FACULTY OF COMPUTING
& INFORMATION TECHNOLOGY

KING ABDULAZIZ UNIVERSITY

FCIT

KAU

كلية الحاسبات
وتقنية المعلومات

جامعة الملك عبدالعزيز

Sample Exam Questions

Mid-Term Exam 2 – Part 2 (Writing Code)

CPIT 110 (Problem-Solving and Programming)

Version 1.1

تنبيه!

- هذه الأسئلة عبارة عن عينة فقط توضح طريقة أسئلة اختبار الدوري الثاني - الجزء الثاني (كتابة البرامج) لمقرر البرمجة وحل المشكلات (CPIT-110).
- هذه الأسئلة لا يُعتمد عليها للمذاكرة.
- قد لا تشمل هذه الأسئلة جميع المواضيع المقررة للاختبار.
- هذه الأسئلة مناسبة للمراجعة بعد الانتهاء من مذاكرة وتطبيق المواضيع المقررة للاختبار.
- حلول الأسئلة مرفقة نهاية صفحات هذا الملف.

تنبيه!

- جميع ما سبق (Chapter 1, 2, 3) يدخل ضمن هذا الاختبار.
- لابد من مراجعة تمارين الأبواب (Chapters) السابقة.
- هذا الملف به تمارين الباب الرابع والخامس فقط ولا يشمل كل الأبواب.

Program #1

Write a program that helps a first-grader practice addition. The program should randomly generate two single-digit integers and should then ask the user for the answer. The program will then display a message stating if the answer is true or false.

```
What is 9 + 4? 12   <Enter>
9 + 4 = 12 is False
```

```
What is 3 + 1? 4   <Enter>
3 + 1 = 4 is True
```

Program #2

Write a program that prompts the user to enter an integer. If the number is a multiple of 5, the program displays the result HiFive. If the number is divisible by 2, the program displays HiEven.

```
Enter an integer: 4  <Enter>
HiEven
```

```
Enter an integer: 15  <Enter>
HiFive
```

```
Enter an integer: 30  <Enter>
HiFive
HiEven
```

Program #3

Write a program that helps a first-grader practice subtraction. The program should randomly generate two single-digit integers, number1 and number2, with number1 >= number2 and should then ask the user for the answer. The program will then display a message stating if the answer is correct. If wrong, the program should display the correct answer.

```
What is 6 - 6? 0   <Enter>
You are correct!
```

```
What is 9 - 2? 5   <Enter>
Your answer is wrong.
9 - 2 is 7
```

Program #4

Write a program that computes the Body Mass Index (BMI) for the user. Your program should prompt the user to enter a weight in pounds and height in inches. Your program should then compute and display the BMI and its interpretation for the user.

BMI	Interpretation
Below 18.5	Underweight
18.5-24.9	Normal
25.0-29.9	Overweight
Above 30.0	Obese

$$\text{BMI} = \frac{\text{Weight (kg)}}{\text{Height (m)}^2}$$

$$1\ pound = 0.45359237\ kilograms$$

$$1\ inch = 0.0254\ meters$$

```
Enter weight in pounds: 146   <Enter>
Enter height in inches: 70   <Enter>
BMI is 20.95
Normal
```

Program #5

Write a program that tests the usage of Boolean operators. Specifically, your program should prompt the user to enter one integer. Your program should then determine if the value is divisible by **2** and **3**, by **2** or **3**, or by **2** or **3** but not both.

```
Enter an integer: 18   <Enter>
18 is divisible by 2 and 3
18 is divisible by 2 or 3
```

```
Enter an integer: 15   <Enter>
15 is divisible by 2 or 3
15 is divisible by 2 or 3, but not both
```

Program #6

Write a program that determines if a given year is a leap year. Specifically, ask the user to enter a year. Then determine if that year is a leap year and display the results.

Note: A year is a leap year if it is divisible by **4** but not by **100** **or** if it is divisible by **400**.

```
Enter a year: 2008  <Enter>
2008 is a leap year? True
```

```
Enter a year: 1900  <Enter>
1900 is a leap year? False
```

```
Enter a year: 2002  <Enter>
2002 is a leap year? False
```

Program #7

Write a program to play a lottery. The program randomly generates a two-digit number, prompts the user to enter a two-digit number, and determines whether the user wins according to the following rules:

1. If the user’s input matches the lottery in the exact order, the award is $10,000.
2. If all the digits in the user’s input match all the digits in the lottery number, the award is $3,000.
3. If one digit in the user’s input matches a digit in the lottery number, the award is $1,000.

```
Enter your lottery pick (two digits): 45  <Enter>
The lottery number is 12
Sorry, no match
```

```
Enter your lottery pick (two digits): 23  <Enter>
The lottery number is 34
Match one digit: you win $1,000
```

Program #8

Write a program that generates two integers under 100 and prompts the user to enter the sum of these two integers. The program then reports true if the answer is correct, false otherwise. Also, the program displays the correct result if the answer was wrong.

```
What is 30 + 43? 50 <enter>
30 + 43 = 50 is False
The correct answer is 73
```

```
What is 30 + 43? 111 <enter>
15 + 96 = 111 is True
```

Program #9

Suppose you shop for rice and find it in two different-sized packages. You would like to write a program to compare the costs of the packages. The program prompts the user to enter the weight and price of each package and then displays the one with the better price.

Here is a sample run:

```
Enter weight and price for package 1: 50, 24.59 <enter>
Enter weight and price for package 2: 25, 11.99 <enter>

Package 2 has the better price.
```

Program #10

Write a program that plays the popular scissor-rock-paper game. (A scissor can cut a paper, a rock can knock a scissor, and a paper can wrap a rock.) The program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0, 1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws.

Here are sample runs:

```
scissor (0), rock (1), paper (2): 1 <enter>
The computer is scissor. You are rock. You won.
```

```
scissor (0), rock (1), paper (2): 2 <enter>
The computer is paper. You are paper too. It is a draw.
```

```
scissor (0), rock (1), paper (2): 2 <enter>
The computer is scissor. You are paper. You lost.
```

Program #11

Write a program that prompts the user to enter an integer for today's day of the week (Sunday is 0, Monday is 1, ..., and Saturday is 6). Also prompt the user to enter the number of days after today for a future day and display the future day of the week.

Here are sample runs:

```
Enter today's day: 5 <enter>
Enter the number of days elapsed since today: 1 <enter>
Today is Friday and the future day is Saturday
```

```
Enter today's day: 1 <enter>
Enter the number of days elapsed since today: 3 <enter>
Today is Monday and the future day is Thursday
```

Program #12

Write a program that helps a first-grader practice subtraction. The program should randomly generate two single-digit integers, number1 and number2, with number1 >= number2 and should then ask the user for the answer. The program will then display a message stating if the answer is correct. If wrong, the program should let the user repeatedly enter a new answer until it is correct.

```
What is 4 - 3? 4   <Enter>
Wrong answer. Try again. What is 4 - 3? 5   <Enter>
Wrong answer. Try again. What is 4 - 3? 1   <Enter>
You got it!
```

Program #13

Write a program that randomly generates a number between **0** and **100**, inclusive. The program will repeatedly ask the user to guess the number until the user gets the number correct. At each wrong answer, the program tells the user if the guess is too low or too high.

```
Guess a magic number between 0 and 100
Enter your guess: 50  <Enter>
Your guess is too high
Enter your guess: 25  <Enter>
Your guess is too low
Enter your guess: 42  <Enter>
Your guess is too high
Enter your guess: 39  <Enter>
Yes, the number is 39
```

Program #14

Write a program to randomly generate five subtraction questions and ask the user for the answer to each. Count how many the user got correct, and display the total time spent, by the user, answering the five questions.

```
What is 1 - 1? 0  <Enter>
You are correct!

What is 7 - 2? 5  <Enter>
You are correct!

What is 9 - 3? 4  <Enter>
Your answer is wrong.
9 - 3 is 6

What is 6 - 6? 0  <Enter>
You are correct!

What is 9 - 6? 2  <Enter>
Your answer is wrong.
9 - 6 is 3

Correct count is 3 out of 5
Test time is 10 seconds
```

Program #15

Write a program to randomly generate subtraction questions and ask the user for the answer to each and let the user decide whether to advance to the next question. Count how many the user got correct, and display the total time spent, by the user, answering the questions.

```
What is 6 - 1? 5  <Enter>
You are correct!
Enter Y to continue and N to quit: Y  <Enter>

What is 8 - 0? 6  <Enter>
Your answer is wrong.
8 - 0 should be 8
Enter Y to continue and N to quit: Y  <Enter>

What is 8 - 3? 5  <Enter>
You are correct!
Enter Y to continue and N to quit: N  <Enter>

Correct count is 2 out of 3
Test time is 24 seconds
```

Program #16

Write a program that will sum up all user inputted values. The user can keep inputting values for as long as the user wishes. If the user enters **"0"**, this means the end of the input. Your program should display the sum to the user.

```
Enter an integer (the input ends if it is 0): 2  <Enter>
Enter an integer (the input ends if it is 0): 3  <Enter>
Enter an integer (the input ends if it is 0): 4  <Enter>
Enter an integer (the input ends if it is 0): 0  <Enter>
The sum is 9
```

Program #17

Write a program that uses nested for loops to print out the **1** through **9** multiplication table.

```
 Multiplication Table
  |   1   2   3   4   5   6   7   8   9
-----------------------------------------
1 |   1   2   3   4   5   6   7   8   9
2 |   2   4   6   8  10  12  14  16  18
3 |   3   6   9  12  15  18  21  24  27
4 |   4   8  12  16  20  24  28  32  36
5 |   5  10  15  20  25  30  35  40  45
6 |   6  12  18  24  30  36  42  48  54
7 |   7  14  21  28  35  42  49  56  63
8 |   8  16  24  32  40  48  56  64  72
9 |   9  18  27  36  45  54  63  72  81
```

Program #18

Write a program to ask the user to enter two positive integers. You should then find the greatest common divisor (GCD) and print the result to the user.

```
Enter first integer: 125   <Enter>
Enter second integer: 2525   <Enter>
The greatest common divisor for 125 and 2525 is 25
```

Program #19

A university charges **$10,000** per year for study (tuition). The cost of tuition increases **7%** every year. Write a program to determine how many years until the tuition will increase to **$20,000**.

```
Tuition will be doubled in 11 years
Tuition will be $21048.52 in 11 years
```

Program #20

Write a program to display the first 50 prime numbers in five lines (so 10 numbers per line).

- Note: any integer greater than **1** is prime if it can only be divided by 1 or itself.
- Example:
  - 2, 3, 5, and 7 are prime numbers
  - 4, 6, 8, and 9 are not prime numbers

```
The first 50 prime numbers are
    2    3    5    7   11   13   17   19   23   29
   31   37   41   43   47   53   59   61   67   71
   73   79   83   89   97  101  103  107  109  113
  127  131  137  139  149  151  157  163  167  173
  179  181  191  193  197  199  211  223  227  229
```

Program #21

Write a program that reads an unspecified number of integers, determines how many positive and negative values have been read, and computes the total and average of the input values (**not counting zeros**). Your program ends with the input 0. Display the average as a floating-point number.

```
Enter an integer, the input ends if it is 0: 5  <enter>
Enter an integer, the input ends if it is 0: 8  <enter>
Enter an integer, the input ends if it is 0: -1 <enter>
Enter an integer, the input ends if it is 0: 2  <enter>
Enter an integer, the input ends if it is 0: 0  <enter>
The number of positives is 3
The number of negatives is 1
The total is 14
The average is 3.5
```

Program #22

Write a program that displays the following two tables side by side (note that 1 kilogram is 2.2 pounds and that 1 pound is 0.45 kilograms):

```
kilograms      pounds       |     pounds      kilograms
---------------------------------------------------------
1                 2.2       |     20                9.09
3                 6.6       |     25               11.36
5                11.0       |     30               13.64
7                15.4       |     35               15.91
...
193             424.6       |     500             227.27
195             429.0       |     505             229.55
197             433.4       |     510             231.82
199             437.8       |     515             234.09
```

Program #23

Write a program that reads integers, finds the largest of them, and counts its occurrences. Assume that the input ends with number 0. Suppose that you entered 3 5 2 5 5 5 0; the program finds that the largest number is 5 and the occurrence count for 5 is 4.

```
Enter a number (0: for end of input): 3 <enter>
Enter a number (0: for end of input): 5 <enter>
Enter a number (0: for end of input): 2 <enter>
Enter a number (0: for end of input): 5 <enter>
Enter a number (0: for end of input): 5 <enter>
Enter a number (0: for end of input): 5 <enter>
Enter a number (0: for end of input): 0 <enter>
The largest number is 5
The occurrence count of the largest number is 4
```

Program #24

Write a program that prompts the user to enter the number of students and each student’s score, and displays the highest score.

```
Enter the number of students: 5 <enter>
Enter a student name:  Ahmad <enter>
Enter a student score: 95.2 <enter>
Enter a student name: Omar <enter>
Enter a student score: 93.5 <enter>
Enter a student name: Jamal <enter>
Enter a student score: 95.5 <enter>
Enter a student name: Yeaser <enter>
Enter a student score: 80.5 <enter>
Enter a student name: Bander <enter>
Enter a student score: 66  <enter>


Top student Jamal's score is 95.5
```

Program #25

Write a program that prompts the user to enter the number of students and each student’s score, and displays the highest and second highest scores.

```
Enter the number of students: 5 <enter>
Enter a student name:  Ahmad <enter>
Enter a student score: 95.2 <enter>
Enter a student name: Omar <enter>
Enter a student score: 93.5 <enter>
Enter a student name: Jamal <enter>
Enter a student score: 95.5 <enter>
Enter a student name: Yeaser <enter>
Enter a student score: 80.5 <enter>
Enter a student name: Bander <enter>
Enter a student score: 66  <enter>


Top two students:
Jamal's score is 95.5
Ahmad's score is 95.2
```

Program #26

Use nested loops that display the following patterns in four separate programs:

Pattern 1:

```
1
1  2
1  2  3
1  2  3  4
1  2  3  4  5
1  2  3  4  5  6
```

Pattern 2:

```
1  2  3  4  5  6
1  2  3  4  5
1  2  3  4
1  2  3
1  2
1
```

Pattern 3:

```
               1
            2  1
         3  2  1
      4  3  2  1
   5  4  3  2  1
6  5  4  3  2  1
```

Pattern 4:

```
1  2  3  4  5  6
   1  2  3  4  5
      1  2  3  4
         1  2  3
            1  2
               1
```

Program #27

Print the following table to display the sin value and cos value of degrees from 0 to 360 with increments of 10 degrees. Round the value to keep four digits after the decimal point.

```
Degree          Sin             Cos
0               0.0000          1.0000
10              0.1736          0.9848
20              0.3420          0.9397
30              0.5000          0.8660
...
340             -0.3420         0.9397
350             -0.1736         0.9848
360             -0.0000         1.0000
```

Program #28

Write a program that displays, ten numbers per line, all the numbers from 100 to 200 that are divisible by 5 or 6, but not both. The numbers are separated by exactly one space.

```
100 102 105 108 110 114 115 125 126 130
132 135 138 140 144 145 155 156 160 162
165 168 170 174 175 185 186 190 192 195
198 200
```

Program #29

Use a while loop to find the smallest integer n such that n^2 is greater than 12,000.

```
This number is 110
```

Program #30

Use a while loop to find the largest integer n such that n^3 is less than 12,000.

```
This number is 22
```

Program #1
(Solution)

```
import random

# Generate random numbers
number1 = random.randint(0, 9)
number2 = random.randint(0, 9)

# Prompt the user to enter an answer
answer = eval(input("What is " + str(number1) + " + "
                    + str(number2) + "? "))

# Display result
print(number1, "+", number2, "=", answer,
      "is", number1 + number2 == answer)
```

Program #2 (Solution)

```
number = eval(input("Enter an integer: "))

if number % 5 == 0:
    print("HiFive")

if number % 2 == 0:
    print("HiEven")
```

Program #3
(Solution)

```
import random

# 1. Generate two random single-digit integers
number1 = random.randint(0, 9)
number2 = random.randint(0, 9)

# 2. If number1 < number2, swap number1 with number2
if number1 < number2:
    number1, number2 = number2, number1 # Simultaneous assignment

# 4. Prompt the student to answer "what is number1 - number2?“
answer = eval(input("What is " + str(number1) + " - " +
    str(number2) + "? "))

# 4. Grade the answer and display the result
if number1 - number2 == answer:
    print("You are correct!")
else:
    print("Your answer is wrong.\n", number1, "-",
        number2, "is", number1 - number2)
```

Program #4 (Solution)

```python
# Prompt the user to enter weight in pounds
weight = eval(input("Enter weight in pounds: "))

# Prompt the user to enter height in inches
height = eval(input("Enter height in inches: "))

KILOGRAMS_PER_POUND = 0.45359237  # Constant
METERS_PER_INCH = 0.0254  # Constant

# Compute BMI
weightInKilograms = weight * KILOGRAMS_PER_POUND
heightInMeters = height * METERS_PER_INCH
bmi = weightInKilograms / (heightInMeters * heightInMeters)

# Display result
print("BMI is", format(bmi, ".2f"))
if bmi < 18.5:
    print("Underweight")
elif bmi < 25:
    print("Normal")
elif bmi < 30:
    print("Overweight")
else:
    print("Obese")
```

Program #5 (Solution)

```python
# Receive an input
number = eval(input("Enter an integer: "))

if number % 2 == 0 and number % 3 == 0:
    print(number, "is divisible by 2 and 3")

if number % 2 == 0 or number % 3 == 0:
    print(number, "is divisible by 2 or 3")

if (number % 2 == 0 or number % 3 == 0) and \
       not (number % 2 == 0 and number % 3 == 0):
    print(number, "is divisible by 2 or 3, but not both")
```

Program #6 (Solution)

```
year = eval(input("Enter a year: "))

# Check if the year is a leap year
isLeapYear = (year % 4 == 0 and year % 100 != 0) or \
    (year % 400 == 0)

# Display the result
print(year, "is a leap year?", isLeapYear)
```

Program #7 (Solution)

```python
import random

# Generate a lottery
lottery = random.randint(10, 99)

# Prompt the user to enter a guess
guess = eval(input("Enter your lottery pick (two digits): "))

# Get digits from lottery
lotteryDigit1 = lottery // 10
lotteryDigit2 = lottery % 10

# Get digits from guess
guessDigit1 = guess // 10
guessDigit2 = guess % 10

print("The lottery number is", lottery)

# Check the guess
if guess == lottery:
    print("Exact match: you win $10,000")
elif (guessDigit2 == lotteryDigit1 and \
  guessDigit1 == lotteryDigit2):
    print("Match all digits: you win $3,000")
elif (guessDigit1 == lotteryDigit1
        or guessDigit1 == lotteryDigit2
        or guessDigit2 == lotteryDigit1
        or guessDigit2 == lotteryDigit2):
    print("Match one digit: you win $1,000")
else:
    print("Sorry, no match")
```

Program #8 (Solution)

```
# To use randint function we need to import random module
import random

# Generate two numbers randomly between 0 and 99 inclusive
number1 = random.randint(0, 99)
number2 = random.randint(0, 99)

# Calculate the sum of two numbers
sum = number1 + number2

# Ask the user to enter the answer
answer = eval(input("What is " + str(number1) + " + "
                    + str(number2) + "? "))

# Check the user answer (answer) with the correct answer (sum)
print(number1, "+", number2, "=", answer, "is", answer == sum)

# Display the correct answer if the answer is wrong
if sum != answer:
    print("The correct answer is ", sum)
```

Program #9 (Solution)

```
weight1, price1 = eval(input("Enter weight and price for package 1: "))
weight2, price2 = eval(input("Enter weight and price for package 2: "))

if price1 / weight1 <= price2 / weight2:
    print("Package 1 has the best price.")
else:
    print("Package 2 has the best price.")
```

Program #10 (Solution)

```
import random

# Generate scissor, rock, paper
computerNumber = random.randint(0, 2)

# Prompt the user to enter scissor, rock, or paper
userNumber = eval(input("scissor (0), rock (1), paper (2): "))

# Check the guess
if computerNumber == 0:
    if userNumber == 0:
        print("The computer is scissor. You are scissor too. It is a draw")
    elif userNumber == 1:
        print("The computer is scissor. You are rock. You won")
    elif userNumber == 2:
        print("The computer is scissor. You are paper. You lost")
elif computerNumber == 1:
    if userNumber == 0:
        print("The computer is rock. You are scissor. You lost")
    elif userNumber == 1:
        print("The computer is rock. You are rock too. It is a draw")
    elif userNumber == 2:
        print("The computer is rock. You are paper. You won")
elif computerNumber == 2:
    if userNumber == 0:
        print("The computer is paper. You are scissor. You won")
    elif userNumber == 1:
        print("The computer is paper. You are rock. You lost")
    elif userNumber == 2:
        print("The computer is paper. You are paper too. It is a draw")
```

Program #11 (Solution)

```
# Prompt the user to enter an integer for today
today = eval(input("Enter today's day: "))
elapsedDays = eval(input("Enter the number of days elapsed since today: "))

if today == 0:
  nameForToday = "Sunday"
elif today == 1:
  nameForToday = "Monday"
elif today == 2:
  nameForToday = "Tuesday"
elif today == 3:
  nameForToday = "Wednesday"
elif today == 4:
  nameForToday = "Thursday"
elif today == 5:
  nameForToday = "Friday"
else: # if today == 6:
  nameForToday = "Saturday"

futureDay = (today + elapsedDays) % 7
if futureDay == 0:
  nameForFutureDay = "Sunday"
elif futureDay == 1:
  nameForFutureDay = "Monday"
elif futureDay == 2:
  nameForFutureDay = "Tuesday"
elif futureDay == 3:
  nameForFutureDay = "Wednesday"
elif futureDay == 4:
  nameForFutureDay = "Thursday"
elif futureDay == 5:
  nameForFutureDay = "Friday"
else: # if futureDay == 6:
  nameForFutureDay = "Saturday"

print("Today is " + nameForToday
    + " and the future day is " + nameForFutureDay)
```

Program #12 (Solution)

```
import random

# 1. Generate two random single-digit integers
number1 = random.randint(0, 9)
number2 = random.randint(0, 9)

# 2. If number1 < number2, swap number1 with number2
if number1 < number2:
    number1, number2 = number2, number1

# 3. Prompt the student to answer What is number1 - number2?
answer = eval(input("What is " + str(number1) + " - "
                    + str(number2) + "? "))

# 4. Repeatedly ask the user the question until it is correct
while number1 - number2 != answer:
    answer = eval(input("Wrong answer. Try again. What is "
                        + str(number1) + " - " + str(number2) + "? "))

# 5. Print the answer is correct
print("You got it!")
```

Program #13 (Solution)

```python
import random

# Generate a random number to be guessed
number = random.randint(0, 100)

print("Guess a magic number between 0 and 100")

guess = -1 # Initial value that doesn't meet the loop condition
while guess != number:
    # Prompt the user to guess the number
    guess = eval(input("Enter your guess: "))

    if guess == number:
        print("Yes, the number is", number)
    elif guess > number:
        print("Your guess is too high")
    else:
        print("Your guess is too low")
```

Program #14 (Solution)

```
import random
import time

correctCount = 0 # Count the number of correct answers
count = 0 # Count the number of questions
NUMBER_OF_QUESTIONS = 5 # Constant

startTime = time.time() # Get start time

while count < NUMBER_OF_QUESTIONS:
    # 1. Generate two random single-digit integers
    number1 = random.randint(0, 9)
    number2 = random.randint(0, 9)

    # 2. If number1 < number2, swap number1 with number2
    if number1 < number2:
        number1, number2 = number2, number1
 # 3. Prompt the student to answer "what is number1 - number2?"
    answer = eval(input("What is " + str(number1) + " - " +
        str(number2) + "? "))

    # 5. Grade the answer and display the result
    if number1 - number2 == answer:
        print("You are correct!")
        correctCount += 1
    else:
        print("Your answer is wrong.\n", number1, "-",
            number2, "should be", (number1 - number2))

    # Increase the count
    count += 1

endTime = time.time() # Get end time
testTime = int(endTime - startTime) # Get test time
print("Correct count is", correctCount, "out of",
    NUMBER_OF_QUESTIONS, "\nTest time is", testTime, "seconds")
```

Program #15 (Solution)

```python
import random
import time

correctCount = 0 # Count the number of correct answers
count = 0 # Count the number of questions

startTime = time.time() # Get start time

continueLoop = 'Y' # User confirmation flag
while continueLoop == 'Y':
    # 1. Generate two random single-digit integers
    number1 = random.randint(0, 9)
    number2 = random.randint(0, 9)

    # 2. If number1 < number2, swap number1 with number2
    if number1 < number2:
        number1, number2 = number2, number1

    # 3. Prompt the student to answer "what is number1 – number2?"
    answer = eval(input("What is " + str(number1) + " - " +
        str(number2) + "? "))

    # 5. Grade the answer and display the result
    if number1 - number2 == answer:
        print("You are correct!")
        correctCount += 1
    else:
        print("Your answer is wrong.\n", number1, "-",
            number2, "should be", (number1 - number2))

    # Increase the count
    count += 1

    # Prompt the user for confirmation
    continueLoop = input("Enter Y to continue and N to quit: ")
    print() # To insert a new line

endTime = time.time() # Get end time
testTime = int(endTime - startTime) # Get test time
print("Correct count is", correctCount, "out of",
    count, "\nTest time is", testTime, "seconds")
```

Program #16 (Solution)

```
data = eval(input("Enter an integer (the input exits " +
    "if the input is 0): "))

# Keep reading data until the input is 0
sum = 0
while data != 0:
    sum += data

    data = eval(input("Enter an integer (the input exits " +
        "if the input is 0): "))

print("The sum is", sum)
```

Program #17 (Solution)

```
print("          Multiplication Table")
# Display the number title
print("  |", end = '')
for j in range(1, 10):
    print("  ", j, end = '')
print() # Jump to the new line
print("-----------------------------------------")

# Display table body
for i in range(1, 10):
    print(i, "|", end = '')
    for j in range(1, 10):
        # Display the product and align properly
        print(format(i * j, '4d'), end = '')
    print()# Jump to the new line
```

Program #18 (Solution)

```
# Prompt the user to enter two integers
n1 = eval(input("Enter first integer: "))
n2 = eval(input("Enter second integer: "))

gcd = 1 # Initial gcd is 1
k = 2 # Possible gcd
while k <= n1 and k <= n2:
    if n1 % k == 0 and n2 % k == 0:
        gcd = k # # Next possible gcd
    k += 1

print("The greatest common divisor for",
    n1, "and", n2, "is", gcd)
```

Program #19 (Solution)

```
year = 0 # Year 0
tuition = 10000   # Year 1

while tuition < 20000:
    year += 1
    tuition = tuition * 1.07

print("Tuition will be doubled in", year, "years")
print("Tuition will be $" + format(tuition, ".2f"),
       "in", year, "years")
```

Program #20 (Solution)

```
NUMBER_OF_PRIMES = 50  # Number of primes to display
NUMBER_OF_PRIMES_PER_LINE = 10 # Display 10 per line
count = 0 # Count the number of prime numbers
number = 2 # A number to be tested for primeness

print("The first 50 prime numbers are")

# Repeatedly find prime numbers
while count < NUMBER_OF_PRIMES:
    # Assume the number is prime
    isPrime = True #Is the current number prime?

    # Test if number is prime
    divisor = 2
    while divisor <= number / 2:
        if number % divisor == 0:
            # If true, the number is not prime
            isPrime = False  # Set isPrime to false
            break  # Exit the for loop
        divisor += 1
# If number is prime, display the prime number and increase the count
if isPrime:
    count += 1 # Increase the count

    print(format(number, '5d'), end = '')
    if count % NUMBER_OF_PRIMES_PER_LINE == 0:
        # Display the number and advance to the new line
        print() # Jump to the new line

# Check if the next number is prime
number += 1
```

Program #21 (Solution)

```
# Define two variables to be counters for positive and negative numbers
countPositive = 0
countNegative = 0
# Define a variable to be a counter for all numbers regardless of +,-
count = 0
# Define a variable to store the sum of all entered numbers
total = 0

# Ask the user to enter a number
num = eval(input(
    "Enter an integer, the input ends if it is 0: "))

# Define a while loop and set its condition to number != 0
# (while number is not equal to 0)
while num != 0:
    # Check the number to determine if its positive or negative
    if num > 0:
        countPositive += 1
    elif num < 0:
        countNegative += 1

    # Add number to total
    total += num
    # Increase count by one
    count += 1

    # Ask the user to enter the next number
    num = eval(input(
        "Enter an integer, the input ends if it is 0: "))

# Check count to determine if the user entered any number except 0,
# and then display the result
if count == 0:
    print("No numbers are entered except 0")
else:
    # Calculate the average
    average = total / count

    print("The number of positives is", countPositive)
    print("The number of negatives is", countNegative)
    print("The total is", total)
    print("The average is", average)
```

Program #22 (Solution)

```
# Table header
print(format("kilograms", "<10s"),
      format("pounds", ">10s"),
      "     |     ",
      format("pounds", "<10s"),
      format("kilograms", ">10s"))
print("--------------------------------------------------------")

# Define the counters
kilograms = 1
pounds = 20
count = 0

# Table body
while count < 100:
    # Convert units
    kilograms_to_pounds = kilograms * 2.2
    pounds_to_kilograms = pounds / 2.2
    # Print a new formatted row
    print(format(kilograms, "<10d"),
          format(kilograms_to_pounds, "10.1f"),
          "     |     ",
          format(pounds, "<10d"),
          format(pounds_to_kilograms, "10.2f"))
    # Update the counters
    kilograms += 2
    pounds += 5
    count += 1
```

Program #23 (Solution)

```python
# Prompt the user to enter the first number
number = eval(input("Enter a number (0: for end of input): "))

if number == 0:
    print("No numbers are entered except 0")
else:
    max = number
    count = 1

    # Prompt the user to enter the remaining numbers
    while number != 0:
        number = eval(input("Enter a number (0: for end of input): "))

        if number > max:
            max = number
            count = 1
        elif number == max:
            count += 1

    print("The largest number is", max)
    print("The occurrence count of the largest number is", count)
```

Program #24 (Solution)

```python
# Prompt the user to enter the number of students
numOfStudents = eval(input("Enter the number of students: "))

# Initialize the variables for the highest score
student1 = input("Enter a student name: ")
score1 = eval(input("Enter a student score: "))

# Get students scores
for i in range(numOfStudents - 1):
    student = input("Enter a student name: ")
    score = eval(input("Enter a student score: "))

    # Check the score of the entered student with the highest score
    if score > score1:
        student1 = student
        score1 = score

# Display the output
print()  # Empty line
print("Top student " + student1 + "'s score is " + str(score1))
```

Program #25 (Solution)

```python
# Prompt the user to enter the number of students
numberOfStudents = eval(input("Enter the number of students: "))

student1 = input("Enter a student name: ")
score1 = eval(input("Enter a student score: "))

student2 = input("Enter a student name: ")
score2 = eval(input("Enter a student score: "))

# Make sure that student1 is for the highest
# and student2 is for the second highest
if score1 < score2:
    # Swap
    student1, student2 = student2, student1
    score1, score2 = score2, score1

for i in range(numberOfStudents - 2):
    student = input("Enter a student name: ")
    score = eval(input("Enter a student score: "))

    if score > score1:
        # new student becomes the highest
        student1, student2 = student, student1 # student1 now is the second highest
        score1, score2 = score, score1
    elif score > score2:
        student2 = student # new student becomes the second highest
        score2 = score

    print("Top two students:")
    print(student1 + "'s score is " + str(score1))
    print(student2 + "'s score is " + str(score2))
```

Program #26 (Solution)

```
# Pattern 1
for i in range(1, 6 + 1):
    for j in range(1, i + 1):
        print(j, end = " ")
    print()
# Pattern 2
for i in range(1, 6 + 1):
    for j in range(1, 7 - i + 1):
        print(j, end = " ")
    print()
# Pattern 3
for i in range(1, 6 + 1):
    # Print leading space
    for j in range(6 - i, 0, -1):
        print("  ", end="")

    for j in range(i, 0, -1):
        print(j, end=" ")

    print()
# Pattern 4
for i in range(1, 6 + 1):
    # Print leading space
    for j in range(i, 1, -1):
        print("  ", end="")

    for j in range(1, 6 + 1 - i + 1):
        print(j, end=" ")

    print()
```

Program #27
(Solution)

```
import math

print(format("Degree", "<15s"), format("Sin", "<15s"),
      format("Cos", "<15s"))

for degree in range(0, 360 + 1, 10):
    print(format(degree, "<15d"),
          format(math.sin(degree * math.pi / 180), "<15.4f"),
          format(math.cos(degree * math.pi / 180), "<15.4f"))
```

Program #28 (Solution)

```python
count = 1
for i in range(100, 201):
    if not (i % 5 == 0 and i % 6 == 0) and (i % 5 == 0 or i % 6 == 0):
        if count % 10 != 0:
            print(i, end = " ")
        else:
            print(i)

        count += 1
```

Program #29 (Solution)

```python
i = 1

while i * i <= 12000:
    i += 1

print("This number is", i)
```

Program #30 (Solution)

```
i = 1

while i ** 3 < 12000:
    i += 1

print("This number is", i - 1)
```