

LAB MANUAL

Course: DATA STRUCTURES AND ALGORITHMS

Course Code: CSC 101

Instructor: SIR: M. AMEEN CHHAJRO



Submitted By:

STUDENT NAME	STUDENT ID
Zuhaib Dayo	BIT – 22F - 019

(INFORMATION TECHNOLOGY)

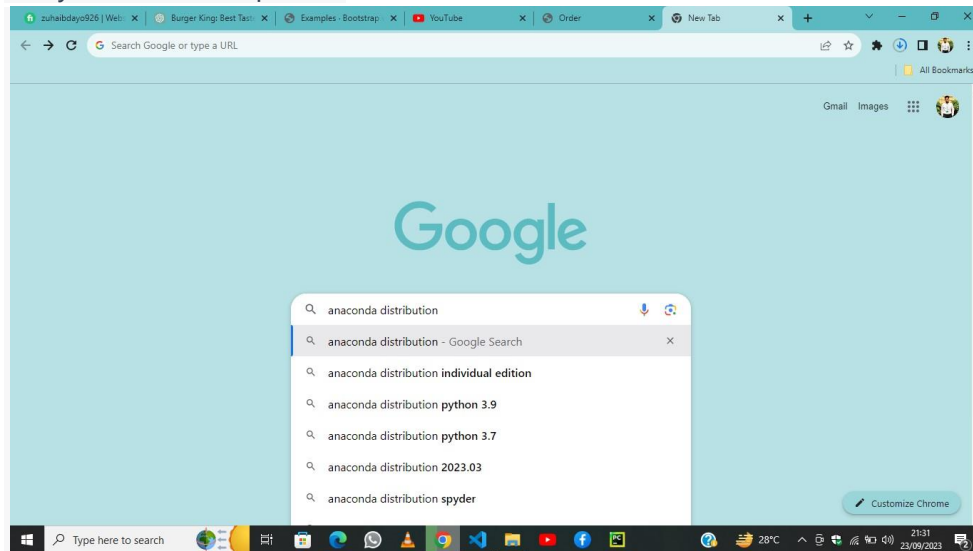
Submitted To:

SIR: M. AMEEN CHHAJRO

Sindh Madrassa tull Islam University
Karachi

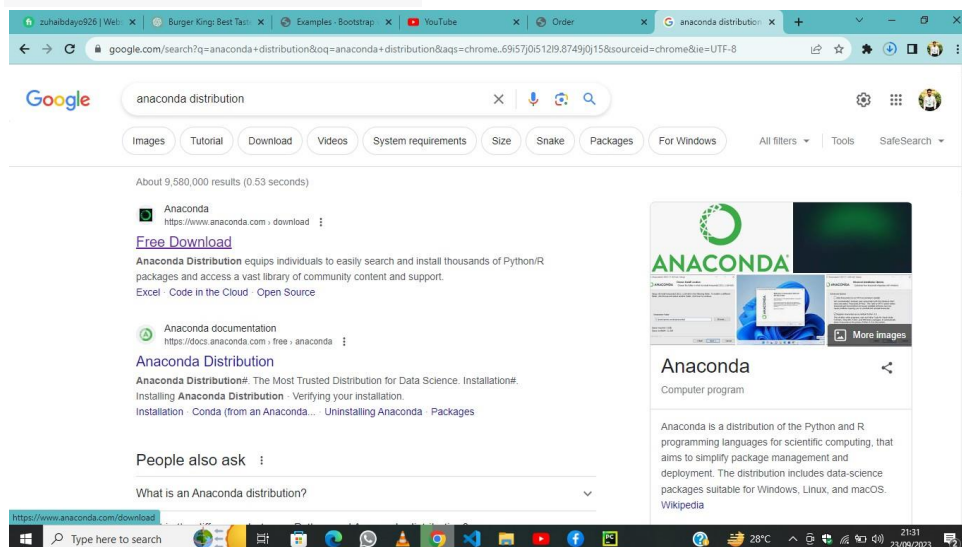
LAB 1

Step 1: Anaconda is a powerful and popular distribution that simplifies the setup and management of data science and machine learning environments by providing a comprehensive package management system and pre-installed libraries essential for data analysis and development.



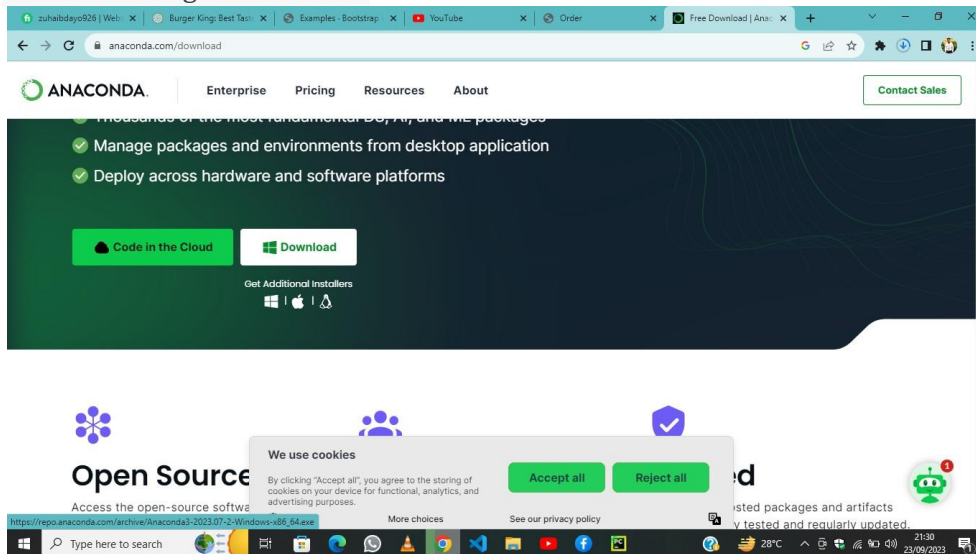
Step 2

Now I click the Anaconda



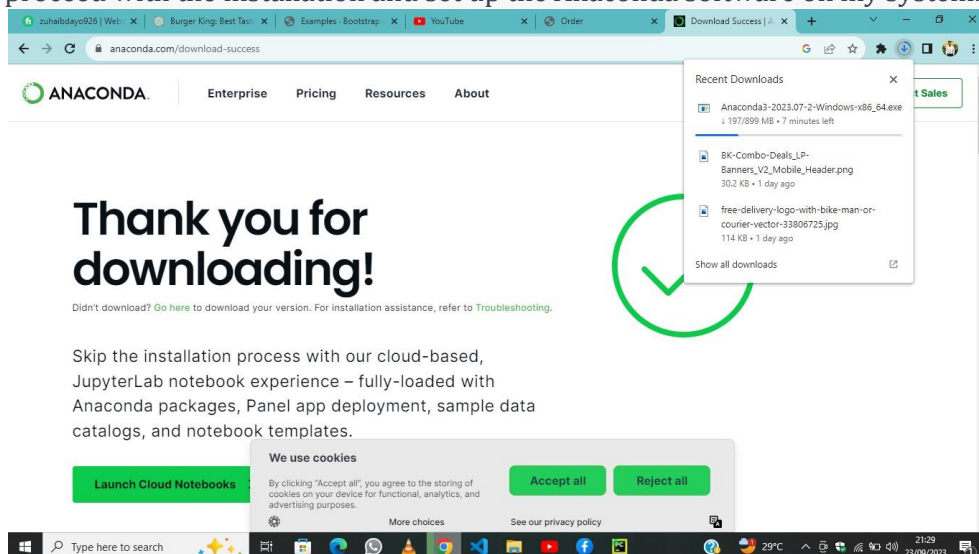
Step 3

I have clicked on the download link for Anaconda on Windows and am currently in the process of downloading the software



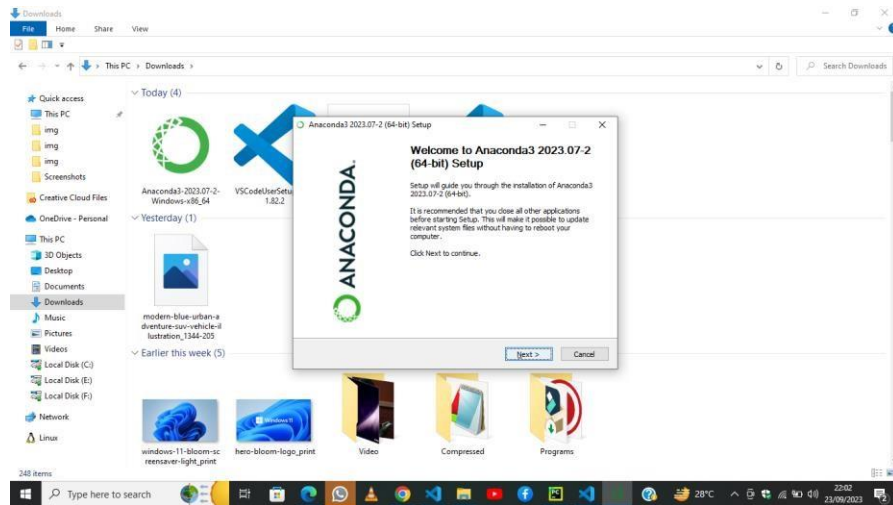
Step 4

The download for Anaconda on Windows has been completed successfully. I am now ready to proceed with the installation and set up the Anaconda software on my system.

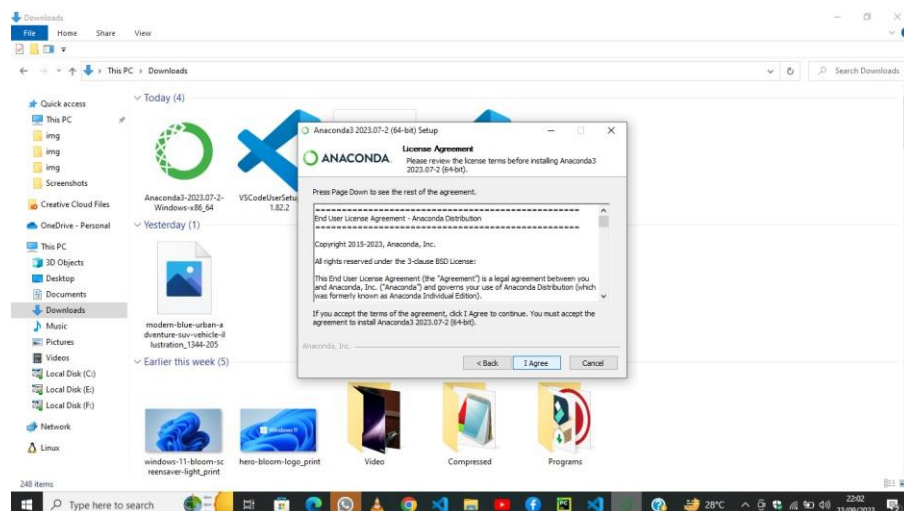


Step 5

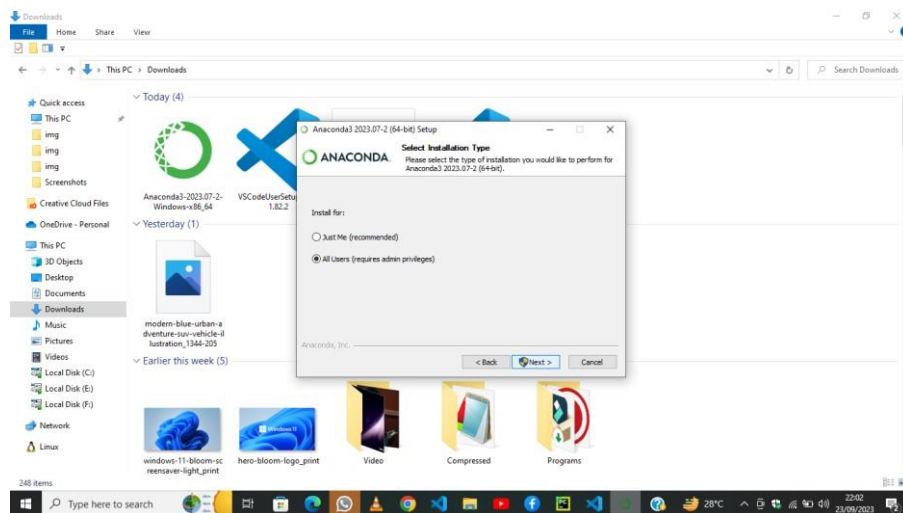
I have completed the download of Anaconda on Windows and am currently in the process of installing the software. I'm excited to set up Anaconda and begin utilizing its powerful features for data science and programming projects



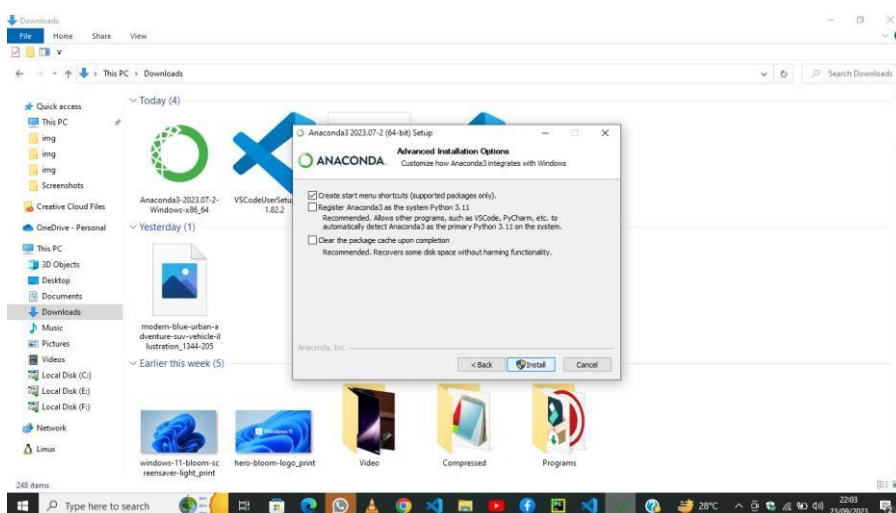
Step 6



Step 7



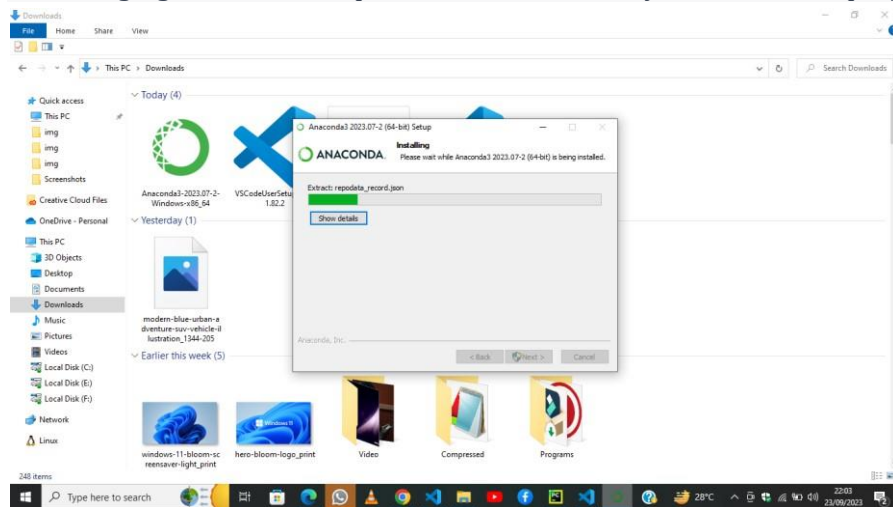
Step 8



Step 9

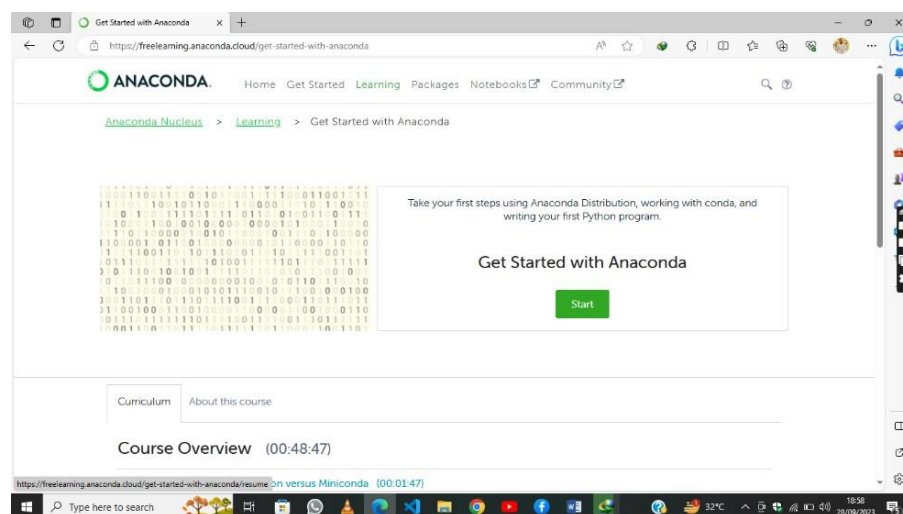
I am currently in the process of configuring Anaconda after successful installation on my Windows system. This involves setting up specific environments, installing necessary packages,

and customizing the settings to suit my data science and programming needs. I'm looking forward to leveraging Anaconda's capabilities to enhance my workflow and project development.



Step 10

I have successfully installed Anaconda on my Windows system. I'm eager to dive into the world of data science and programming, leveraging the powerful tools and libraries that Anaconda provides. This marks the beginning of an exciting journey towards enhancing my skills and undertaking various projects.



LAB 2

Lab2 Tasks

Q1: Perform all arithmetic operations using assignment operators for current semester courses.

ANS:

```
GPA = 100
a = int(input("Enter your marks for ICT:
")) b = int(input("Enter your marks for
Math: "))
c = int(input("Enter your marks for Applied Physics:
")) d = int(input("Enter your marks for PF: ")) e =
int(input("Enter your marks for English: "))
ict = (a / GPA) * 100 math=
(b / GPA) * 100 applied_phy
= (c / GPA) * 100 pf= (d /
GPA) * 100 english = (e /
GPA) * 100
def
calculate_grade(percentage):
if percentage >= 90:
return 'A', 4.0 elif
percentage >= 80:
return 'B', 3.0 elif
percentage >= 70:
return 'C', 2.0 elif
percentage >= 60:
return 'D', 1.0 else:
return 'F', 0.0

grade, gpa = calculate_grade(ict) grade,
gpa = calculate_grade(math) grade, gpa =
calculate_grade(applied_phy) grade, gpa
= calculate_grade(pf) grade, gpa =
calculate_grade(english)

total_grade_points = (gpa+ gpa + gpa+gpa + gpa)
total_subjects = 5 # Total number of subjects
cgpa = total_grade_points / total_subjects

# Print the results
print("Percentage for ICT:", ict, "% - Grade:", grade, " - GPA:", gpa)
print("Percentage for Math:", math, "% - Grade:", grade, " - GPA:", gpa)
print("Percentage for Applied Physics:", applied_phy, "% - Grade:", grade,
" - GPA :", gpa)
print("Percentage for PF:", pf, "% - Grade:", grade, " - GPA:", gpa)
print("Percentage for English:", english, "% - Grade:", grade, " - GPA:",
gpa)
print(" Your CGPA is :", cgpa)
```

OUTPUT:

Enter your marks for ICT: 88

Enter your marks for Math: 99

Enter your marks for Applied Physics: 87

Enter your marks for PF: 98

Enter your marks for English: 67

Percentage for ICT: 88.0 % - Grade: D - GPA: 1.0

Percentage for Math: 99.0 % - Grade: D - GPA: 1.0

Percentage for Applied Physics: 87.0 % - Grade: D - GPA : 1.0

Percentage for PF: 98.0 % - Grade: D - GPA: 1.0

Percentage for English: 67.0 % - Grade: D - GPA: 1.0

Your CGPA is : 1.0

Q2: With the help of arithmetic operation describe which number is even and odd from following 101, 500, 204.

```
ANS: numbers = [101, 500, 204]

for number in numbers:
    if number % 2 == 0:
        print(f"{number} is even.")
    else:
        print(f"{number} is odd.")
```


OUTPUT:

101 is odd.

500 is even.

204 even.

Q3: If you pay 15% tax of your earning that is 30000 write a program in python to calculate the tax you pay

ANS:

```
Salary = 30000
tax = 0.15

tax_paid = Salary * tax

print("Tax paid:", tax_paid)
```

OUTPUT:

Tax paid: 4500.0

Q4: Use the round of function in python and round of any three numbers to 2,3,4 decimal points.

ANS:

```
Value1 = 12.3456789
Value2 = 45.6789123
Value3 = 67.8912345

Index = round(Value1, 2)
Index2= round(Value2, 2)
Index3 = round(Value3, 2)
print("The Value of Value1 rounded to 2 decimal points:", Index) print("The Value of Value2 rounded to 2 decimal points:", Index2) print("The Value of Value3 rounded to 2 decimal points:", Index3)
```

OUTPUT:

The Value of Value1 rounded to 2 decimal points: 12.35

The Value of Value2 rounded to 2 decimal points: 45.68

The Value of Value3 rounded to 2 decimal points: 67.89

Q5: Use type method in python for c = 12.4 and 3000.

ANS:

```
Value1 = 12.4
Value2 = 3000

Value1 = type(Value1)
Value2 = type(Value2)

print(" The Type of Value is :", Value1)
print("The Type of Value is :", Value2)
```

OUTPUT:

The Type of Value is : <class 'float'>

The Type of Value is : <class 'int'>

Q6: Convert 6.0 into 6

ANS:

```
float = 6.0
int = int(float)
print("Original float value:", float)
print("Converted integer value:", int)
```

OUTPUT:

Original float value: 6.0

Converted integer value: 6

Lab Task 3

Q: 1) Use string slicing to grab the word 'thin' from inside 'thinktank'

Ans

```
original_string = 'thinktank'
extracted_word = original_string[0:4]
print("Extracted word:", extracted_word)
```

OutPut:

Extracted word: thin

Q :2) Use indexing value of your class id to grab character from the string

```
my_string = "abcdefghijklmnopqrstuvwxyz"
index= my_string[0][::-1][9:]
print("Character at index :", index)
```

Output:

Character at index 2: zyxwvuts

Q:3 Store ABC in a variable and perform following operations

Q:4 Print all strings

Q:5 Grab alternate letters from ABC

Q:6 Reverse your string

Q:7 Grab 8 letters from the string and start from the letter which is the first letter of your name.

Q:8 Write the code in python to generate following output using string logic

Q: 9 abbcccddeeeeee

```
alphabet = "ABCDE"

print("All strings:", alphabet)

alternate_letters = alphabet[::2]
print("Alternate letters from ABC:", alternate_letters)

reversed_alphabet = alphabet[::-1] print("Reversed string:",
reversed_alphabet)

grabbed_letters = alphabet[:8]
print("Grabbed 8 letters starting from the first letter:", grabbed_letters)

desired_output = ''.join([letter * i for i, letter in enumerate(alphabet,
start=1)])
print("Generated output:", desired_output)
```

Output:

All strings: ABCDE

Alternate letters from ABC: ACE

Reversed string: EDCBA

Grabbed 8 letters starting from the first letter: ABCDE

Generated output: ABBCCDDDDDEEEEEE

Q:10 Apply the .format method on semester courses

Name of one course

Name of three courses

Ans:

```
def display_courses(course_name, *other_courses):
    print("Name of one course: {}".format(course_name))
    print("Name of three courses: {}, {}, {}".format(course_name,
*other_courses))

# Example course names
course1 = "Computer Science"
course2 = "Mathematics"
course3 = "History"

# Display the course names
display_courses(course1, course2, course3)
```

Output:

Name of one course: Computer Science

Name of three courses: Computer Science, Mathematics, History

Q:11 Write intro about you , convert into list structure and then grab all words excluding 1st and last word, from your introduction.

Ans:

```
introduction_list = [  
    "I", "am", "Zohaib,", "a", "language", "model", "developed", "by",  
    "Anonymous",  
]  
index = introduction_list[1:-1]  
print("Words excluding the first and last words:")  
print(index)
```

OutPut:

Words excluding the first and last words:

['am', 'Zohaib,', 'a', 'language', 'model', 'developed', 'by']

Q:12 Name of six courses randomly and arrange them in alphabetical order using indexing in .format method

Ans:

```
import random  
course_names = [  
    "Biology",  
    "Mathematics",  
    "History",  
    "Computer Science",  
    "Physics",  
    "English Literature"  
]  
sorted_course_names = sorted(course_names)  
formatted_courses = "{0}. {1}\n{2}. {3}\n{4}. {5}\n{6}.  
{7}\n{8}.  
{9}\n{10}. {11}".format(  
    1, sorted_course_names[0],  
    2, sorted_course_names[1],  
    3, sorted_course_names[2],  
    4, sorted_course_names[3],  
    5, sorted_course_names[4],  
    6, sorted_course_names[5]  
)  
print("Courses in alphabetical order:\n", formatted_courses)
```

Output:

Courses in alphabetical order:

1. Biology
2. Computer Science
3. English Literature
4. History
5. Mathematics
6. Physics

Q: 13 Assign the key words to all semester courses and call them using. format method.

Ans:

```
def format_courses_with_keywords(courses):
    formatted_courses = ""
    for i, course in enumerate(courses, start=1):
        formatted_courses += "{0}. {1} - Keyword: {2}\n".format(i, course["name"], course["keyword"])
    return formatted_courses

# Define the courses and their keywords
semester_courses = [
    {"name": "Mathematics", "keyword": "Algebra"},
    {"name": "Physics", "keyword": "Newtonian Mechanics"},
    {"name": "English Literature", "keyword": "Shakespearean Literature"},
    {"name": "History", "keyword": "World War II"},
    {"name": "Computer Science", "keyword": "Data Structures"},
    {"name": "Biology", "keyword": "Genetics"}
]

# Call the function to format and display the courses with keywords
formatted_courses_with_keywords = format_courses_with_keywords(semester_courses)
print("Semester Courses with Keywords:\n", formatted_courses_with_keywords)
```

Output:

Semester Courses with Keywords:

1. Mathematics - Keyword: Algebra
2. Physics - Keyword: Newtonian Mechanics
3. English Literature - Keyword: Shakespearean Literature
4. History - Keyword: World War II
5. Computer Science - Keyword: Data Structures
6. Biology - Keyword: Genetics

Q:14 Apply float formatting method on following numbers with precision of 4 a.

200.340982589

b. 40/66611

c. 1.534854395

Ans:

```
a = 200.340982589 b = 40/66611 c = 1.534854395
a = '{:.4f}'.format(a) b = '{:.4f}'.format(b) c = '{:.4f}'.format(c)
print("Formatted number a:", a) print("Formatted number b:", b) print("Formatted number c:", c)
```

Output:

Formatted number a: 200.3410

Formatted number b: 0.0006

Formatted number c: 1.5349

LAB :4

1. Create a list of your current semester courses
2. Add course Python in the end of semester courses list
3. Create an empty list with the name my course
4. Extend my course list with semester courses list
5. Insert "programming" at index 2 in my course list
6. Remove element at index 1

7. Add "Math1" to index 1

8. Sort all courses in ascending order

```
# lab task #4

# 1.    Create a list of your current semester courses

a = "DSA"
b = "Linear"
c = "Distric math"
d = "Software Requirement"
e = "pp"
f = "Connect with Computer"

print("This is My course in this semester is: "+ a +" In python")
print("This is My course in this semester is:", b)
print("This is My course in this semester is:", c)
print("This is My course in this semester is:", d)
print("This is My course in this semester is:", e)
print("This is My course in this semester is:", f)

# 2.    Add course Python in the end of semester courses list

g = "Python"
add = g
print("This is Add Course in this semester is:",add)

# 3.    Create an empty list with the name my_course

my_course = []
print("This is Empty List :",my_course)

# 4.    Extend my_course list with semester courses list

my_course = a,b,c,d,e,f
print("now This is list and This is Connect With Other Program :",
my_course )

# 5.    Insert "programming" at index 2 in the my_course list
courses = [a, b, c, d, e, f]
index = courses.index(b)
courses[index] = "Programming"

print("Updated semester_courses list:", courses)

# 6.    Remove element at index 1

courses = [a, b, c, d, e, f]
index = courses.pop(0)
print("Cut the course in the semester_courses list:", courses)

# 7.    Add "Math1" to index 1

courses = [ b, c, d, e, f]
courses.insert(0, "Math")
print("Cut the course in the semester_courses list:", courses)
```

```
# 8.      Sort all courses in ascending order

courses = [a, b, c, d, e, f]
courses.sort()
print("Arrangement the Courses Of semester_courses list:", courses)
```

OUTPUT:

This is My course in this semester is: DSA In python

This is My course in this semester is: Linear

This is My course in this semester is: District math

This is My course in this semester is: Software Requirement

This is My course in this semester is: pp

This is My course in this semester is: Connect with Computer

This is Add Course in this semester is: Python

This is Empty List: []

now This is list and This is Connect with Other Program: ('DSA', 'Linear', 'District math', 'Software Requirement', 'pp', 'Connect with Computer')

Updated semester courses list: ['DSA', 'Programming', 'District math', 'Software Requirement', 'pp', 'Connect with Computer']

Cut the course in the semester courses list: ['Linear', 'District math', 'Software Requirement', 'pp', 'Connect with Computer']

Cut the course in the semester courses list: ['Math', 'Linear', 'District math', 'Software Requirement', 'pp', 'Connect with Computer']

Arrangement the Courses Of semester courses list: ['Connect with Computer', 'DSA', 'District math', 'Linear', 'Software Requirement', 'pp']