

CONFIDENTIAL



UNIVERSITI TUN HUSSEIN ONN MALAYSIA

**FINAL EXAMINATION
SEMESTER II
SESSION 2017/2018**

COURSE NAME : DATA STRUCTURE AND ALGORITHMS
COURSE CODE : BIT 10703
PROGRAMME CODE : BIT
EXAMINATION DATE : JUNE / JULY 2018
DURATION : 3 HOURS
INSTRUCTIONS : A) ANSWER ALL QUESTIONS
B) PLEASE WRITE YOUR ANSWERS IN THIS QUESTION BOOKLET

THIS QUESTION PAPER CONSISTS OF THIRTEEN (13) PAGES

CONFIDENTIAL

TERBUKA

Q1 Answer Q1(a) - Q1(b) based on the information given in Figure Q1.

```
#include <stdio.h>
#include <string.h>

struct EmpList{
    char name[20];
    double salary;
    struct EmpList *next;};

typedef struct EmpList EmpList;

int main()
{
    EmpList *front = NULL;
    EmpList e1, e2, e3, e4, e5;

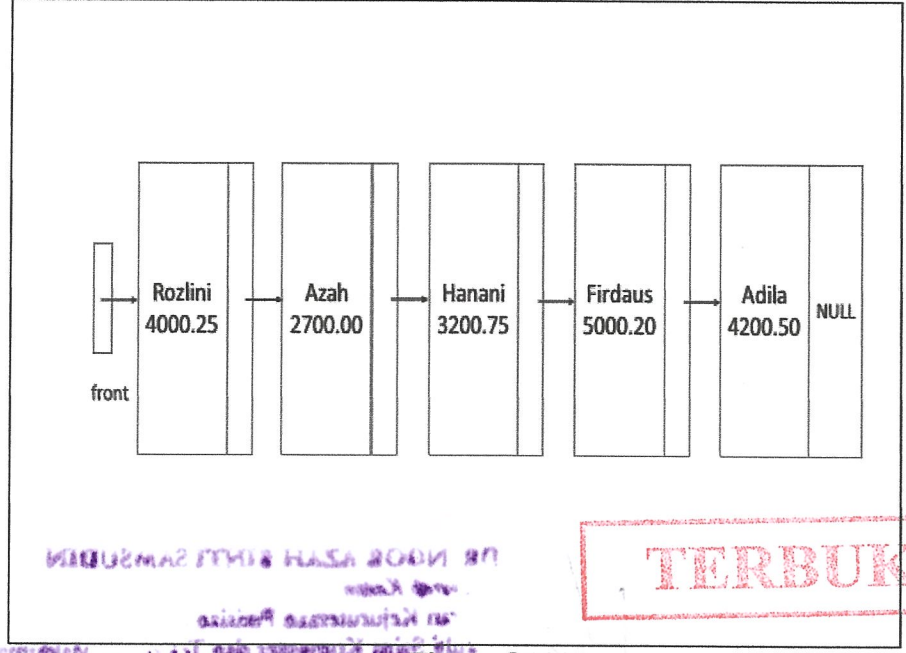
    strcpy (e1.name,"Adila");
    strcpy (e2.name,"Azah");
    strcpy (e3.name,"Firdaus");
    strcpy (e4.name,"Hanani");
    strcpy (e5.name,"Rozlini");

    e1.salary = 4200.50;
    e2.salary = 2700.00;
    e3.salary = 5000.20;
    e4.salary = 3200.75;
    e5.salary = 4000.25;

    return 0;}
```

Figure Q1

(a) Write programming statements to establish the linked list in Figure Q1(a). (10 marks)



TERBUKA

Figure Q1(a)

Answer:

- (b) Write a program fragment to determine and display information of the employee with minimum salary from the linked list established in Q1(a).

(15 marks)

Answer:

TERBUKA

KELOMPOK ITI HANA KODM 379
KEMENTERIAN KOMUNIKASI
KEMENTERIAN PERKAWINAN
KEMENTERIAN KESEHATAN
KEMENTERIAN PERTANIAN
KEMENTERIAN PERUMAHAN DAN KAWASAN
KEMENTERIAN PERUMAHAN DAN KAWASAN

Q2 Answer **Q2(a)** - **Q2(c)** based on the information given in **Figure Q2(a)** and **Figure Q2(b)**.

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

struct NumList{
    double num;
    struct NumList *next;};

typedef struct NumList NumList;

int main()
{
    NumList *head = NULL, *p1, *p2;
    double value, numArray[]={100.4,50.7,87.9,95.2,29.3};
    int i;

    for (i=0; i<5; i++){
        value = numArray[i];

        p1 = malloc(sizeof(NumList));

        if (p1!=NULL){
            p1->num = value*100;
            p1->next = NULL;}

        if (head == NULL)
            head = p1;
        else{
            p2 = head;
            while (p2->next!=NULL)
                p2 = p2->next;

            if (p2->next == NULL)
                p2->next = p1;}
    }

    p2 = head;
    head = head->next;
    free(p2);

    p2 = head;
    while (p2!=NULL){
        printf("%.2lf ", p2->num);
        p2 = p2->next;}

    return 0;
}
```

TERBUKA

Figure Q2(a)

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

struct NumList{
    double num;
    struct NumList *next;};

typedef struct NumList NumList;

int main()
{
    NumList *head = NULL, *p1, *p2;
    double value, numArray[]={100.4,50.7,87.9,95.2,29.3};
    int i;

    for (i=0; i<5; i++){
        value = numArray[i];

        p1 = malloc(sizeof(NumList));

        if (p1!=NULL){
            p1->num = value*10;
            p1->next = NULL;};

        if (head == NULL)
            head = p1;
        else{
            p1->next = head;
            head = p1;};
    }

    p2 = head;
    head = head->next;
    free(p2);

    p2 = head;
    while (p2!=NULL){
        printf("%.2lf ", p2->num);
        p2 = p2->next;};

    return 0;
}
```

Figure Q2(b)

OR MORE KEAR SINTI SAMUDIN
Kantor Kepala
Jabatan Koordinator Pesisir
Pusat Sains Kelautan dan
Perikanan
Jalan Pesisir 1
Tanjung Pagar
80132 Palembang, Sumatera Selatan

TERBUKA

Q3 Answer Q3(a) – Q3(d) based on Figures Q3(a)-Q3(c).

```
#include <stdio.h>
#define SIZE 10

int main()
{
    int value[SIZE] = {122,105,167,198,145,132,201,199,173,110};
    void function1(int[],int);

    function1(value,SIZE);

    return 0;
}

void function1(int list[],int size)
{
    int i,j,k,temp;

    for(i=0; i<(size-1); i++)
    {
        for(j=1; j<size; j++)
        {
            if (list[j]<list[j-1])
            {
                temp = list[j];
                list[j] = list[j-1];
                list[j-1] = temp;
            }
        }

        printf("\n value: ");
        for(k=0; k<size; k++)
            printf("  %d", list[k]);
    }
}
```

Figure Q3(a)**TERBUKA**

DR. NOOR AZAH BINTI SAMUDIN
Penerbit Kertas
Fakulti Kejuruteraan Elektronik
Universiti Teknikal Malaysia Melaka
Jalan Hang Tuah, 76100 Durian Tunggal, Melaka

CONFIDENTIAL

```
#include <stdio.h>
#define SIZE 10

int main()
{
    int value[SIZE] = {232,15,77,108,55,42,111,109,83,20};
    void function2(int[],int);

    function2 (value,SIZE);

    return 0;
}

void function2(int list[],int size)
{
    int i,j,k,valueM,idxM,temp;

    for(i=0; i<(size-1); i++)
    {
        valueM = list[i];
        idxM = i;
        for(j=i+1; j<size; j++)
        {
            if (list[j]<valueM)
            {
                valueM = list[j];
                idxM = j;
            }
        }

        if (valueM<list[i])
        {
            temp = list[i];
            list[i] = valueM;
            list[idxM] = temp;
        }

        printf("\n value: ");
        for(k=0; k<size; k++)
            printf("  %d", list[k]);
    }
}
```

Figure Q3(b)

TERBUKA

UNIVERSITY OF MALAYA
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL ENGINEERING
LABORATORY OF CONTROL SYSTEMS
NO. 10, JALAN KEMENTERIAN
62500 PUTRAJAYA, MALAYSIA


```
#include <stdio.h>
#define SIZE 10
#define TRUE 1
#define FALSE 0

int main()
{
    int value[SIZE] = {55, 60, 72, 82, 95, 117, 123, 148, 149,
151};
    void function3(int[],int,int);

    function3(value,SIZE,149);

    return 0;
}

void function3(int list[],int size,int num)
{
    int i,check,idx1,idx2,idxM;

    i = -1;
    check = FALSE;
    idx1 = 0;
    idx2 = size-1;
    while (idx1 <= idx2 && !check)
    {
        idxM = (int)((idx1+idx2)/2);
        if (num == list[idxM])
        {
            check = TRUE;
            i = idxM;
        }
        else if (num>list[idxM])
            idx1 = idxM+1;
        else
            idx2 = idxM-1;

        printf("\nValue: %d    %d",idxM,list[idxM]);
    }

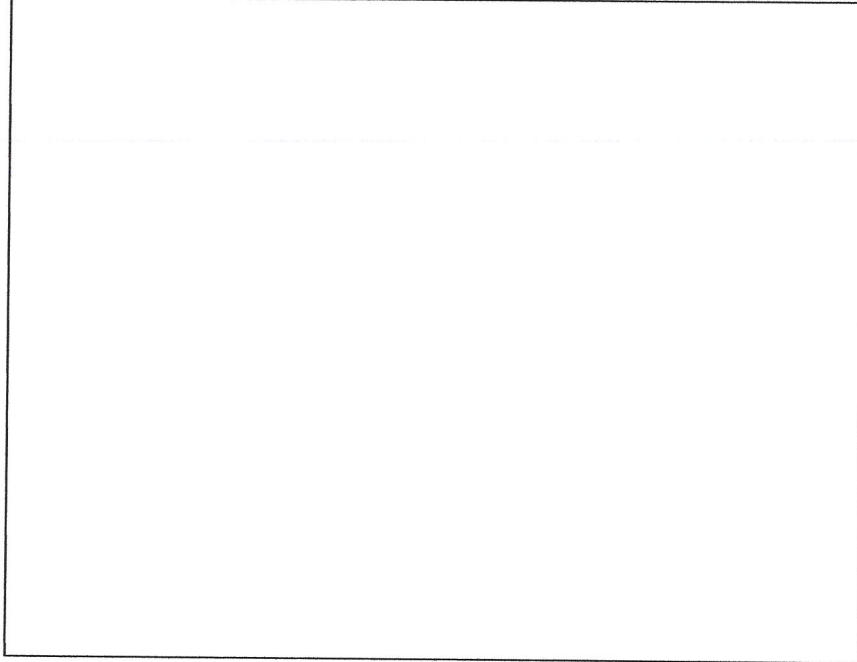
    if (i>-1)
        printf("\nOutput value %d",i);
    else
        printf("\nEnd of program");
}
```

Figure Q3(c)**TERBUKA**

(a) Determine the output for **Figure Q3(a)**.

(9 marks)

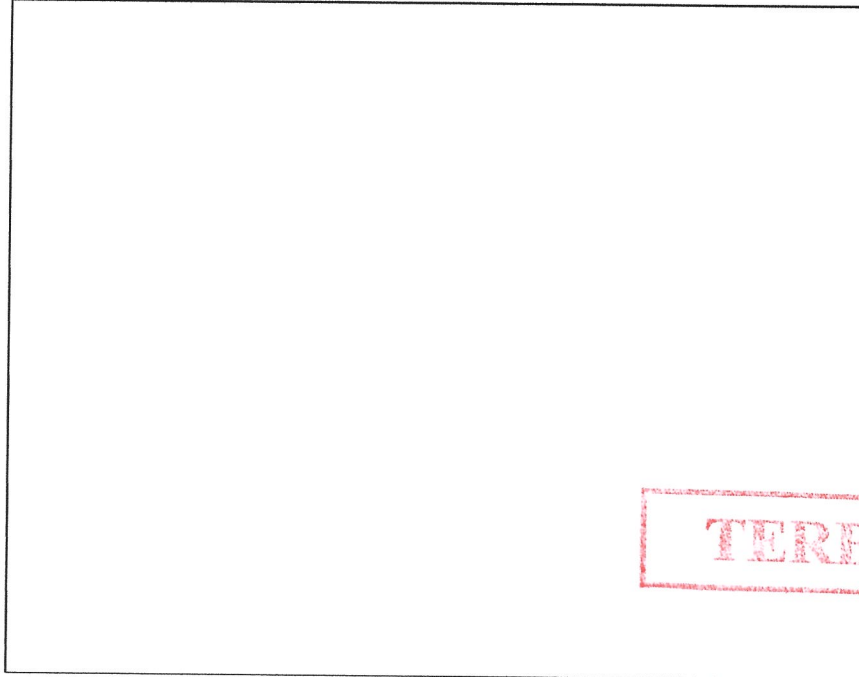
Answer:



(b) Determine the output for **Figure Q3(b)**.

(9 marks)

Answer:



TERBUKA

DR. MOHD ASHRI BINI BAKRUDIN
Penyarah Kanan
Jurusan Kejuruteraan Perisian
Fakulti Sains Komputer dan Teknologi
Matrikulasi

(c) Determine the output for **Figure Q3(c)**.

(9 marks)

Answer:

--

(d) Name the algorithm for every function in **Figures Q3(a) - Q3(c)**.

(3 marks)

Functions	Answer
function1	
function2	
function3	

TERBUKA

11

Q4 Perform traversal algorithms for the tree in **Figure Q4**.

(15 marks)

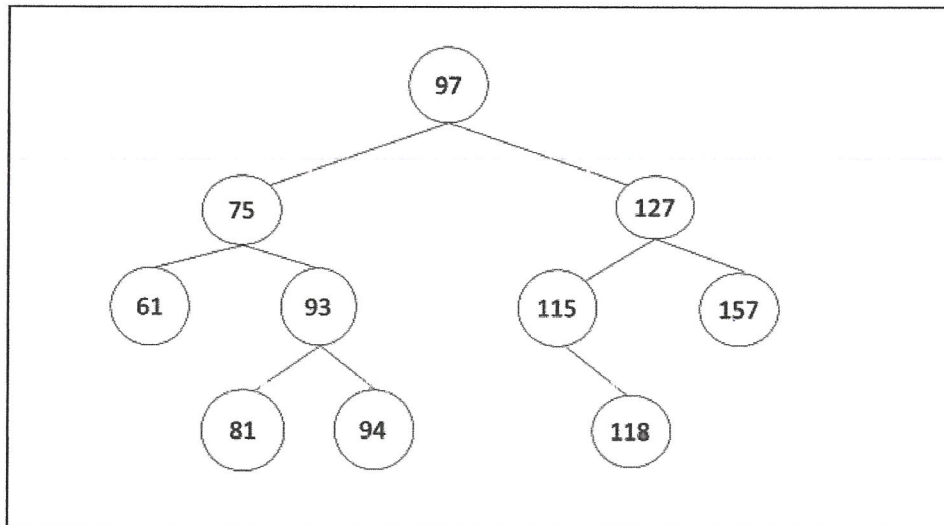


Figure Q4

Traversal algorithms	Answer
Preorder	
Inorder	
Postorder	

TERBUKA

INSTITUT TEKNOLOGI SEPTEMBER SOEDIRO
Jalan Karawang 101, Gedung 101, Kampus Teknik
Sepuluh Nopember, Surabaya 60115, Indonesia
Telp. (031) 5931500, Faks. (031) 5931501

- Q5** A function call to perform a linear search is included in the main program of **Figure Q5**. If a value is found, the linear search function shall return the position of the value in the array. Otherwise, -1 is returned from the function. Complete **Figure Q5** with implementation of the linear search algorithm.

(10 marks)

```
#include <stdio.h>
#define SIZE 7
#define TRUE 1
#define FALSE 0

int main()
{
    int nums[SIZE] = {155, 60, 272, 820, 95, 217, 153};
    int position, num;
    int linearSearch(int [],int,int);

    printf("\nEnter a value that you want to search: ");
    scanf("%d",&num);
    position = linearSearch(nums,SIZE,num);

    if (position>0)
        printf("\nThe value is found at index %d",position);
    else
        printf("\nThe value is not found.");

    return 0;
}

int linearSearch(int list[],int size,int value)
{
    int i = 0, index = -1, found = FALSE;
```

Answer:

TERBUKA

Figure Q5

END OF QUESTION -