

(Answer All the Questions)

Group A

1. Answer the following Questions in short (Any **Fifteen**). 15 × 1 = 15

- (a) Let `int a=545, b = 257`; Use only **Bitwise Assignment Operators** to swap the values of **a** and **b**.
- (b) Rewrite the **binary** number 1001111111111100000000001101010010101010101 as a **hexadecimal** number.
- (c) What is the difference between `.` and `->` in case of accessing the elements of a structure?
- (d) What is the difference between `#include <wow.h>` and `#include "wow.h"`?
- (e) An array is declared by `double x[10][34][21]` where `&x[0][0][1] = EF8`. What would be the address of `x[7][5][11]`?
- (f) What are the three rules to make an **Identifier**?
- (g) What is the difference between **201751** and **201751UL**?
- (h) What is **null statement**?
- (i) What does the **circumflex** used for?
- (j) `int a, b, c = 'k'; scanf("%c%c%c",&a,&b,&c);printf("%c%c%c",a,b,c);` For an input `a b`, what would be printed? [For more specification, Input is 'a',space,'b']
- (k) What is the difference between `%o` and `%#o`?
- (l) `char a[]="bc", b[]="abc"; printf("%d",strcmp(a,b));`—What would be printed?
- (m) Which of the followings are valid identifiers? In case of invalid identifier, state the reason.

i) short *ii)* _UndErScore *iii)* cSe'17 *iv)* 9ty

- (n) Fill in the gap with one word so that the code executes without any error.

```
#include <stdio.h>                                int main()
----- struct Books {                            {
    char title[50];                                Book book;
    int book_id;                                   return 0;
} Book;                                           }
```

- (o) Give an example of **bitfield**.
- (p) For input **12324322342** what would be the value of `a,b,c,d` after executing the code—`int a=1,b,c,d=9;scanf("%2d%3d%1d%4d",&a,&b,&c,&d);?`
- (q) Rewrite the following code with conditional operator—`if(x&1)x^=((1<<y)-1); else x&=((y>>2)-1);`
- (r) What would be the output of the following piece of code?

```
int x = 8, k = 16;
if(x++==(k>>1) || k-->9) x--;
printf("%2X%2o\n",x,k);
```
- (s) Write the syntax of **fgets** to take a line from standard input and store in a character array named **ins**.
- (t) `char s[]="cse17 batch is amazing!";printf("%-7.5s:\n",s);`—What would be printed? Represent spaces by box (□);

2. Answer the following Questions (Any **Three**). 3 × 5 = 15

- (a) Observe the following two structure declarations for a **Singly Linked List**. Are they syntactically correct? Compare them visualizing their memory representations.

```
i) struct node{
    int x;
    struct node n;
};
```

```
ii) struct node{
    int x;
    struct node *n;
};
```

- (b) Write a program to find the Abundant numbers (integers) between N and M [$1 \leq N, M \leq 1000$].

Note: An **Abundant Number** or **Excessive Number** is a number for which the sum of its proper divisors is greater than the number itself.

- (c) Write a program to print the following sequence (where every ... means rest of the similar strings are skipped)

```
ABC ABD ... ABZ ACD ACE ... ACZ ADE ADF ... ... AWX AWY AWZ AXY AXZ AYZ
BCD BCE ... BCZ BDE BDF ... BDZ BEF BEG ... ... BWX BWY BWZ BXY BXZ BYZ
CDE CDF ... CDZ CEF CEG ... CEZ CFG CFH ... ... CWX CWY CWZ CXY CXZ CYZ
...
WXY WXZ WYZ
XYZ
```

- (d) Write the output of the following program:

```
#include <stdio.h>
int main()
{
    int x, y, z=3;
    for (x=26,y=z; ((z%40)%21)!=4; z%=7)
    {
        y = ((x++ + ++z) * 210 - y) % 49;
        printf("(%%d= %d) x=%d,y=%d,z=%d\n",z,x,y,z,x,y,z);

        y = ((--x + z++) * 210 - y) % 49;
        printf("(%%d= %d) x=%d,y=%d,z=%d\n",z,x,y,z,x,y,z);

        z = ((++x+y+z)%(x*y))%60;
        printf("(%%d= %d) x=%d,y=%d,z=%d\n",z,x,y,z,x,y,z);
    }
    return 0;
}
```

- (e) Write a program that generates the **cumulative exclusive or** of the first **n** terms of the **Lucas Series** where $L_0 = 2, L_1 = 1$. **n** (< 20) would be given through the standard input. Show all cumulative exclusive or values through the standard output.

3. Answer the following Questions (Any **Two**).

$2 \times 10 = 20$

- (a) Write the code to generate all primes within 10000 using **sieve of eranthoses**. After running your code, all generated primes should be in an integer array named **primes**.
- (b) Write the output of the following program:

```
#include <stdio.h>

int bar(int x)
{
    if(x&2&& x>0)printf("%05d ", x);
    if(x<0||x>10) return -x;
    int ret = 0;
    switch(x)
    {
        default:
            ret+=bar((x-->4)-3)+bar(x);
        case 4:
            ret-=bar(x+13);
            break;
        case 0:
            ret%=bar(x-2);
    }
    return ret;
}

int main()
{
    int y = 0,n=8;
    puts("What the heck is this puts!");
    printf("Recursion means %#4o!\n",n);
    if( ((1<<4)&n) == 1 ) puts("No!");
    for (n=~n;;y++)
    {
        puts("Should I call?");
        if(y%3) continue;
    }
}
```

```

    printf("bar(%d)=%d\n",y,n=bar(y));  printf("Well done Mr Recursion!");
    if(y>n) break;                      return 0;
    printf("Milestone#%02d\n",y);      }
}

```

- (c) Write a function **do_work** which takes a function as parameter. The parameter functions' name should be either **multi** or **politi**. The function **multi** takes three integers as parameter and returns the summation of them. The function **politi** also takes three integers as parameter and returns subtracting the third argument from the multiplication of the first two arguments.
- (d) Given an adjacency matrix named `int adj[100][100]` (declared globally) of a graph with `n` nodes. Write a function `int findCycle(int n)` which returns 1 if there exists a cycle in the graph, or returns 0 otherwise.

Group B

1. Answer the following Questions in short (Any **Fifteen**). 15 × 1 = 15

- (a) `int x = sqrt(24.5);printf("%.2f\n",(float)x);`—What would be printed by this piece of code?
- (b) `double k = 28.6;int p = k; x = (p>>2); printf("%d\n",x);`—What would be the output?
- (c) What is the difference between `%x` and `%X` while printing a hexadecimal number?
- (d) “I am in comment”—Write this line without quote in a multi-line comment.
- (e) `enum wow{a,b=-2,c,d=1,e};printf("%02d%03d%04d",e, c, a);`—What would be the output?
- (f) Is there any difference between the declarations `char cse[]={'1','7'};` and `char cse[]="17";`? Show the reason behind your answer.
- (g) What is **Modularity**?
- (h) `int x[10]={4,5}, *p=x;*x++;printf("%d\n", *x++);`—What would be printed?
- (i) Give an example of **assignment suppression**.
- (j) What is the difference between `%e` and `%f`?
- (k) `int i=51;printf("%-6d:%+6d\n",i,i);`—What would be printed?
- (l) Write an expression to generate the mask **00000001111111111111000**.
- (m) What is the difference between declaring a variable like `int a;` and `register int a;`?
- (n) Let (U)`const static v=9;` and (V)`static const int v=9;`—Which one is syntactically correct?

i) U ii) V iii) both of them iv) none of them

- (o) What is **Garbage Value**? Give an example.
- (p) `int p=16,q=14;p^=q^=p^=q;printf("%X,%X",p,q);`— What would be the output?
- (q) What is the difference between `%f` and `%.#f`?
- (r) Draw a table to represent the 2D array `x` which is initialized by `int x[4][3]={3,5,2,1,9,-4}`.
- (s) Can you make the following code compilable by adding only one line? Remember, you could not remove or change anything else.

```

#include <stdio.h>
int main()
{
    int x=5, y yy;
    x = x y x;
}

```

- (t) What is the difference between declaring `k` as `union {int a; double b;}k;` and `struct {int a; double b;}k;`?

2. Answer the following Questions (Any **Three**).

$3 \times 5 = 15$

- (a) Write the output of the following program showing the calculations.

```
#include <stdio.h>
#define SZ 5+6
#define SZ2 9-SZ+SZ
int main()
{
    int N = (~( ~( (12*SZ2*2)<<2)-1))%366;
    printf("%010o\n",++N+6);
    return 0;
}
```

- (b) Sort the following numbers using **Counting Sort**: 193, 213, 432, 198, 587, 478, 897, 911, 199, 121. Show all the steps.

- (c) Write a program with the following specification and determine it's complexity.

Input: A number N , then N floating-point numbers.

Output: A permutation of the given N floating-point numbers where i would occur before j if $i \leq j$.

- (d) Write a program with the following specification. The program must have logarithmic complexity.

Input: A number N , then N floating-point numbers sorted in ascending order. Then a floating-point number K .

Output: Print **YES** if K occurs in the sorted N floating-point numbers, otherwise print **NO**.

- (e) Write a C function named **average** which takes variable number of integer arguments and returns the average of those arguments as a floating point value. Example: Calling **average(5,3,4,5,6,7)** returns **5**. and Calling **average(4,100,3,10,1)** returns **28.5**.

3. Answer the following Questions (Any **Two**).

$2 \times 10 = 20$

- (a) i. Write a program to sort the array initialized by

```
int a[10]={76,3,443,323,232,988,1,434,-98};.
```

5

- ii. Write a program to convert a **binary number** to a **decimal number**.

5

- (b) Write the output of the following program:

```
#include<stdio.h>
int foo(int *x,int z[])
{
    static int y = 90;
    x=&y;z[1]++;++y;
    (*x)++;z[5]=11;
    return *x;
}

int main()
{
    int x = 8,i,y[10]={3,9},z;
    z=foo(&x,y)+foo(&x,y);
    printf("%d %d\n",x,z);
    for (i=0; i<y[1]; i++)
        printf("y[%d]=%d\n",i,y[i]);
    return 0;
}
```

- (c) There are 3 files named **Name.in**, **Age.txt** and **CGPA.txt**. In each file, there are 10 lines. Each line in Name.in, Age.txt and CGPA.txt contains name (sequence of characters), age (integer) and cgpa (floating point number) of a student respectively. Create another file named **Data.out** and write all these information of a student in a line separated by comma. Example: A student named "Saban Ali", aged 19 years old having CGPA 4.00 should be stored in a line like: **Saban Ali, 19, 4.00** Remember, you must not take any two students data in memory simultaneously.

- (d) Multiply two matrices kept in two 2D array declared by `int a[10][10]`, `b[10][10]`; and keep the result in another matrix declared by `int c[10][10]`. The dimension of a and b matrices are $n \times m$ and $p \times q$ respectively. Access all the values in these arrays using pointer notation.