VINUNIVERSITY

Lab Practice № 8 (Solution): Tuples, Dictionaries, Algorithm Complexity, and Linear Search

COMP1010 Introduction to Programming **Week 09**

Lab Practice Submission Instructions:

- This is an individual lab practice and will typically be assigned in the laboratory (computer lab).
- Your program should work correctly on all inputs. If there are any specifications about how the program should be written (or how the output should appear), those specifications should be followed.
- Your code and functions/modules should be appropriately commented. However, try to avoid making your code overly busy (e.g., include a comment on every line).
- Variables and functions should have meaningful names, and code should be organized into functions/methods where appropriate.
- Academic honesty is required in all work you submit to be graded. You should **NOT** copy or share your code with other students to avoid plagiarism issues.
- Use the template provided to prepare your solutions.
- You should upload your file(s) to the Canvas **before the end of the laboratory session** unless the instructor gave a specified deadline.
- Late submission of lab practice without an approved extension will incur the following penalties:

  (a) No submission during the laboratory session will incur one (1) point deduction from the total possible score of the lab practice.

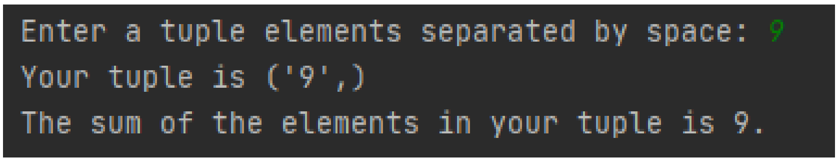
  (b) The instructor will deduct an additional point for each business day past the deadline.

  (c) The penalty will be deducted until the maximum possible score for the lab practice reaches zero (0%) unless otherwise specified by the instructor.

Problem 1 - The sum of the elements in a tuple

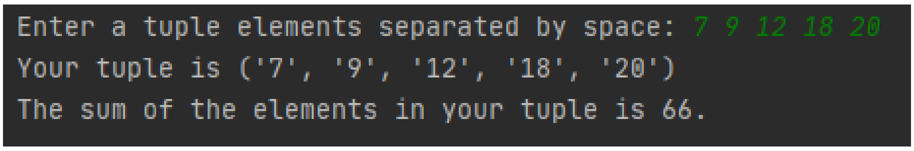
Write a program to create a tuple named `my_tuple` with the elements input by the user. Next, find the sum of the elements in `my_tuple`. The sample outputs are as follows:

Sample output 1:



```
Enter a tuple elements separated by space: 9
Your tuple is ('9',)
The sum of the elements in your tuple is 9.
```

Sample output 2:



```
Enter a tuple elements separated by space: 7 9 12 18 20
Your tuple is ('7', '9', '12', '18', '20')
The sum of the elements in your tuple is 66.
```

Submit your program to CMS.

Problem 2 – Dictionary

Mr. Baba is a finance manager at ABC mart, and at the end of the year, he needs to prepare a 4-year sales report for the company. He needs to check which month in a specific year that achieved the monthly sales target set by the company CEO. Also, he needs to report the total sales for each year. The sales record for the year 2017, 2018, 2019, and 2020 are given as follows:

Month	2020	2019	2018	2017
Jan	500	1500	100	410
Feb	1200	1000	1500	620
March	750	850	2150	1150
Apr	250	1250	560	190
May	950	980	780	380
June	1350	1020	820	220
Jul	300	800	400	490
Aug	320	1320	220	1120
Sept	1000	960	1960	190
Oct	200	1200	920	1130
Nov	1400	1300	600	330
Dec	980	1980	180	720

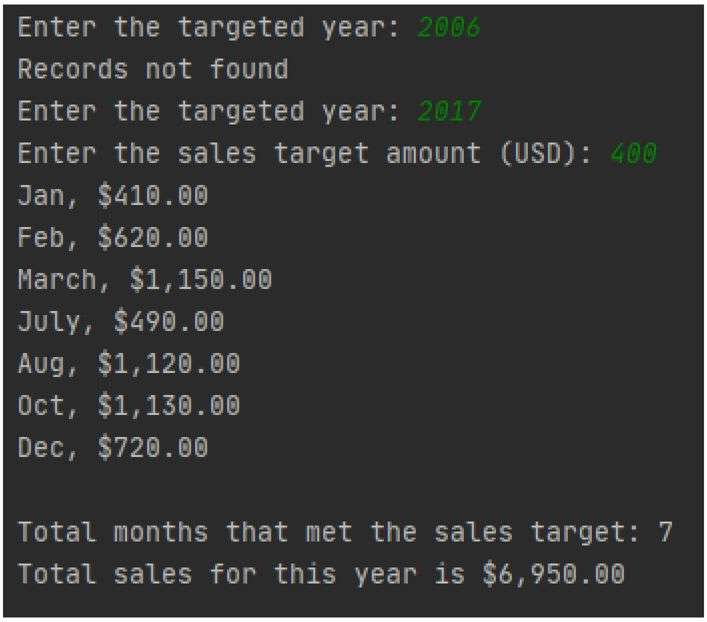
Assume that the CEO has yet to inform the sales target of each year to Mr. Baba, write a program that allows Mr. Baba to extract the necessary information from the above sales records. First, create a dictionary that holds the sales record for each year, e.g., `sales_2020` for the year 2020. The months and sales will be used as the dictionary's keys and values, respectively. Next, create a variable `sales_year={2020, 2019, 2018, 2017}` that holds the years for all the sales record. The program should allow Mr. Baba to:

1. Search for monthly records that met the target sales in a specific year.
2. See the total months that met the target sales.
3. See the total sales in a specific year.
4. Notify the user if the sales record of the year specified is not found.

Use the sample output to verify your program. Note that the sample output is based on random sales target (for testing purposes).

Sample Output:

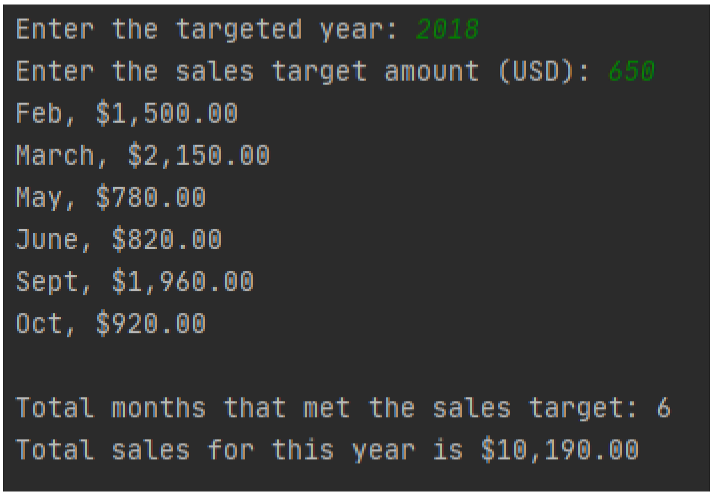
Case A:



```
Enter the targeted year: 2006
Records not found
Enter the targeted year: 2017
Enter the sales target amount (USD): 400
Jan, $410.00
Feb, $620.00
March, $1,150.00
July, $490.00
Aug, $1,120.00
Oct, $1,130.00
Dec, $720.00

Total months that met the sales target: 7
Total sales for this year is $6,950.00
```

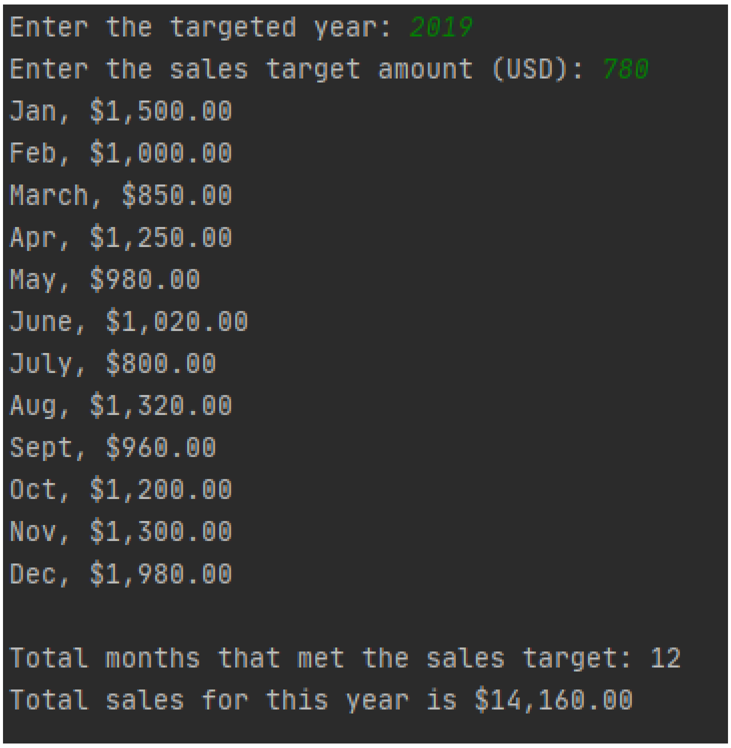
Case B:



```
Enter the targeted year: 2018
Enter the sales target amount (USD): 650
Feb, $1,500.00
March, $2,150.00
May, $780.00
June, $820.00
Sept, $1,960.00
Oct, $920.00

Total months that met the sales target: 6
Total sales for this year is $10,190.00
```

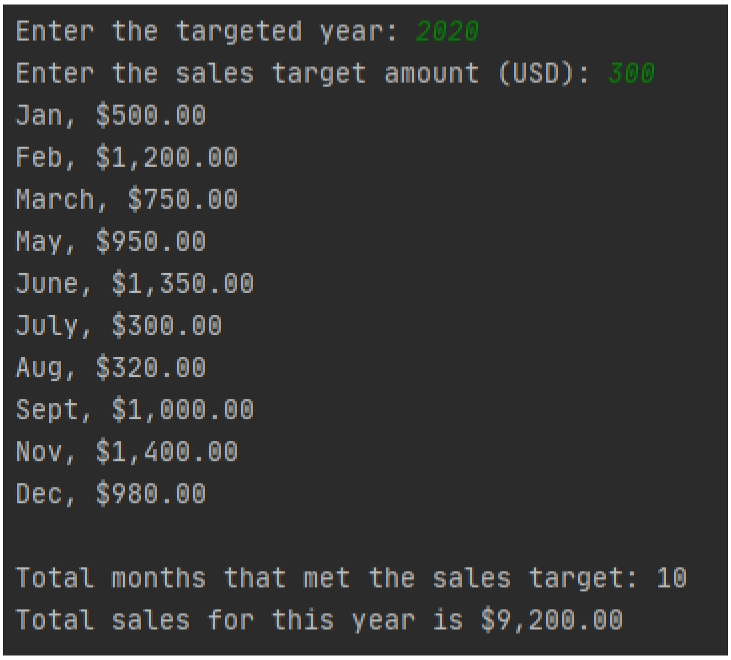
Case C:



```
Enter the targeted year: 2019
Enter the sales target amount (USD): 780
Jan, $1,500.00
Feb, $1,000.00
March, $850.00
Apr, $1,250.00
May, $980.00
June, $1,020.00
July, $800.00
Aug, $1,320.00
Sept, $960.00
Oct, $1,200.00
Nov, $1,300.00
Dec, $1,980.00

Total months that met the sales target: 12
Total sales for this year is $14,160.00
```

Case D:



```
Enter the targeted year: 2020
Enter the sales target amount (USD): 300
Jan, $500.00
Feb, $1,200.00
March, $750.00
May, $950.00
June, $1,350.00
July, $300.00
Aug, $320.00
Sept, $1,000.00
Nov, $1,400.00
Dec, $980.00

Total months that met the sales target: 10
Total sales for this year is $9,200.00
```

Problem 3 – Complexity and Big-O Notation

Problem 3A. Determine the complexity of the iterative Fibonacci function in Big-O notation. Briefly explain your answer in 2 – 3 sentence. You can use "multi-line comments" in Python to write your answer and submit it to Canvas. CMS is not required for this problem.

Listing 1: Problem 3A Solution.

```python
def my_fib_iter(n):

    out = [1, 1]

    for i in range(2, n):
        out.append(out[i - 1] + out[i - 2])

    return out
```

Problem 3B. What is the complexity of the following function in Big-O notation? Briefly explain your answer in 2 – 3 sentence. You can use "multi-line comments" in Python to write your answer and submit it to Canvas. CMS is not required for this problem.

Listing 2: Problem 3C Solution.

```python
def my_divide_by_two(n):

    out = 0
    while n > 1:
        n /= 2
        out += 1

    return out
```

Problem 4 – Linear Search

Given an array `arr = [10, 7, 2, 13, 4, 52, 6, 17, 81, 49]`, implement Linear Search function called `linear_search(arr, num_find)` to search whether the given element (an integer) entered by the user is present within the list or not. If the element is present in list then the program will return its position in the list else it will return "not found". See the example outputs below:

Sample output 1:

```
Enter first number:  52
```

```
Number 52 found at position 6
```

Sample output 2:

```
Enter first number:  81
Number 81 found at position 9
```

Sample output 3:

```
Enter first number:  50
Number 50 not found
```

Submit your solution to CMS.