

1. Evaluate each of the expressions in the table given below. Write the result in the space provided. [5×2]

Solution:

Expression	Result
int i = 1, j = 3, k = 2; 1/(1 + i + j) + (i + j + k++) / (k + 3 * i + 2 * j++)	0
int x = 1, y = 5, z = -2; -x + (z + y * (z + x) / z - y / x) + x - y / 2	-7
int i = 1, k = 2; 3 * i - 4/k < 2	1
int k = 2; 'x' + k++ == 'x' + k + 1	0
int i = 1, x = 10, y = 10, z = -10; i + 5 x < !i && z	1

Note: The parts of these question are shuffled two sets. So the order of the answers will be different in the other set.

2. Study the following program which counts the number of characters, the number of white spaces and the number of digits in a text given as input, where the end of the input is signalled by inserting a CTRL-D.

```
#include <stdio.h>
int main() {
    char c;
    int nspace = 0, ndigit = 0, nother = 0;
    while ((c = getchar()) != EOF) {
        switch (c) {
            case '0' :
            case '1' :
            case '2' :
            case '3' :
            case '4' :
            case '5' :
            case '6' :
            case '7' :
            case '8' :
            case '9' : ndigit++;
                       break;
            case ' ' :

```

```

        case '\n':
        case '\t': nspace++;
                break;
        default  : nother++;
                break;
    }
}
printf("Number of digits = %d\n", ndigit);
printf("Number of white spaces = %d\n", nspace);
printf("Number of other characters = %d\n", nother);
}

```

The program is rewritten using if-else as follows.

```

#include <stdio.h>
int main() {
    char c;
    int nspace = 0, ndigit = 0, nother = 0;
    while ((c = getchar()) != EOF) {
        if (?1?)
            ndigit++;
        else if (?2?)
            nspace++;
        else
            ?3?;
    }
    printf("Number of digits = %d\n", ndigit);
    printf("Number of white spaces = %d\n", nspace);
    printf("Number of other characters = %d\n", nother);
}

```

But the segments marked ?1? through ?3? are not clear. Provide the correct substitutes (use of comma operator is not permitted) for those segments so that the program is equivalent to the one given above. [4+4+2]

Solution:

```

?1?:  c >= '0' && c <= '9'
?2?:  c == ' ' || c == '\n' || c == '\t'
?3?:  nother++

```

3. Carefully understand the following C program.

```

#include <stdio.h>

```

```
int main() {
    int a, b, d, n = 0;

    printf("Enter a 1 digit number: ")
    scanf("%1d", &a);

    if (a < 2)
        return 0;

    printf("Enter another number (2 or more digits): ")
    scanf("%d", &b);

    while (b > 0) {
        d = b % 10;
        if (d <= a-1) {
            n = n * 10 + d;
            b /= 10;
            continue;
        }
        printf("Incorrect number for the base %d\n", a);
        return 0;
    }

    b = n;
    n = 0;

    while (b > 0) {
        d = b % 10;
        n = n * a + d;
        b /= 10;
    }

    printf("output: %d\n", n);
}
```

What is the output if inputs are: $a = 4$, $b = 1221$.

[10]

Solution: The program takes a as the base and b as the number in base a , then it computes decimal equivalent of b .
output: 105 (1221 in base 4 in this set)
output: 186 (1221 in base 5 for the other set)

4. Fibonacci sequence is defined by following rules:

$$f_n = \begin{cases} 1, & n = 0 \\ 1, & n = 1 \\ f_{n-1} + f_{n-2}, & n \geq 2 \end{cases}$$

A C program for computing Fibonacci numbers is given below with some segment of code missing. The missing segments are marked as ?1? through ?5?. Find the correct substitute for each of these segments by a simple statement or an expression (use of comma operator is not permitted) so that the program works as desired. [5×2]

```
#include <stdio.h>
int main() {
    int f_new, f_cur, f_old, i;
    int n;

    printf("Enter a number between 1 to 15: ");
    scanf("%d", &n);

    ?1?;
    ?2?;
    f_old = 1;

    while (i < n-1) {
        f_new = ?3?;
        f_old = f_cur;
        f_cur = ?4?;
        ?5?;
    }
    printf("%6d\n", f_cur);
}
```

Solution:

?1?: i = 0

?2?: f_cur = 1

?3?: f_cur + f_old

?4?: f_new

?5?: i++

5. Suppose we are interested for a C program that should take three integers as input and print them in ascending order. Some one provided the following program to us:

```
L1 :   #include <stdio.h>
L2 :   int main() {
L3 :       int x1, x2, x3;

L4 :       printf("Enter three numbers: ");
L5 :       scanf("%d %d %f", &x1, &x2, &x3);

L6 :       if (x1 > x2) {
L7 :           x1 = x2 - x1;
L8 :           x2 = x2 - x1;
L9 :           x1 = x1 + x2;
           }

L10:      if (x2 > x3) {
L11:          x3 = x3 + x2;
L12:          x2 = x3 - x2;
L13:          x3 = x3 - x2;
           }

L14:      if (x1 > x3) {
L15:          x1 = x3 - x1;
L16:          x2 = x3 - x1;
L17:          x3 = x1 + x2;
           }

L18:      printf("%5d%5d%5d\n", x1, x2, x3);
           }
```

But the program, when executed, did not produce the correct result. On debugging five errors were found and fixed. After that the program executed correctly. Identify and correct those errors so that the program excutes as desired. You should mention the line number of the code where an error is found, then give the correct substitute for that line. [5×2]

Roll:

Name:

Section:

Solution:

Error	Line No	Correction
1	L5	substitute "%d" for "%f"
2	L14	substitute "x2" in place of "x3"
3	L15	substitute "x2" in place of "x3"
4	L16	substitute "x2" in place of "x3"
5	L17	substitute "x1" in place of "x3"

Note: The program does not have any syntactical errors. All syntax errors are caught at compile time. Logical errors are caught during debugging.