INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSC-101 PROGRAMMING WITH C AND C++
Autumn 2023

Quiz 2 (solutions) Total Marks: 10 (including 2 marks for attendance) Time - 50 mins.

Note: Full justification and arguments are required for all questions. Simple answers without proper reasoning will not carry any weight.

1. Examine the code below:

```c
#include<stdio.h>

int main() {
    int x = 3, y = 6, z = 9;
    int *px = &x, *py = &y, *pz = &z;

    if (*px * 2 > y) {
        if (*py + *pz == *px * 5)
            printf("%d %d %d", ++(*px), (*py) -= 2, (*pz)++);
        else
            printf("%d %d %d", (*px)--, ++(*py), (*pz) -= 3);
    } else if (*px + *pz < *py * 2) {
        printf("%d %d %d", *px += 3, (*py)--, ++(*pz));
    } else {
        printf("%d %d %d", (*px)--, *py += 4, *pz -= 2);
    }

    return 0;
}

```

What will be the output of the above code? [**2 Marks**]

Sol.

The values of x, y, and z are initialized to 3, 6, and 9, respectively. We also have pointers px, py, and pz that point to x, y, and z respectively.

Now, let's check the conditions:

Here, *px is 3 (since px points to x), so 3 * 2 = 6. Checking if 6 is greater than y (6), we find it is not. So, this condition is false.

We then move to the next condition:

Here, *px is 3, *pz is 9, and *py is 6. So, 3 + 9 = 12 and 6 * 2 = 12. Checking if 12 is less than 12, we find it is not. This condition is also false.

This means the final else block will be executed:

Here:

(*px)- - : Decrease the value of x by 1. So, x becomes 2 but the value passed on is 3 as it is post-decrement operator.

*py += 4: Increase the value of y by 4. So, y becomes 10.

*pz -= 2: Decrease the value of z by 2. So, z becomes 7.

The printf statement will therefore print: 3 10 7

2. Consider the following code:

```c
#include <stdio.h>

typedef struct {
    int a;
    int b;
} Pair;

int main() {
    Pair arr[5] = {{1, 2}, {3, 4}, {5, 6}, {7, 8}, {9, 10}};
    int i = 0, j = 0;

    while (i < 5) {
        printf("%d+%d ", arr[i].a, arr[i].b);
        j += arr[i].b;
        i += j/5;
    }

    return 0;
}

```

If $(a_n)_1^N$ represents the sequential output pairs (in the format 'a+b') of the above script, what will be the value of $\frac{a_1.b_N + a_N.b_1}{a_{\lfloor \frac{N}{2} \rfloor}.b_{\lfloor \frac{N}{2} \rfloor + 1}}$? Here, a_n and b_n represent the first and second integers of the n-th pair output, respectively. And $\lfloor . \rfloor$ denotes the greatest integer function. **[3 Marks]**

Sol.

1. First Iteration:

‘i = 0’, ‘j = 0’

‘printf’ prints: ‘1+2’

‘j’ becomes ‘2’ after ‘j += arr[i].b;’

‘i’ remains ‘0’ after ‘i += j/5;’ because ‘2/5’ in integer division is ‘0’.

2. Second Iteration:

‘i = 0’, ‘j = 2’

‘printf’ prints: ‘1+2’ again

‘j’ becomes ‘4’ after ‘j += arr[i].b;’

‘’ remains ‘’ after ‘i += j/5;’ because ‘4/5’ in integer division is ‘0’.

3. Third Iteration:

‘i = 0’, ‘j = 4’

‘printf’ prints: ‘1+2’ again

‘j’ becomes ‘6’ after ‘j += arr[i].b;’

‘i’ becomes ‘1’ after ‘i += j/5;’ because ‘6/5’ in integer division is ‘1’.

4. Fourth Iteration:

‘i = 1’, ‘j = 6’

‘printf’ prints: ‘3+4’

‘j’ becomes ‘10’ after ‘j += arr[i].b;’

‘i’ becomes ‘3’ after ‘i += j/5;’ because ‘10/5’ in integer division is ‘2’.

5. Fifth Iteration:

‘i = 3’, ‘j = 10’

‘printf’ prints: ‘7+8’

‘j’ becomes ‘18’ after ‘j += arr[i].b;’

‘i’ becomes ‘6’ after ‘i += j/5;’ because ‘18/5’ in integer division is ‘3’.

Now, since ‘i’ is 6 (greater than 5), the loop exits.

Final Output:

1+2 1+2 1+2 3+4 7+8

Ergo, the value of $\frac{a_1.b_N + a_N.b_1}{a_{\lfloor \frac{N}{2} \rfloor}.b_{\lfloor \frac{N}{2} \rfloor + 1}}$ is 11.

3. Analyze the following code snippet:

```
#include <iostream>
using namespace std;

int main() {
    int arr[] = {2, 3, 4, 5};
    int &ref = arr[1];
    ref *= 2;

    for(int i = 0; i < 4; i++) {
        cout << arr[i] << "-";
    }

    while(ref -= 2) {
        cout << "In the quiet hum, dynamics play on ... ";
    }

    return 0;
}

```

If n represents the number of characters in the output of the above code, s denotes the sum of all digits in the output, and t represents the number of times the text "In the quiet hum, dynamics play on ... " is printed, find the value of $\left\lceil \frac{n}{s+t} \right\rceil$. Here, $\lceil . \rceil$ denotes the least integer function. **[3 Marks]**

Sol.

The 'for' loop iterates over each element of arr. It uses a reference x to directly access the elements.

For each iteration of the for loop, it prints the current element followed by a "-". This will output the entire array in the format: 2-6-4-5-

The 'while' loop decreases the value ref references by 2 each time it executes and checks if the resulting value is non-zero to continue the loop. Remember, ref refers to arr[1], which currently holds the value 6.

On the first iteration, ref (or arr[1]) is decreased by 2 to become 4. Since 4 is non-zero, the loop body is executed.

On the second iteration, ref becomes 2. Still non-zero, so the loop body is executed again.

On the third iteration, ref becomes 0. Now that it's zero, the loop terminates and won't execute the body again.

Ergo, the output is

```
2-6-4-5-In the quiet hum, dynamics play on ... In the quiet hum, dynamics play on ...
```

1. n (number of characters in the output): '2-6-4-5-': This has 8 characters.

'In the quiet hum, dynamics play on ... ': This has 39 characters (including spaces).

Since this text is printed twice: 39 characters $\times$ 2 = 78 characters.

Summing up: $n = 8 + 78 = 86$ characters.

2. s (sum of all digits in the output):

The digits in the output are: 2, 6, 4, and 5.

Summing up: $s = 2 + 6 + 4 + 5 = 17$.

3. t (number of times the text "In the quiet hum, dynamics play on ... " is printed):

The text is printed twice, so $t = 2$.

Given:

$$\left\lceil \frac{n}{s+t} \right\rceil$$

Substituting the values:

$$\left\lceil \frac{86}{17+2} \right\rceil = \left\lceil \frac{86}{19} \right\rceil = \lceil 4.5263 \rceil$$

Using the ceiling function, the value is rounded up to the nearest integer, so the final answer is:

$$\lceil 4.5263 \rceil = 5$$