

German University in Cairo
Media Engineering and Technology Faculty
Prof. Dr. Slim Abdennadher
Dr. Amr Desouky

November 26, 2015

CSEN102: Introduction to Computer Science Winter Semester 2015-2016 Midterm Exam

Bar Code

Instructions: Read carefully before proceeding.

- 1) Please tick your major

	Major
	Engineering
	BI

- 2) Duration of the exam: 2 hours (120 minutes).
- 3) No books or other aids are permitted for this test.
- 4) This exam booklet contains 12 pages, including this one. Three extra sheets of scratch paper are attached and have to be kept attached. **Note that if one or more pages are missing, you will lose their points. Thus, you must check that your exam booklet is complete.**
- 5) Write your solutions in the space provided. If you need more space, write on the back of the sheet containing the problem or on the four extra sheets and make an arrow indicating that. **Scratch sheets will not be graded unless an arrow on the problem page indicates that the solution extends to the scratch sheets.**
- 6) When you are told that time is up, stop working on the test.

Good Luck!

Don't write anything below ; -)

Exercise	1	2	3	4	5	6	Σ
Possible Marks	10	8	12	8	10	12	60
Final Marks							

Exercise 1 Tracing
Mysterious Task

(3+4+3 = 10 Marks)

- a) What does the following program display for any boolean values x , y and z ? Choose one answer from the below choices and Justify.

```
x = eval(input())
y = eval(input())
z = eval(input())
if (x and y):
    print((not x) or z)
else:
    print((x and y) or z)
```

1. The value of x
2. The value of y
3. The value of z
4. The value of x and y
5. Always true
6. True if either x and y are both True or z is True, and False otherwise.

Solution:

The solution is the value of z .

If x and y are true, then $\text{not } x$ is false and the then part will display then the value of z . Otherwise, either x or y is false, then x and y will be false too, thus the else part will display the value of z too.

- b) Consider a program which changes the value of a boolean variable that tracks whether a light is on or off. Three people, creatively named Sarah, Ali, and Mina, who are claiming to be expert light switchers have written different implementations of this function.

Sarah's Version:

```
lightOn = eval(input())
if (lightOn):
    lightOn = False
else:
    lightOn = True
```

Ali's Version:

```
lightOn = eval(input())
if (lightOn):
    lightOn = False
if (not lightOn):
    lightOn = True
```

Mina's Version:

```
lightOn = eval(input())
lightOn = not lightOn
```

Are these implementations equivalent, i.e. performing the same task? Are all three people expert light switchers? Why? Justify your answer.

Solution:

No, these implementations are not equivalent. Sarah and Mina are experts, but Ali is a fraud. If the light is on, his code turns it off then the second condition will be true and will go to the then part, so it will always remain on.

c) Given the following algorithm

```
smog = eval(input())
emissions = 0
while (not smog == 0):
    emissions = 10 * emissions
    emissions = emissions + (smog % 10)
    smog = smog // 10
print(emissions)
```

1. What is the output of the algorithm above for `smog = 143`?

Solution:

341

2. What does the algorithm do for any integer number `smog`?

Solution:

The algorithm computes and displays the reverse of any integer number.

Exercise 2 Conditional Algorithms
Median of a List

(8 Marks)

The median, or middle value, of a list of numbers is extremely useful in a variety of computer and statistical algorithms. But it is notoriously slow to compute relative to the number of times it needs to be computed. Often, it is better to settle for an approximation of the median than to calculate it exactly. One of the simplest approximations is to pick three random elements, and calculate the median of those three.

Write a Python program that takes a list of integers and calculates the median of three randomly chosen elements from the list. **Note:** The median of three numbers is the second largest element.

For example,

- for the list `[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]` and if the three randomly selected numbers are 10, 1 and 8, your algorithm should output 8 since 8 is the second largest element among the three randomly selected numbers.
- for the list `[10, 2, 22, 4, 5, 6, 11]` and if the three randomly selected numbers are 6, 6 and 6, your algorithm should output 6.

Note: You can use the method `random.randint`.

Solution:

```
import random
l = eval(input())
a = l[random.randint(0, len(l)-1)]
print(a)
b = l[random.randint(0, len(l)-1)]
print(b)
c = l[random.randint(0, len(l)-1)]
print(c)
if (a <= b and b <= c):
    print(b)
else:
    if (c <= b and b <= a):
        print(b)
    else:
        if (b <= a and a <= c):
            print(a)
        else:
            if (c <= a and a <= b):
                print(a)
            else:
                print(c)
```

Exercise 3 Iterative Algorithms
Pair of Dice

(12 Marks)

Write a Python program that takes as input two integers x and y (between 1 and 12). Assume that the two integers are a result of throwing a pair of dice. Let r be the thrown pair of dice.

The program should simulate repeated rolls of a pair of dice until 100 rolls of r have been obtained. The program should print how many rolls it took to obtain 100 rolls of r . **Note:** Use the function `random.randint`.

For example, if the user enters $x = 5$ and $y = 9$, the program should display the following message:

```
Roll 1
dice1 = 3
dice2 = 10
Number of times you rolled the pair of dice is 1
Number of times you obtained the pair of dice 5 and 9 is 0

Roll 2
dice1 = 5
dice2 = 11
Number of times you rolled the pair of dice is 2
Number of times you obtained the pair of dice 5 and 9 is 0

Roll 3
dice1 = 5
dice2 = 9
Number of times you rolled the pair of dice is 3
Number of times you obtained the pair of dice 5 and 9 is 1

Roll 4
dice1 = 11
dice2 = 9
Number of times you rolled the pair of dice is 4
Number of times you obtained the pair of dice 5 and 9 is 1

Roll 5
dice1 = 9
dice2 = 5
Number of times you rolled the pair of dice is 5
Number of times you obtained the pair of dice 5 and 9 is 2

...

Roll 14815
dice1 = 9
dice2 = 5
Number of times you rolled the pair of dice is 14815
Number of times you obtained the pair of dice 5 and 9 is 100
Congratulations
```

Solution:

```
import random
r1 = eval(input())
r2 = eval(input())
c = 0
i = 0
while(c < 100):
    x = random.randint(1,12)
    y = random.randint(1,12)
    i += 1
    if((r1==x and r2 == y) or (r1 == y and r2 == x)):
        c += 1
    print("Roll", i)
    print("dice1 = ", x)
    print("dice2 = ", y)
    print("Number of times you rolled the pair of dice is ", i)
    print("Number of times you obtained the pair of dice ", r1, " and ", r2, " is ",
print("Congrats")
```

Exercise 4 Iterative Algorithms
Tracing

(5+3=8 Marks)

Given the following program

```

n = eval(input("Size:"))
print(n)
list = [0] * n
i = 0
while(i < n):
    x = eval(input("Enter a number:"))
    list[n - i - 1] = x
    i = i + 1

i = 1
while(i < n):
    list[i] = list[i] + list[i - 1]
    i = i + 1

print(list)

```

Note: `list = [0] * n` creates a list of length `n` and all elements have the value 0.

a) What does the algorithm display for the following input? Trace your algorithm.

```

Size: 5
Enter a number: 1
Enter a number: 3
Enter a number: -1
Enter a number: 4
Enter a number: 2

```

Solution:The program produces and displays the list `[2, 6, 5, 8, 9]` which corresponds to
$$[2, 2 + 4, 2 + 4 + (-1), 2 + 4 + (-1) + 3, 2 + 4 + (-1) + 3 + 1]$$
b) What does the algorithm do for any sequence of numbers of length `n`.**Solution:**

The program asks the user to enter `n` numbers and returns an integer list of length `n` where the value at index 0 is the sum of the last number read, the value at index 1, the sum of the last two numbers read, the value at index 2, the sum of the last three values read, etc.

Exercise 5 Iterative Algorithms
Vegas, Here We Come

(10 Marks)

Write a Python program that takes an integer list L and returns a new integer list that is a reordered version of L , in which elements from the second half of L are perfectly interleaved with the elements from the first half. If L contains an odd number of elements, the central element should end up as the last element of the reordered list. You can assume that the program receives a valid integer list, but it might be of length 0. Here are some example calls of the program and their results:

Initial List	Reordered List
<code>[]</code>	<code>[]</code>
<code>[1, 2, 3, 4, 5]</code>	<code>[1, 4, 2, 5, 3]</code>
<code>[1, 2, 3, 4, 5, 6, 7, 8]</code>	<code>[1, 5, 2, 6, 3, 7, 4, 8]</code>

Solution:

```
l = eval(input())
r = [0]*len(l)
mid = (len(l) + 1) // 2
i = 0
while(i < len(r)):
    r[i] = l[i // 2]
    i += 2
i = 1
while(i < len(r)):
    r[i] = l[mid + i // 2]
    i += 2
print(r)
```


Exercise 6 Iterative Algorithms (12 Marks)
Increasing Sequence

Write a Python program that takes a list of integers, and computes and displays the length of the longest sequence of consecutive, increasing numbers in the list. For example, the longest sequence of consecutive increasing numbers in the list

```
[3, 45, 6, 12, 18, 21, 21, 19, 41, 2]
```

is

```
[6, 12, 18, 21]
```

Therefore, the program would display 4 for this list. If the list is empty, the program should display 0. As long as the list has at least one element, the longest sequence of consecutive increasing numbers should be at least 1.

Solution:

```
list = eval(input())

if (len(list) == 0):
    print(0)
maxLength = 1
count = 1
i = 1
while(i < len(list)):
    if (list[i] > list[i - 1]):
        count += 1
    else:
        if (count > maxLength):
            maxLength = count
        count = 1
    i += 1
print(maxLength)
```

Scratch paper

Scratch paper

Scratch paper
