

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT) 6

ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

SEMESTER FINAL EXAMINATION

WINTER SEMESTER, 2022-2023

DURATION: 3 HOURS

FULL MARKS: 150

CSE 4175: Computer Programming**Programmable calculators are not allowed. Do not write anything on the question paper.**Answer all **6 (six)** questions. Figures in the right margin indicate full marks of questions whereas corresponding CO and PO are written within parentheses.

1. a) How would you differentiate between implicit and explicit type conversion? - Explain with proper examples. 7
(CO1)
(PO1)

- b) Consider the program of Code Snippet 1. 8
(CO2)
(PO1)

```

1 int x(int n) {
2     static int i = 1;
3     if (n%8==0) return n;
4     n = n+i;
5     i++;
6     printf("%d\n",n);
7     return x(n);
8 }

```

Code Snippet 1: C program containing a function for Question 1.b)State the values returned from $x(1)$ and explain the output of each recursion call.

- c) Write a program that accepts an integer as input and outputs each digit in written English. Refer to Table 1 for a sample input and output. Print the output using a user-defined function. 10
(CO3)
(PO1)

Table 1: Sample input and output for Question 1.c)

Sample Input	Sample Output
932	nine three two

2. a) Write a C program that takes an integer n , the size of a pattern as input and prints the pattern as illustrated in Table 2. 7
(CO3)
(PO2)

Table 2: Sample input and output for Question 2.a)

Sample Input	Sample Output
	# #
4	# # #
	# # # #
	# # # # #

b) Code Snippet 2 represents function definitions and variable declarations of a C program.

4 × 2
(CO1)
(PO1)

```
1 int p, q;  
2 void f(void) {int b,d;}  
3 void g(int a) {  
4     int c;  
5     {int x,y;}  
6 }  
7 int main(void) {  
8 int u,v;  
9 }
```

Code Snippet 2: Function definitions and variable declarations for Question 2.b)

For each of the following scopes, list all variable and parameter names visible in that scope.

- The `f` function.
 - The `g` function.
 - The block in which `x` and `y` are declared.
 - The `main` function.
- c) Write a function `find_substring(char *str1, char *str2)` that takes two character pointers pointing to strings as arguments. It returns 1 if `str2` is found as a substring anywhere in `str1`, otherwise, it returns -1. It is assumed that `str2` will always be smaller in size than `str1`. Refer to Table 3 for a sample input and output.

10
(CO3)
(PO2)

Table 3: Sample input and output for Question 2.c)

Sample Input	Sample Output
Madam adam	1

3. a) Explain how a text file differs from a binary file with proper examples.

7
(CO1)
(PO1)

- b) Write the output of Code Snippet 3:

```
1 #include <stdio.h>  
2 int main() {  
3 int a[6]={25,24,23,22,21,20};  
4 int *pl, t, u, v,w,x,y,z;  
5 pl=a;  
6 t=(*pl++)++;  
7 u=++(*pl);  
8 v=*pl;  
9 w=+++(++pl);  
10 x=*pl++;  
11 printf("%d %d %d %d %d ", t, u, v, w,x);  
12 return 0;  
13 }
```

10
(CO1)
(PO1)

Code Snippet 3: C code segment for Question 3.b)

- c) You were dreaming of a journey to the planet "RetroP" on a spaceship in space represented as a plane. You are going to start your journey at point (x, y) and the planet is located at (p_x, p_y) point. The navigation system of your spaceship follows a list of orders which can be represented as a string s . The system reads s from left to right where each letter represents a direction towards planet "RetroP". For example, when you are at point (x, y) and if:

8
(CO3)
(PO2)

- $s_i = U$, you move to $(x, y + 1)$;
- $s_i = D$, you move to $(x, y - 1)$;
- $s_i = R$, you move to $(x + 1, y)$;
- $s_i = L$, you move to $(x - 1, y)$.

Now, write a program that, given a starting point and an order string as input, prints the location of the planet "RetroP". See Table 4 for a sample input and output.

Table 4: Sample input and output for Question 3.c)

Sample Input	Sample Output
10 5 RRURLLLLUULUUU	9 11

4. a) Analyze Code Snippet 4 and trace the values of `text1` and `text2` at each step, assuming input from the keyboard as "Hello World" for `text1` and "Programming is fun" for `text2`. State the transformation of `text1` and `text2` from lines 7 to 14.

7
(CO2)
(PO1)

```
1 #include<stdio.h>
2 #include<string.h>
3 int main() {
4     char text1[100],text2[100], temp[100];
5     gets(text1);
6     scanf("%s",text2);
7     strncpy(temp,text1,7);
8     strcat(text2,temp,4);
9     strcpy(temp,text2);
10    strcat(temp,text1,4);
11    if (strcmp(text2,temp)>0) strncpy(text1,temp,2);
12    else strncpy(text2,temp,2);
13 }
```

Code Snippet 4: C code segment for Question 4.a)

- b) Write down the output of Code Snippet 5:

8
(CO2)
(PO1)

```
1 #include <stdio.h>
2 int main (void){
3     char s[] = "Hsjodi";
4     char *p;
5     for (p = s; *p; p++)
6         --* p;
7     puts(s);
8     return 0;
9 }
```

Code Snippet 5: C code segment for Question 4.b)

- c) Write a user-defined function to check the strength of a password for a registration system. The function returns 1 if the password is strong; otherwise, it returns -1, adhering to specific criteria for a legal password as follows: 10
(CO3)
(PO1)
- The password needs to have at least 12 characters;
 - The password has to be a mix of lowercase and uppercase letters;
 - The password must contain at least one number;
 - The password must contain at least any one of these special characters) , . ; : < > (
5. a) In the game "Jungle Run," two players can engage in a match, each having an individual run and earning points. The points for each match are then stored in a text file. A match can have a winner or end in a tie. 3 +
3 +
3 +
5 + 5
(CO3)
(PO2)
- i. Create an enumeration for a match result, which will have player1 set to 1, player2 set to 2, and tie set to 0.
 - ii. Create a structure type named match. Each match will store two integer scores for both participants and an integer for the result.
 - iii. Write a user-defined function that will take input of two points for a match from the user and save it in the structure match variable. The function will return the address of a match structure object.
 - iv. Write a user-defined function that takes the address of a match structure object for calculating the winner and saves it in the result field of that object.
 - v. Write a program that will store the updated variable of match structure type in a file named "Jungle Run Score.txt".
- b) Explain the advantages and disadvantages of using external variables in a C program. 6
(CO1)
(PO1)
6. a) Imagine developing an inventory system for a gift shop named "Wishee" using a C program. Create efficient structures for the system, where each item encompasses the following information: 10
(CO3)
(PO2)
- stock ID (integer)
 - price (float)
 - item type (integer)
 - product information depending on the item type,
 - t-shirt: brand (string), size (number), design (string)
 - toy: brand (string), category (string), age limit (number)
 - card: color (string), occasion (string), design (string)
- b) With the upcoming New Year sales at "Wishee," special discounts will be available. Various offerings will apply to a customer's total purchase, including: 8
(CO3)
(PO1)
- If a user purchases any item that costs between \$101 and \$200, the discount will be 5%.
 - If a user purchases any item that costs between \$201 and \$400, the discount will be 10%.
 - If a user purchases any item that costs between \$401 and \$800, the discount will be 20%.
 - If a user purchases any item that costs more than \$800, the discount will be 25% up to \$350.
- Now write a program that will take an array of structure items (from Question 6.a)) and show the total price and the price after the discount.
- c) Explain the difference between structure and union with the necessary diagrams. 7
(CO1)
(PO1)