**Name:** Ali Zain

**Sec**: 1-M **Nu-ID:** 21k-4653

**Problem: 1**

Write appropriate data types for each of the following items. Provide one line justification for

your choice. Solve this problem on a paper sheet.

|  |  |
| --- | --- |
| a) Age of a person (in years) | **Unsigned Integer**  Ages are in digits form and positive. |
| b) Speed of light | **Integer**  Speed of light is 3x108 or 299792458. |
| c) Gender | **String/Char**  Gender is either a strong or char. |
| d) Coordinates of a point | **Float**  Coordinates helps in graph plotting each pixel is vital so float. |
| e) Factorial of a number | **Unsigned Long Long Integer**  Factorials are always positive and can be too long so unsigned Long Long. |
| f) The number of plants in a region | **Long float/double or Integer**  if it’s number of total plants over certain area in sqfts then float else Integer |
| g) Mass of an electron | **Long double**  Many decimal places so double |

**Problem: 2**

Predict the output of the following expressions. Also, justify your answer by solving the

expressions manually. Solve this problem on a paper sheet.

a) If d is a float, then the operation d=2/7 would store **0.000000** in d.

**2/7 = 0.285 but 2 and 7 are integer they will neglect value after decimal so d= 0.0000000**

b) If x is an integer, then the expression x=-7%2-8 would evaluate to **-9**.

**-7%2 = -1**

**-1-8 = -9**

c) If c=0, then the expression 5&&c!= 8||!c would evaluate to **True**.

**5&&c = False 8||!c = True**

**False != True**

d) The expression a=b=c=3+4 would evaluate to: a=**7**, b=**7**, c=**7**.

e) The expression y=z=-3%-8/2+7 would evaluate to: y=**6**, z=**6**.

-**3%-8 = -3 🡪 -3/2 = -1 🡪 -1+7 = 6**

**Problem: 3** (ImageRef : P3.png)

**Shape, polygon

Description automatically generated**

**Problem: 4**

Start

Get “Enter any 4 digit Roll number ?” into number

If (number < 0 or number/10000 != 0)

Then display “Invalid input! Execute program again and then enter a valid integer”

Else

Then

Get “Enter the last character of your rollno” into rollno

While (number != 0)

testcheck = number % 10

if (testcheck == rollno)

then flag++

number /=10

Display “(rollno) occurred (flag) times”

End

**Problem: 5**

|  |  |  |
| --- | --- | --- |
| s.no | Program | Output |
| a | main( ){  int i = 4, j = -1, k = 0, w, x, y, z ;  w = i || j || k ;  x = i && j && k ;  y = i || j && k ;  z = i && j || k ;  printf ( "\nw = %d x = %d y = %d z = %d", w, x, y, z ) ;  } | **Logic:** compiler will give more priority to and (&&) than or (||).  w = 1 (True)  x = 0 (False)  y = 1 (True)  z = 1 (True) |
| b | main(){  int i = 2, j = 3, k, l ;  float a, b ;  k = i / j \* j ;  l = j / i \* i ;  a = i / j \* j ;  b = j / i \* i ;  printf( "%d %d %f %f", k, l, a, b ) ;  } | **Integer values:**  **k = 2/3 \* 3 = 0 \* 3 = 0**  *2/3 is originally 0.6667 but because it is integer, we will take 0 (neglect .667)*  **I = 3/2 \* 2 = 1 \* 2 = 2**  *3/2 is originally 1.5 but because it is integer, we will take 1(neglect .5)*  **Floating values**  **a = 2/3 \* 3 = 0 \* 3 = 0.000000**  *2/3 is originally 0.6667 but because it is integer, we will take 0 (neglect .667)*  But a is float so 0.00000 similarly b will be 2.00000  **b = 3/2 \* 2 = 1 \* 2 = 2.000000** |
| c. | main( )  {  int m, n ;  m = -3%2!=!3 ;  n = -3\*10.5/2 – 3 ;  printf ( "a = %d\n b = %f", m, n ) ;  printf ( "a = %d\n b = %d", m, n ) ;  } | **Logic :**  m = -1 != !3 = 1  -1 is not equal to null hence True (1)  but for n  Replace %f with %d to get  -18  Also, n is declared as integer not float.  **Final outputs :**  a = 1 and b = -18 |

**Problem: 7**

Start

Get float x1, x2, x3, x4, y1, y2, y3, y4

If ((y2-y1/x2-x1)) == ((y3-y2/x3-x2))

Then display “The given points fall on a straight line!”

Else

Then display “The given points do not fall on a straight line!”

End

**Problem: 9**

Start

Get last 2 digit of roll number into variable number

If number < 0 or number / 100 !=0

Then display “Invalid input. Please try again.”

Else

Then

While (number > 0 )

rem = number % 2

binaryNum += rem \* 10 \*\* i

number /=2

increment i by one

Display “Binary Number of (number) is (binaryNum)

While (binaryNum != 0)

dec = binaryNum % 10

decimalNum += dec \* base

binaryNum /= 10

base \*= 2

Display “Decimal Number of (binaryNum) is (decimalNum)”

If decimalNum == number

Then display “Hurrah !! your program is correct.”

Else

Then display “There might be an error !!”

End

**Problem: 10**

Text, letter

Description automatically generated