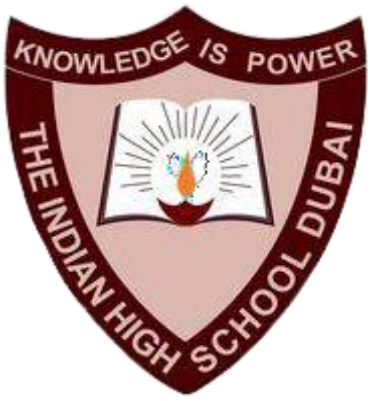
THE INDIAN HIGH SCHOOL - DUBAI



SUBJECT

Comp. Sci Journal
2025-26

Name: Abyaz Javid
Roll no.: 4

CERTIFICATE

This is to certify that the work in this journal is the bonafide work of

Master _____

Class _____ *Div* _____

recorded in the school lab during the academic year

20xx-20xx

Date: _____

Teacher in charge: _____

ACKNOWLEDGEMENT

I would like to take this opportunity to thank the Central Board of Secondary Education (CBSE) and The Indian High School-Dubai, for granting me the opportunity to deepen my knowledge in my favorite subject, Computer Science

I would also like to thank my teacher Mrs. Swapnil Verma for guiding me and sharing her wide variety of knowledge

INDEX

Program 1

Aim: To write a function that takes 'n' number of integers and count the number of odd and even numbers.

Modules used: N/A

Data types used: Int

Script:

```
def count(*a) -> None:
    o, e = 0, 0
    for i in a:
        (e := e + 1) if i % 2 == 0 else (o := o + 1)
    print(f"No. of odd elements: {o}\nNo. of even elements: {e}")

count(1,2,3,4,5,6,7,8)
```

Output:

```
No. of odd elements: 4
No. of even elements: 4
>>>
```

Program 2

Aim: To write a function to find the sum of the series: $1 + x^2 + x^3 + \cdots + x^n$

Modules used: N/A

Data types used: Int

Script:

```
def series(x: int, n: int) -> None:
    s = 1
    for i in range(n): s += x**(i+1)
    print(f"Sum of the series is: {s}")

series(2, 16)
```

Output:

```
    Sum of the series is: 131071
>>>
```

Program 3

Aim: To write a function to find the factorial of a number without taking an argument

Modules used: N/A

Data types used: Int

Script:

```
n = int(input(">>> "))
def fact() -> int:
    global n
    return n * (n := n - 1, fact())[1] if n > 1 else 1

print(f"Factorial of {n} is: {fact()}")
```

Output:

```
>>> 5
Factorial of 5 is: 120
```

Program 4

Aim: To write a function that takes a list of strings and return the emails that contain the substring "@cmail"

Modules used: N/A

Data types used: String, List

Script:

```
r = input(">>> ").split()
def validMailID(l: list) -> list:
    return [i for i in l if "@cmail" in i]

print(f"Valid email IDs are:\n - " + "\n - ".join(validMailID(r)))
```

Output:

```
>>> asdf@cmail.com fdsa@cmail.com asdf@gmail.com swag@yahoo.gov
Valid email IDs are:
 - asdf@cmail.com
 - fdsa@cmail.com
```

Program 5

Aim: To write a function that takes a list of strings and returns the strings that are longer than 5 characters.

Modules used: N/A

Data types used: String, List

Script:

```
r = input(">>> ").split()
def longWords(l: list):
    return [i for i in l if len(i) > 5]

print(f"Words longer than 5 letters are:\n - " + "\n - ".join(longWords(r)))
```

Output:

```
>>> word1 antidisestablishmentarianism word2 wee asdf fdsa
Words longer than 5 letters are:
 - antidisestablishmentarianism
```

Program 6

Aim: To write a menu driven program to find odd/even numbers and prime numbers.

Modules used: math

Data types used: Int, Bool

Script:

```
import math

print("""
    #--------------------#
    |       NUMBERS      |
    | 1. even/odd        |
    | 2. prime/consonant |
    | 3. exit            |
    #--------------------#""")

def evenity(n: int) -> bool:
    return True if n % 2 == 0 else False

def primality(n: int) -> bool:
    return False if n <= 1 else (True if n == 2 else (False if n % 2 == 0 else all(n % i != 0 for i in range(3, int(math.sqrt(n)) + 1, 2))))


while True:
    o = int(input(">>> "))
    if o == 1:
        print(evenity(int(input("n: "))))
    elif o == 2:
        print(primality(int(input("n: "))))
    elif o == 3:
        break
    else:
        print("Invalid option selected")
```

Output:

```
    #--------------------#
    |       NUMBERS      |
    | 1. even/odd        |
    | 2. prime/consonant |
    | 3. exit            |
    #--------------------#
>>> 1
n: 43
False
>>> 2
n: 5
True
>>> 4
Invalid option selected
>>> 3
```

Program 7

Aim: To write a function that returns the greater of two numbers.

Modules used: N/A

Data types used: Tuple, Float

Script:

```
a, b = tuple([*map(float, input(">>> ").split())])
def findBig() -> float:
    global a, b
    return a if a > b else b

print(f"Bigger number: {findBig()}")
```

Output:

```
>>> 1 3
Bigger number: 3.0
```

Program 8

Aim: To write a function that takes a list and moves all the elements divisible by 5 to the end of the list.

Modules used: N/A

Data types used: List, Int

Script:

```
x = [*map(int, input(">>> ").split())]
def move(l: list) -> None:
    l[:] = [x for x in l if x % 5] + [x for x in l if x % 5 == 0]
    print(f"Ordered list is: {l}")

move(x)
```

Output:

```
>>> 1 2 5 4 6 58 65 2350 15
Ordered list is: [1, 2, 4, 6, 58, 5, 65, 2350, 15]
```

Program 9

Aim: Given a dictionary containing information about vehicles, display the vehicles that were released in 2020 and order the dict in alphabetical order by brand name.

Modules used: N/A

Data types used: Dict, Int, List, String

Script:

```
vehicle = {
    "car1": ["Toyota", "Camry", 2020, 25_000],
    "car2": ["Ford", "Explorer", 2019, 32_000],
    "car3": ["Chevy", "Silverado", 2021, 40_000],
    "car4.5": ["Honda", "Civic", 2020, 22_000],
    "car5": ["anotherRealCarBrand", "Model nine", 2023, 45_000]
}

def _2020(D: dict):
    print(f"No. of vehicles released in 2020: {len([c for c in D if c[2] == 2020])}")

def sort(cars_dict):
    sorted_cars = sorted(cars_dict.items(), key=lambda x: x[1][0].lower())
    for key, value in sorted_cars:
        print(f"{key}: {value}")

_2020(vehicle)
print("\nSorted dict:")
sort(vehicle)
```

Output:

```
No. of vehicles released in 2020: 0

Sorted dict:
car5: ['anotherRealCarBrand', 'Model nine', 2023, 45000]
car3: ['Chevy', 'Silverado', 2021, 40000]
car2: ['Ford', 'Explorer', 2019, 32000]
car4.5: ['Honda', 'Civic', 2020, 22000]
car1: ['Toyota', 'Camry', 2020, 25000]
```

Program 10

Aim: To write a function that takes a tuple and returns the indices of the non-zero elements.

Modules used: N/A

Data types used: Int, Tuple, List

Script:

```
t = tuple([*map(int, input(">>> ").split())])
def indexTuple(t: tuple) -> list:
    return [i for i, v in enumerate(t) if v != 0]

print(f"Non zero indices are: {indexTuple(t)}")
```

Output:

```
>>> 1 0 25 03 64 00 5
Non zero indices are: [0, 2, 3, 4, 6]
```

Program 11

Aim: To write a function to count the number of vowels in user input.

Modules used: N/A

Data types used: String

Script:

```
r = input(">>> ")
def vowelCount() -> None:
    global r
    print(f"No. of vowels: {sum(i in "AEIOUaeiou" for i in r)}")

vowelCount()
```

Output:

```
>>> hello my name is star.stalker9160
No. of vowels: 8
```