_input == 1):
    error = 0.000000000000001;
    n = int(input("Enter a positive number: "));
    print("The guess answer is " + str(sqrt(n, error)));
    print("The right answer is " + str(math.sqrt(n)));
    user_input = (int(input("To continue? (1 - yes, 0 - no):
"")));
```