

Computer Science 217

Midterm Exam

October 29, 2014

First Name: _____

Last Name: _____

ID: _____

Class Time (Circle One): 1:00pm 3:00pm

Instructions:

- Neatly print your names and ID number in the spaces provided above.
- Pick the **best answer** for each multiple choice question.
- Answer each question by writing the correct answer in the space provided. **Answer all multiple choice questions using UPPER CASE letters.**
- This exam consists of 12 pages, including the cover. Before answering any questions count the pages and ensure that they are all present.
- You have 1 hour 30 minutes to complete this exam.
- Unless noted otherwise, each question is worth one mark.
- This exam is closed book. You are not permitted to use any electronic devices or reference materials.
- **DO NOT TURN PAST THIS PAGE UNTIL YOU ARE INSTRUCTED TO BEGIN**

1. (12 marks) The greatest common divisor of two integers, a and b, is the largest integer that divides evenly into both a and b. The following algorithm can be used to find the greatest common divisor, x, of the integers:

Set x equal to the absolute value of the larger of a and b

While the greatest common divisor has not been found

 If x divides evenly into both a and b

 Mark that the greatest common divisor has been found

 Otherwise

 Reduce the value of x by 1

When this algorithm completes x will contain the greatest common divisor of a and b. Note that this algorithm will result in a division by 0 error if both a and b are 0.

Write a program that reads pairs of integers from the user. Display the greatest common divisor of each pair entered, including an output message that matches the sample output shown below. Continue reading values from the user until the user enters 0 for a and 0 for b. Your program should not generate any output if the first pair of values entered is a pair of zeros.

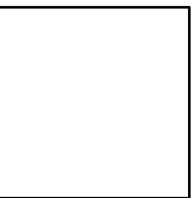
Use the algorithm shown above to compute the greatest common divisors. Do **not** write any functions as part of your solution. You may include comments in your solution if you would like to, but they are not required.

Sample input and output is shown below. Values entered by the user are shown in bold:

```
Enter an integer: 75
Enter an integer: 50
The gcd of 75 and 50 is 25
Enter an integer: 0
Enter an integer: 4
The gcd of 0 and 4 is 4
Enter an integer: 12
Enter an integer: 0
The gcd of 12 and 0 is 12
Enter an integer: -6
Enter an integer: 9
The gcd of -6 and 9 is 3
Enter an integer: -21
Enter an integer: -14
The gcd of -21 and -14 is 7
Enter an integer: 0
Enter an integer: 0
```

Place your answer to this question on the next page. Do **not** answer the question at the bottom of this page.

Place your answer to question 1 on this page.



2. Which of the following is the **best** definition of a modern computer?
- A. A mechanical device capable of outputting information
 - B. A tool that receives, processes and presents data
 - C. An electronic device that accepts information in digital form
 - D. A mechanism for performing logic and control
 - E. A boxy device with a typewriter-like interface that processes information

Answer: _____

3. Which of the following devices was used to assist with calculations for the first time in the 1940s?
- A. Abacus
 - B. Babbage difference engine
 - C. ENIAC
 - D. Moore's Machine
 - E. Transistor

Answer: _____

4. The area of computer science that has complexity and computability subfields is:
- A. Artificial Intelligence
 - B. Distributed Systems
 - C. Human Computer Interaction
 - D. Software Engineering
 - E. Theory of Computation

Answer: _____

5. Which of the following components has a direct connection to the CPU in a modern computer architecture?
- A. Graphics subsystem
 - B. Hard disk
 - C. Keyboard
 - D. Printer
 - E. Sound subsystem

Answer: _____

6. Which of the following levels of competence indicates the greatest depth of understanding?
- A. Analysis
 - B. Comprehension
 - C. Elicitation
 - D. Knowledge
 - E. Synthesis

Answer: _____

7. Consider the following steps:

- Begin with the entire problem
- Break it down into 3 to 5 steps
- Repeat the previous operation for each step that is still too complex

These steps are known as:

- A. Application
- B. Bloom's Taxonomy
- C. Complexity Reduction
- D. Programming
- E. Top Down Design

Answer: _____

8. Consider a program that computes the number of roots of a quadratic equation and stores the result in a variable. Which of the following names would be the **best** choice for the variable?

- A. NUMROOTS
- B. num_roots
- C. The-Number-Of-Roots-Of-The-Equation
- D. z
- E. #roots

Answer: _____

9. Which of the following statements is most correct?

- A. Compiling a program creates an executable file that can be run on any kind of computer
- B. Compiling a program makes it more difficult for a competitor to copy the ideas and techniques that you used to solve a problem
- C. Running a program on a virtual machine typically gives better performance than running a compiled program
- D. Exactly two of the above answers are correct
- E. Answers a, b, and c are all correct

Answer: _____

10. What value is displayed by the following line of code?

```
print(1 + 2 ** 3 * 4)
```

Answer: _____

Consider the following program for the next two questions. It is supposed to compute the hypotenuse of a right triangle after the user enters the lengths of the other two sides.

```
a = float(input("Enter the length of the first side: "))
b = float(input("Enter the length of the second side: "))

h = sqrt(a * a + b * b)

print("The length of the hypotenuse is", h)
```

When this program is run, the following output is generated (note that input entered by the user is shown in bold):

```
Enter the length of the first side: 3
Enter the length of the second side: 4
Traceback (most recent call last):
  File "triangle.py", line 4, in <module>
    h = sqrt(a * a + b * b)
NameError: name 'sqrt' is not defined
```

11. The error that is displayed by this program is an example of:

- A. An input error
- B. A logic error
- C. A runtime error
- D. A syntax error
- E. A variable error

Answer: _____

12. In one sentence, what would you need to do to correct this error so that the program computes the desired result?

13. Which of the following expressions will result in an error message being displayed when a program containing it is run?

- A. $2.0 / 4$
- B. `"3" + "Hello"`
- C. $4 \% 15$
- D. `int("5") / float("3")`
- E. `float("6" / "2")`

Answer: _____

14. Which of the following statements displays the value of variable x so that is rounded to 3 decimal places?

- A. `print("The value of x is %.3f" % x)`
- B. `print("The value of x is %.3x" % f)`
- C. `print("The value of x is %f" % round(x, 3))`
- D. `print("The value of x is %x", .3f)`
- E. `print("The value of x is %x" % .3f)`

Answer: _____

15. Data that has two possible values, true and false, is known as _____ data.

16. Which of the following sequences of characters is **not** a valid number in base 14?

- A. 0
- B. 14
- C. 4A
- D. CA
- E. E8

Answer: _____

17. Convert 505 base 6 to base 10. Ensure that you place your answer in the space provided.

Answer: _____

18. Convert 101 base 10 to binary. Ensure that you place your answer in the space provided.

Answer: _____

19. (2 marks) Convert 4321 base 5 to base 15. Ensure that you place your answer in the space provided.

Answer: _____

20. The ASCII character encoding scheme uses 7 bits to represent each character. How many different characters can be represented in this system?

Answer: _____

21. Using two sentences or less, explain why floating point numbers are only an approximation of real numbers.

22. Which type of statement should be used to select exactly one of several alternatives?

- A. Compound statement
- B. If statement
- C. If-elif statement
- D. If-elif-else statement
- E. If-else statement

Answer: _____

23. Complete the truth table for the expression (not A or not B) and (A or B). **Draw a box around the column that represents your answer.**

A	B	
0	0	
0	1	
1	0	
1	1	

24. Which of the following is an example of a relational operator?

- A. ^
- B. =
- C. %
- D. !=
- E. not

Answer: _____

Consider the following code segment:

```
a = int(input())
b = int(input())

if a > 0 and b > 0:
    a = a + 1
    b = b - 1
if a > 0 or b < 0:
    b = b - 1
if b > 0 or a < 0:
    a = a + 1

print(a, b)
```

25. What output is generated if the user enters 1 for a and 1 for b? _____

26. What output is generated if the user enters 2 for a and 3 for b? _____

27. What output is generated if the user enters 0 for a and -1 for b? _____

Consider the following code segment:

```
a = int(input())
b = int(input())

if a > 0:
    b = b + 1
    if b > 0:
        a = a - 1
    else:
        a = a + 1
elif b > 0:
    a = a + 1
    if a > 0:
        b = b - 1
    else:
        b = b + 1

print(a, b)
```

28. What output is generated if the user enters 1 for a and 0 for b? _____

29. What output is generated if the user enters 0 for a and 2 for b? _____

Consider the following code segment:

```
a = int(input())
x = 2

for i in range(2, a):
    x = x * 2

print(x)
```

30. What output is generated by the code segment if the user enters 2 for a? _____

31. What output is generated by the code segment if the user enters 8 for a? _____

Consider the following code segment:

```
for i in range(1, 10):
    print("X")
    for j in range(1, 50):
        print("X")
        print("X")
```

32. How many copies of the letter X are displayed when this code segment runs? _____

Consider the following code segment:

```
a = int(input())
b = int(input())
x = 0
y = 0

for i in range(0, a):
    j = i
    while j < b:
        if j % 3 == 0:
            y = y + 1
            j = j + 1
            x = x + 1

print(j, x, y)
```

33. What output is generated if the user enters 2 for a and 2 for b? _____

34. What output is generated if the user enters 2 for a and 3 for b? _____

35. What output is generated if the user enters 3 for a and 2 for b? _____

36. (3 marks) A date consists of a day number, a month number and a year number. A magic date is a date where the day multiplied by the month is equal to the last two digits of the year. For example, 13/05/1965 is a magic date because $13 * 5 = 65$.

The following program is supposed to find all of the magic dates in the 1900s. However, it contains 3 bugs on three different lines that are preventing it from completing its intended task. All of the comments are correct, describing what each line of code is supposed to do. Circle each of the 3 bugs (and only the bug) and write **one sentence or less next to the bug describing what you would need to change to fix it**. Note that in this context a bug means an error that causes the program to crash or compute an incorrect result. It does **not** including stylistic issues like poor variable names.

```
# For each year from 1900 to (and including) 1999
for year in range(1900, 2000):
    # For each month January to (and including) December
    for month in range(1, 12):
        # Determine if it is a leap year
        # All years that are divisible by 400 are leap years
        if year % 400 == 0:
            isLeapYear = True
        # All of the remaining years that are divisible by 100 are not leap years
        elif year % 100 == 0:
            isLeapYear = False
        # All remaining years that are divisible by 4 are leap years
        elif year % 4 == 0:
            isLeapYear = True
        # All remaining years are not leap years
        else:
            isLeapYear = False

        # Determine the number of days in the month
        # Start out by assuming that the month will have 31 days because that is
        # the most common case
        days = 31
        # If the month is April, June, September or November then reduce the
        # number of days to 30
        if month == 4 and month == 6 and month == 9 and month == 11:
            days = 30
        # If the month is February then reduce the days to 29 if it is a leap
        # year, or 28 if it is not a leap year
        if month == 2:
            if isLeapYear:
                days = 29
            else:
                days = 28

        # Check each day to see if it is magic
        # For every day in the month
        for day in range(1, days + 1):
            # If day * month equals the 2 last two digits of the year
            if day * month == year // 100:
                # Display a message indicating that the date is magic
                print("%d/%d/%d is a magic date." % (day, month, year))
```