ourdiu.com | Shared by S. M. Jarif Haider Amio

Daffodil International University

Faculty of Science & Information Technology

Department of Software Engineering

Final Examination, Summer 2026

Course Code: SE 121; Course Title: Structured Programming

Sections & Teachers: A, B, C (MIR), D, E (MR), F, G, H (MNA), I (KRY)

Time: 2:00 Hrs **Marks: 40**

Answer ALL Questions

[The figures in the right margin indicate the full marks and corresponding course outcomes. All portions of each question must be answered sequentially.]

1. a) **Examine** the following programs and **modify** them to correct the errors. **[Marks-8]** **CLO-2** *Level-3*

(i)
```
#include <stdio.h>
int main(){
char name = "Alice";
printf("Enter your name: ");
scanf("%s", &name);
printf("The first letter of your
name is: %s\n", name[1]);
return 0; }
```

(ii)
```
#include <stdio.h>
#include <string.h>
int main(){
char str1[20], str2[20];
strcpy(str2);
for(int i = 0; str1[i] = '\0'; i++)
printf("%s", str1[i]);
printf("\nLength = %d",
strlen(str2, str1));
return 0; }
```

b) Daffodil International University is developing a new Student Registration Portal to simplify the course registration process. To ensure that only registered DIU students can create an account, the portal accepts registrations only from official university email addresses. During registration, the system verifies whether the entered email address ends with @diu.edu.bd. If it does, the registration proceeds; otherwise, the email is rejected. **[Marks-7]**

Build a C program based on the scenario above that takes an email address as input and checks whether it is a valid DIU email address.

Sample Input	**Sample Output**
Enter your DIU email: mr@diu.edu.bd	Valid DIU Email

2. a) **Determine** the output of the following C program and **trace** your answer. **[Marks-8]** **CLO-3** *Level-3*

ourdiu.com | Shared by S. M. Jarif Haider Amio

(i)
```
#include<stdio.h>
int calculate(int arr[], int i){
    if(arr[i] % 2 == 0 && arr[i] >5)
        return arr[i] * 2;
    else return arr[i] - 1; }
int main() {
    int data[] = {4, 8, 5, 10};
    int output= 0;
    for(int i = 0; i <= 4; i++) {
        output+= calculate(data, i); }
    printf("Result: %d", output);
    return 0; }
```

(ii)
```
#include <stdio.h>
int count(char str[]){
    int c = 0;
    for(int i = 0; str[i] != '\0'; i++){
        if(str[i] == 'a' || str[i] == 'c'|| str[i]
== 'i'){
            c += 2; }
        else c++;
    } return c; }
int main(){
    char word[] = "hacked";
    printf("Output: %d", count(word));
    return 0; }
```

b) A friendly rivalry breaks out between CR7 and LM10. To settle it, they decide on a challenge. First, the user enters the size of an array, followed by the array elements. CR7 earns a point for every prime number in the array, while LM10 earns a point for every non-prime number. At the end of the challenge, the winner is determined as follows: if the number of prime elements is greater than the number of non-prime elements, print "CR7 Wins"; if the number of non-prime elements is greater, print "LM10 Wins"; otherwise, print "It's a Draw". [Marks-7]

Construct a C program for the given scenario.

Sample Input	**Sample Output**
Enter the size: 6 Numbers: 2 4 7 9 10 11	Prime count: 3 Non-Prime count: 3 It's a Draw

3. a) A group of volunteers has only one power bank with a limited battery capacity. The organizer wants to know how many mobile phones can be fully charged before the power bank runs out. Each phone requires the same amount of charge. [Marks-4] CLO-5 *Level-4*

Identify the maximum number of mobile phones that can be fully charged before the power bank runs out by writing a C program that uses a user-defined function.

Sample Input	**Sample Output**
Enter capacity and charge: 25000 4200	Phones Fully Charged = 5

b) An e-commerce platform maintains information about its products using a database. Since the number of products available may vary over time, the system first asks the administrator to enter the number of products to be stored. For each product, the system records the Product ID, Product Name, and Price. To help customers find products within their budget, the platform allows users to specify a minimum and maximum price. The system then displays the Product Name and Price of all products whose prices fall within the given price range (inclusive). [Marks-6]

Classify the products based on the specified price range by writing a C program using a structure.