United International **University (UIU)**

Dept. of Computer Science & Engineering (CSE)

Mid-term Exam: : Trimester: Fall 2023

Course Code: CSE 1111, Course Title: STRUCTURED PROGRAMMING LANGUAGE

Time: 1 hour 45 min   Total Marks: 30

Answer all the questions.

"Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules."

1. (a) Which of the following variable names are **invalid** and **why**? [1]

   (i) int_a (ii) _num (iii) 99p (iv) "my_val"

   (b) **Compute** the values of the variables **a, b, c,** and **d**. ASCII codes: A-65, a-97, 0-48. [2]

   ```
   (i) float a = 101/7;
   (ii) float b = (float)(3%5);
   (iii) float c = 21>43 || 6!=6;
   (iv) double result = 12 + (1 * '3');
   ```

   (c) **Find outputs** of the following code segment for (i) **num = 2.3,** and (ii) **num =127.** [3]

   ```
   int num;
   scanf("%d", &num);
   if (num % 2 != 0)  {
     printf("Mashrafe\n");
   }
   if (num < 100)  {
     printf("Shakib\n");
   }else if (num >= 100){
     printf("Mahmudullah\n");
   }
   if (num >= 0 && num < 5){
     printf("Imrul\n");
   }else if (num >= 0 && num
   <= 49){
     printf("Tamim\n");
   }else{
     printf("Rubel");
   }
   ```
   C Code for **1(c)**

   ```
   int a,b,c;
   scanf("%d%d%d",&a,&b,&c);
   int result=a--/b++;
   switch(a+b){
       case 1:
           result+=a/c*2;
           b++;
       case 2:
       case 3:
           result=a*c/b;
           a++;
       case 4: break;
               a=2;
       default: result=5;
   }
   printf("%d %d %d %d",
               a,b,c,result);
   ```
   C Code for **2(a)**

2. (a) **Rewrite** the code segment (see above right) using "**if … else**" without changing the logical meaning. [3]

   (b) **Manually trace** the following code segment and show **all** the change of values of the variables *start*, *end*, *i*, *count* in each step. [3]

   ```
   int start=105,end=112,count=0;
   for(int i=end;  i>=start;  i--){
       if(end%2 != 0){
           count++;
           start++; end+2;
       }else{
           end--; start+1;
       }
   }
   ```

3. (a) Write a C program to display the following 'M' pattern for n. For example, for n = 3, and n=5 [3]
the output pattern will be as follows. You must program for n, not for 3 or 5.

Sample input	n=3			n=5				
Sample output	*		*	*				*
	*	*	*	*	*		*	*
	*		*	*		*		*
				*				*
				*				*

(b) Replace the "**outer**" **for** loop using "**while**" loop and the "**inner**" **for** loop using "**do while**" [3]
loop in the following code without changing the logical meaning of the program.

```
int a=10, b=20, count=0;
for(int i=b;i>=a;i--){
    for(int j=a;j<=b;j++){
        printf("%d ",j);
    }
    if(b%2!=0){
        printf("%d \n",a);
    }else{
        printf("%d \n",b);
    }
}
```
C Code for **3(b)**

```
int n;
printf("Enter a +ve integer:");
scanf("%d", &n);
if (n <= 0) {
    printf("Enter a +ve integer.");
}else {
    int fact = 1, i = 1;
    do {
        fact *= i;
        i++;
    } while (i <= n);
    printf("Factorial of %d is %d",
                          n, fact);
}
```
C Code for **4(a)**

4. (a) Draw a flow chart of the given code segment (see above right). [3]
   (b) Write a C program to **take** input of all the bank account balance of *n* clients of a bank.
       Remove any balance less than 500.00 taka. **Now, display** all the balances. [3]

5. (a) **Manually trace** the given code segment (see below left). Show the changes of all the [3]
       variables *i*, and array *ara* elements in each step.

```
int ara[5] = { 8,6,2,4,7 };
for(int i = 1; i < 5; i += 2){
    ara[i] =  3 * ara[i - 1];
}
for(int i = 1; i < 5; i++){
    if(i % 2 == 0){
        ara[i] = i * 4 + ara[i-1];
    }
}
```
C Code for **5(a)**

```
int row, col, sum = 0;
int A[][3]={{1,2,3}, {11,5,6},
            {12,7,9},{8,13,4}};
for(row=0; row<4; row++){
    for(col=0; col<3; col++){
        if(col>row)  {
            sum += A[row][col];
        }
    }
}
```
C Code for **5(b)**

(b) **Manually trace** the given code segment (see above right) and show the changes of all the [3]
variables *row*, *col*, and *sum* in each step.