```python
# To find the downstream depth of open channel
# Given Data
Q= float (input("Enter the value of Discharge 4.8 "))
T= int(input("Enter the value of top width 2 "))
g=float(input("Enter the value of acceleration due to Gravity 9.81"))
y1 = float(input( enter the value of upstream depth 1.6 "))
Z= float(input("Enter the value of hump 0.1 "))
# Discharge per meter width
q=Q/T
print ("The value of discharge per meter width is ", q)
# Area Calculation
A1= T*y1
print ("The value of upstream area is ", A1)
# Calculation of Froude Number
Fr1 = (Q*Q*T/(g*A1* A1 *A1)) ** 0.5
print ("The value of Froude number is ", Fr1)
if Fr1>1:
    pass # Replace with actual code
    print("The flow is Super critical flow")
else:
    print("The flow is Sub Critical Flow")
#Upstream Energy
E1 = y1 + (Q*Q)/(2 *g*A1 *A1)
print ("The value of Energy at Initial Section is ", E1)
# Downstream Energy
E2 = E1 -Z
print ("The value of downstream Energy E2 is", E2)
# Critical Depth
# You need to define the values for 'a' and 'e' based on your problem.
# For example
a = 1.0  # Replace with the actual value of 'a'
e = 1.0  # Replace with the actual value of 'e'
yc = (q*a/e)**(0.333)
print ("The Value of critical depth is ", yc)
# Note: In Python, a tuple is defined using parenthesis. Using a comma creates a tuple.
# To perform multiplication, use the '*' operator
Ec = 1.5*yc
print ("The value of critical Energy is", Ec)
if Ec>E2:
    print ("Checking Condition")
else:
    print ("SAFE")
# Calculation of Zmax
# You need to define the value for 'E1' based on your problem.
# For example
E1 = 1.0 # Replace with the actual value of 'E1'
Zmax =E1- Ec
print ("The value of maximum hump is ", Zmax)
```

```
Enter the value of Discharge 4.8 4.8
Enter the value of top width 22
Enter the value of acceleration due to Gravity 9.819.81
enter the value of upstream depth 1.61.6
Enter the Value of hump 0.10.1
The value of discharge per meter width is  2.4
The value of upstream area is  3.2
The value of Froude number is  0.3756781119266054
The flow is Sub Critical Flow
The value of Energy at initial Section is  1.7146788990825689
The value of downstream Energy E2 is  1.6146788990825689
The Value of critical depth is  1.3386264295597894
The value of critical Energy is 2.008240244339684
Checking Condition
The value of maximum hump is  -0.208240244339684
```

```python
# To find the downstream depth of open channel
# Given Data
Q= float(input("Enter the value of Discharge 15"))
B1 = float(input("Enter the value of width at upstream 1.5 "))
B2 = float(input("Enter the value of width at downstream 2.5 "))
g= float(input("Enter the value of acceleration due to Gravity 9.81"))
y1= float(input("enter the value of upstream depth 2"))
# Discharge per meter width
q1=Q/B1
q2= Q/B2
print ("The value of discharge per meter width is ", q1) # Changed q1 to q1
```

```
print ("The value of discharge per meter width is:", q1)
# A1 calculation
A1 = B1*y1 # Changed y1 to y1 as y1 was not defined
print ("The value of upstream area is:", A1)
# Calculation of Froude Number
Fr1 = ((Q**2*B1)/(g*(A1**3)))**0.5
print ("The value of Froude number is:", Fr1)
if Fr1>1:
    print("The flow is Super Critical flow") # Indent this line to be part of the 'if' block
else:
    print("The flow is sub critical flow")
# Upstream energy
e = 1.0026 [illegible] # Define e here or import it from a library if it represents a physical constant
E1 = y1 + (Q**2)/(2*g*(A1**2)) # Use y1 instead of y1 if it's a different variable
print ("The value of Energy at initial section is:", E1) # Correct the variable name to 'E1'.
# [illegible] Remove or comment out this invalid line.
B2min = ((27*Q*Q)/(8*g*(E1**3)))**0.5
print ("The value of minimum width to be kept to avoid checking is:", B2min)
if B2min > B2:
    print ("Checking condition") # Indent this line
else:
    print ("SAFE") # Indent this line to be part of the 'else' block
# Critical Depth
e = 0.8 # [illegible] value
yc = ((Q**2)/(B2**2*g))**(1/3)
print ("The value of critical depth is:", yc)
Ec = 1.5*yc
```

```
Enter the value of Discharge 15
Enter the value of width at upstream3.5
Enter the value of width at downstream2.5
Enter the value of acceleration due to gravity9.81
Enter the value of upstream depth 2
The value of discharge per meter width is: 4.285714285714286
The value of discharge per meter width is: 6.0
The value of upstream area is: 7.0
The value of Froude number is: 0.4[illegible]
The flow is Sub critical flow
The value of Energy at initial section is: 2.[illegible]
The value of minimum width to be kept to avoid checking is: [illegible]
Checking Condition
The value of critical depth is: 1.5[illegible]
The value of critical energy is: 2.3[illegible]
```

```
#Design of Efficient channel section
Q= float(input("Enter the value of Discharge 100"))
n= float(input("Enter the value of Rugosity coefficient 0.015"))
So= float(input("Enter the value of bed slope 0.0004"))
g= float(input("Enter the value of acceleration due to gravity 9.81"))
#Manning's formula
#Q = (AR^2/3 S^1/2)/n
yn = ((Q*n*So** 0.5)/(1.73)) ** (3/8)
print ("The value of yn is:", yn)
#to encounter the effect of free board
yn1= 1.1*yn
print ("The value of yn1 is:", yn1)
# Cross Sectional Area
A = 1.732 * yn * yn1
print ("The cross sectional Area is:", A)
# Top width
t= 4* yn/1.732
print ("The value of top width is:", t)
# Bottom width
b= 2 * yn/1.732
print ("The value of bottom width is:", b)
Fr= ((Q*Q*t)/(g*A*A*A)) ** 0.5 # fixed the Fr calculation
print ("The value of Froude number is:", Fr)
if Fr>1:
    print("The flow is Super Critical flow") # Indented this line
else:
    print("The flow is Sub Critical flow")
```

```
Enter the value of Discharge 100100
Enter the value of Rugosity coefficient 0.0150.015
Enter the value of bed slope 0.00040.0004
Enter the value of acceleration due to gravity 9.819.81
The value of yn is: [illegible]
The value of yn1 is: [illegible]
```

```
The cross sectional Area is 1271.6576815774754
The value of top width is 5?.66600400635711
The value of bottom width is 8
The value of froude number is 1.4788266440181834e-05
The flow is sub critical flow
```

THOMAS KHOMANE (21CV044)