



**RAJARATA UNIVERSITY OF SRI LANKA
FACULTY OF APPLIED SCIENCES**

**B.Sc. (General) Degree in Applied Sciences
First Year - Semester II Examination – October/November 2017**

COM 1407 – COMPUTER PROGRAMMING (THEORY)

Time: Three (3) hours

Examination Index No: _____

Important Instructions:

- This paper has 4 questions in 16 pages.
- Answer all questions (25 marks each).
- Write your answers in English using the space provided in this question paper.
- Do not tear off any part of this question paper.
- Note that questions appear on both sides of the paper.
- If a page is not printed, please inform the supervisor immediately.

To be completed by the examiners:

Questions	Question numbers				Total Marks
	1	2	3	4	
Marks					

Question 01

- i. How does the Selection control structure function in the structured programming paradigm? (2 Marks)

- ii. What is the meaning of following statement with respect to the programming languages?
“The programs written in Low Level Languages are not portable.” (2 Marks)

- iii. What is the main task of the Compiler and the Assembler in the C program execution environment? (4 Marks)

- iv. Write a C program to display the following pattern using two `for` loops. (5 Marks)

```
* * * * *
* * * *
* * *
* *
* 
```

v. What will be the output of following C code snippets?

(6 Marks)

```
#include <stdio.h>
int main ()
{
    char c = 125;
    c=c+10;
    printf("%d",c);
    return 0;
}
```

```
#include<stdio.h>
int main()
{
    int a=2;
    if(a==2)
    {
        printf("%d\n",~a);
        printf("%d\n",~a+2);
        a=~a+2<<1;
        printf("%d",a);
    }
    return 0;
}
```

```
#include <stdio.h>
int main (void)
{
    int u=2, v=10, temp;
    while ( v != 0 )
    {
        temp = u % v;
        u = v;
        v = temp;
    }
    printf ("%d\n", u);
    return 0;
}
```

```
#include <stdio.h>
int main ()
{
    int x, y, z=0 ;
    for (x = 1, y = 50; x<=5; x++, y=y-10 )
        z = x* y;
    printf ("%d ",z);
    return 0;
}
```

- vi. Following recursive function finds the factorial of a given number n . Explain how it generates 120 as the output when $n = 5$. Show all steps (6 Marks)

```
int factorial(int n)
{
    if (n >= 1)
        return n*factorial(n-1);
    else
        return 1;
}
```

Question 02

- i. State the importance of adding comments into the source codes in programming. (2 Marks)

- ii. What is a function prototype and why is it essential in C programming? (3 Marks)

- iii. Mark TRUE or FALSE for the statements given below. (5 Marks)

(a) The C language enables to define a set of ordered data items in same type into a structure called array.	TRUE	FALSE
(b) C arrays do not store data in contiguous memory locations.	TRUE	FALSE
(c) Following C statement will take the 4 th element from the array and assign the value to salary variable. <code>double salary = balance[3];</code>	TRUE	FALSE
(d) Following statement follows correct C syntax. <code>double balance[3] = {1000.0, 2.0, 3.4};</code>	TRUE	FALSE
(e) C does not do array bound checking.	TRUE	FALSE

Following Table 1 lists the number of passenger transits recorded in month of September and the figures are organized based on the three different train types which were operated on three main transit railway stations of the country.

Station with Station Identification (SIC)	Station Code	Long distance (Train Type No = 1)	Express (Train Type No = 2)	Commuter (Train Type No = 3)
Fort (SIC = F)		7000	5800	70000
Galle (SIC = G)		8000	3400	20000
Kandy (SIC = K)		10000	4000	30000

Table 1 – Railway passenger transit of month 09

Write the answers for the following Question 2 part (iv), (v) and (vi) based on the information given in Table 1.

- iv. Declare and initialize a two dimensional array called `passenger_info` to keep the facts listed in Table 1. (3 Marks)

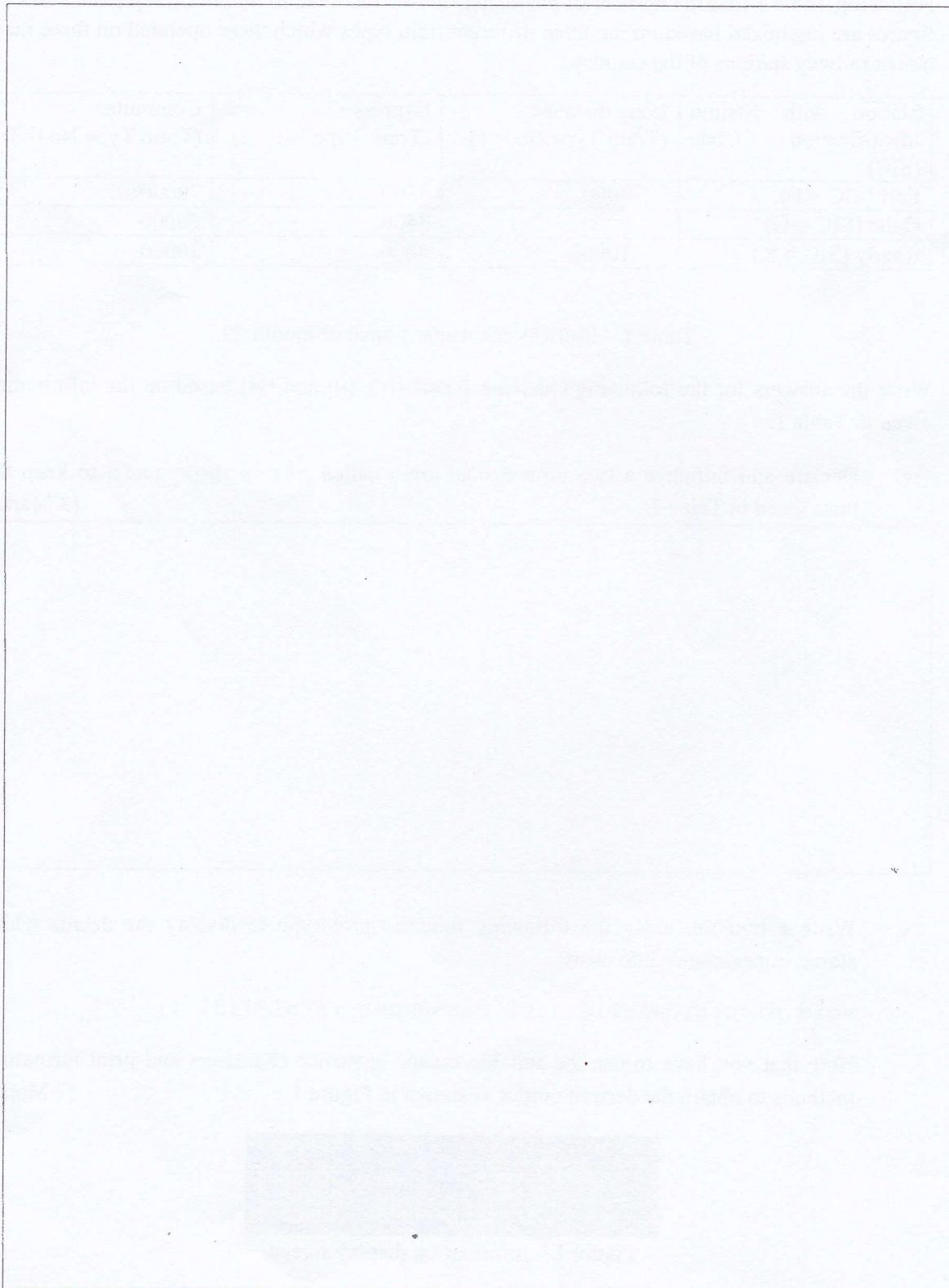
- v. Write a function using the following function prototype to display the details which stored in `passenger_info` array.

```
void displayDetails (int passenger_info[3][3] );
```

Note that you have to use the suitable escape sequence characters and print formatting methods to obtain the desired output as shown in Figure 1. (7 Marks)

SIC	Type1	Type2	Type3
F	7000	5800	70000
G	8000	3400	20000
K	10000	4000	30000

Figure 1 – Information display screen



- vi. Write a function using the following function prototype to return the maximum passenger transits handled according to the input train type number.

```
int findMaximumTransitsHandler (  
    int passenger_info[3][3], int trainTypeNumber);
```

Note that the array index for train type number is 1 less than the user entered train type number. e.g. array index is 0 for the user entered train type number 1. (5 Marks)

Question 03

- i. Comment on following C statement. (2 Marks)

```
int ptr = (float*) calloc(25, sizeof(float));
```

- ii. What is the main difference between C structure and union? (2 Marks)

- iii. Mark TRUE or FALSE for the statements given below. (5 Marks)

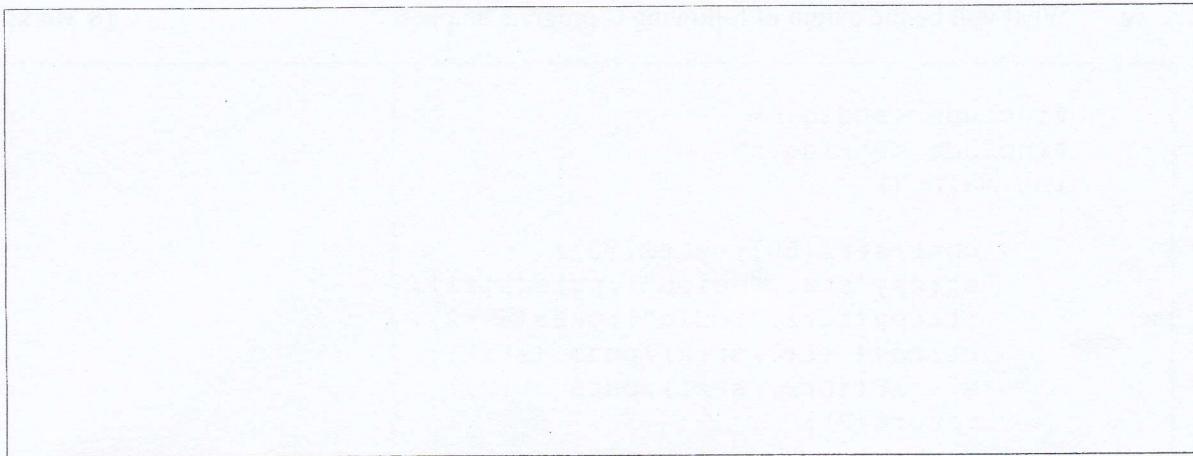
(a) Following union Data type will allocate 20 bytes of memory space. <pre>union Data { int i; float f; char str[20]; }data;</pre>	TRUE	FALSE
(b) Dynamically allocated memory created with either <code>calloc()</code> or <code>malloc()</code> does not get freed on its own.	TRUE	FALSE
(c) C structures cannot keep an array as one of its' elements.	TRUE	FALSE
(d) Following C statement initializes both Player1 and Player2 instances with the values given in the pair of braces. <pre>struct Participant Player1, Player2 = {"Korea", "Kenya", 4.6, 19};</pre>	TRUE	FALSE
(e) Following C statement declares an array called Player[] to store two values of Payer type structures. <pre>struct Participant Player[2];</pre>	TRUE	FALSE

iv. What will be the output of following C program snippets?

(8 Marks)

```
#include <stdio.h>
#include <string.h>
int main ()
{
    char str1[50], str2[50];
    strcpy(str1,"aeiou");puts(str1);
    strcpy(str2,"bcdgf");puts(str2);
    strcpy(str1,str2);puts(str2);
    strcat(str2,str1);puts(str2);
    return(0);
}
```

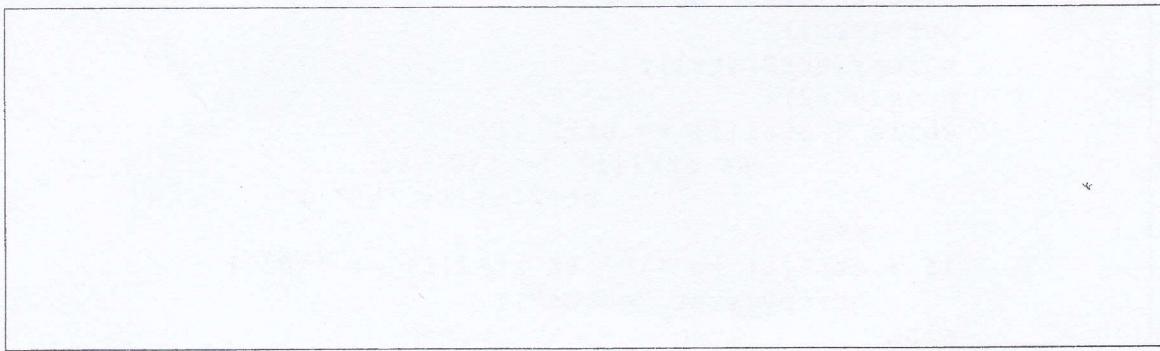
```
#include <stdio.h>
#include <stdbool.h>
int main ()
{
    int i = 0;
    char stat[10];
    char str1[50] ="abc";
    char str2[50] ="abc";
    strcat(str1,"abcabc");
    puts(str1);
    strcpy(str2,str1);
    puts(str2);
    while (str1[i] == str2[i]
           && str1[i] != '\0' &&
           str2[i] != '\0')
        ++i;
    if (str1[i] == '\0' && str2[i] == '\0')
        strcpy(stat,"asbcs");
    else
        strcpy(stat,"bsacs");
    puts(stat);
    strcmp(str1, str2)?
        strcpy(stat,"bsacs") : strcpy(stat,"asbcs");
    puts(stat);
    return 0;
}
```



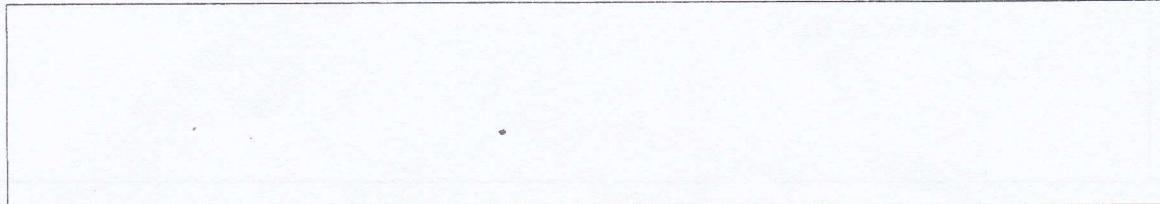
- v. Declare a C structure called `trainTicketInfo` to store the train ticket details given in the following Table 2. (2 Marks)

Ticket Contents	Description
source	Ticket issued railway station
destination	Journey end railway station
cost	Cost of the ticket
id	Ticker identification number

Table 2 – Train Ticker details



- vi. Create a train ticket named as `ticket001` using the C struct type defined in Question 3, part (v). Note that this ticket is printed for travelling from FORT to KANDY. Further, it will cost 150 LKR and ticket identification number is 1436. (2 Marks)



- vii. It is required to add the `issue_date` in to the `trainTicketInfo` struct type defined in Question 3, part (v) as its' fifth data element. `issue_date` has the type of Date. Date structure has three elements namely day, month and year. State the C statement for declaring Date structure type and re-declare the `trainTicketInfo` structure by adding date information. (3 Marks)

- viii. Redefine and initialized the same train ticket (named as `ticket001`) which you created in Question 3 part (vi) and set the `issue_date` as 10/10/2017. (1 Mark)

Question 04

i. What is meant as a stream in C file input output? (2 Marks)

ii. What is the importance of data segment and instruction code segment in the computer memory for executing a computer program? (2 Marks)

iii. Mark TRUE or FALSE for the statements given below. (6 Marks)

(a) The * operator appears before a pointer variable when declaring a pointer variable and dereferencing a pointer variable.	TRUE	FALSE
(b) Each time the pointer is incremented, it points to the next integer and similarly, when a pointer is decremented, it points to the previous integer.	TRUE	FALSE
(c) p and q are pointer variables and p && q return false when p has the memory address of 3455600 and q has the memory address of 3455604.	TRUE	FALSE
(d) Following C statement is succeeded if pointer ptr is null. <code>if (ptr)</code>	TRUE	FALSE
(e) Pointers to arrays generally result in code that uses less memory and executes faster.	TRUE	FALSE
(f) Element of an array are stored in sequential memory locations with the first element in the lowest address.	TRUE	FALSE

- iv. Find the output of following C program using the memory allocation shown in Figure 2.
 (8 Marks)

	200			
Variables	data	*anotherptr		**theptr
Memory Address	0028FF28	0028FF2C		0028FF30

Figure 2 – Memory allocation

```
#include <stdio.h>
int main ()
{
    int **theptr;
    int *anotherptr;
    int data = 200;

    anotherptr = &data;
    theptr = &anotherptr;

    printf("%d\n", **theptr);
    printf("%d\n", *anotherptr);
    printf("%p\n", theptr);
    printf("%p\n", &theptr);
    printf("%p\n", anotherptr);
    printf("%p\n", &anotherptr);
    printf("%p\n", &data);
    printf("%d\n", data);
    return 0;
}
```

- v. Fill the blanks in the following C program which is written to get the summation of values stored in array called values. (7 Marks)

```
#include <stdio.h>
int main (void)
{
    int sum = 0, *ptr;

    int values[10] = { 3, 7, -9, 3, 6, -1, 7, 9, 1, -5 };
    [ ] * [ ] arrayEnd = values + [ ];
    for ( ptr = [ ]; [ ] < arrayEnd; [ ] )
        sum = sum + [ ];
    printf ("The sum is %d\n", sum);
    return 0;
}
```

*** END ***

- v. Fill the blanks in the following C program which is written to get the summation of values stored in array called `values`. (7 Marks)

```
#include <stdio.h>
int main (void)
{
    int sum = 0, *ptr;

    int values[10] = { 3, 7, -9, 3, 6, -1, 7, 9, 1, -5 };
    [ ] * [ ] arrayEnd = values + [ ];
    for ( ptr = [ ]; [ ] < arrayEnd; [ ] )
        sum = sum + [ ];
    printf ("The sum is %d\n", sum);
    return 0;
}
```

*** END ***