Indian Institute of Technology Patna
CS102: Programming and Data Structure

EndSem
Time: 3 hours

27th April 2018
Full Marks 100

Name (in Capital): _____

Roll No: _____

Signature: _____

Invegilator's Signature: _____

Instruction:

- Write your name and Roll No and sign in question paper.

- Assume that there will be no error due to non-inclusion of header.

- All question belongs to question number 1 are of MCQ type with four options. Please choose (put a tick) exactly one option. Question will not be evaluated, if there are overwritten choices. 25% marks will be deducted for each wrong answer in MCQ type question.

- In question number 2 you are asked to write output/error of a code/code snippet. There is no negative marking.

- In question number 3 you are asked to fill in the blanks. There is no negative marks for wrong answer.

For Office Use:

Question1(46)	Question2(33)	Question3(21)	Total(100)

1. Choose the correct option. All questions carry equal marks. [23*2=46]

    (a) Consider an implementation of unsorted circular linked list. Suppose it has its representation with a head pointer only. Given the representation, which of the following operation can be implemented in constant time?

    i) Insertion at the front of the linked list ii) Insertion at the end of the linked list iii) Deletion of the front node of the linked list iv) Deletion of the end node of the linked list

        i. I and II;
        ii. I and III;
        iii. I, II, III and IV;
        iv. None

    (b) Consider the following code snippet

```
struct node
{
int data;
struct node * next;
}
typedef struct node NODE;
NODE *ptr;
```

Which of the following c code is used to create new node?

        i. ptr=(NODE*)malloc(sizeof(NODE));
        ii. ptr=(NODE*)malloc(NODE);
        iii. ptr=(NODE*)malloc(sizeof(NODE*));
        iv. ptr=(NODE)malloc(sizeof(NODE)):

    (c) The following C function takes a single-linked list of integers as a parameter and rearranges the elements of the list. The function is called with the list containing the integers 1, 2, 3, 4, 5, 6, 7 in the given order. What will be the contents of the list after the function completes execution?

```
struct node
{
  int value;
  struct node *next;
};
void rearrange(struct node *list)
{
  struct node *p, * q;
  int temp;
  if ((!list) || !list->next)
      return;
  p = list;
  q = list->next;
  while(q)
  {
      temp = p->value;
      p->value = q->value;
      q->value = temp;
      p = q->next;
      q = p?p->next:0;
  }
}
```

        i. 1,2,3,4,5,6,7
        ii. 2,1,4,3,6,5,7
        iii. 1,3,2,5,4,7,6
        iv. 2,3,4,5,6,7,1

    (d) What would be the output of the following code snippet:

```
struct item
{
  int data;
  struct item * next;
};

int f(struct item *p)
{
  return (
```

```
            (p == NULL) ||
            (p->next == NULL) ||
            (( P->data <= p->next->data) &&
            f(p->next))
            );
}
```

    i. the list is empty or has exactly one element

    ii. the elements in the list are sorted in non-decreasing order of data value

    iii. the elements in the list are sorted in non-increasing order of data value

    iv. not all elements in the list have the same data value

(e) The result evaluating the postfix expression 10 5 + 60 6 / * 8 - is

    i. 284

    ii. 213

    iii. 142

    iv. 71

(f) The postfix form of the expression (A+ B)*(C*D-E)*F / G is?

    i. AB+ CD*E - FG /**

    ii. AB + CD* E - F **G /

    iii. AB + CD* E - *F *G /

    iv. AB + CDE * - * F *G /

(g) Consider the following pseudo code. Assume that IntQueue is an integer queue. What does the function fun do?

```
void fun(int n)
{
    IntQueue q;
    q.enqueue(0);
    q.enqueue(1);
    for (int i = 0; i < n; i++)
    {
        int a = q.dequeue();
        int b = q.dequeue();
        q.enqueue(b);
        q.enqueue(a + b);
        ptint(a);
    }
}
```

    i. Prints numbers from 0 to n-1

    ii. Prints numbers from n-1 to 0

    iii. Prints first n Fibonacci numbers

    iv. Prints first n Fibonacci numbers in reverse order.

(h) Choose the right output for the following code.

```
#include <stdio.h>
struct student
{
    char *name;
};
```

```
void main()
{
    struct student s, m;
    s.name = "Patna";
    m = s;
    printf("%s%s", s.name, m.name);
}
```

    i. Compile time error

    ii. Patna Patna

    iii. Patna any-random-string

    iv. Junk values

(i) what would be the output of the following code snippet?

```
#includestdio.h
int main()
{
    struct address
    {
        char city[] = "Bihta";
        int pin = 106;
    };
    struct address *ptr;
    printf("%s ", ptr->city);
    printf("%d", ptr->pin);
    return 0;
}
```

    i. Bihta 106

    ii. Compile Error

    iii. Runtime Error

    iv. Bihta

(j) What would be printed if following code is executed.

```
#include<stdio.h>

int main()
{
    int a = 5;
    switch(a)
    {
    default:
        a = 4;
    case 6:
        a--;
    case 5:
        a = a+1;
    case 1:
        a = a-1;
    }
    printf("%d \n", a);
    return 0;
}
```

    i. 4

    ii. 6

    iii. 5

    iv. 3

(k) Which combination of the integer variables x, y and z makes the variable a get the value 4 in the following expression?

```
a = ( x > y ) ? (( x > z ) ? x : z) :
(( y > z ) ? y : z )
```

   i. x = 5, y = 4, z = 5

   ii. x = 6, y = 3, z = 5

   iii. x = 6, y = 5, z = 3

   iv. x = 3, y = 4, z = 2

(l) What would be the output of following code?

```
#include<stdio.h>
int main()
{
    int n;
    for (n = 9; n!=0; n--)
        printf("n = %d", n--);
    return 0;
}
```

   i. 9 7 5 3 1

   ii. compilation error

   iii. Runtime error

   iv. infinite loop

(m) The concatenation of two lists is to be performed in constant time. Which of the following implementations of a list should be used?

   i. single linked list

   ii. double linked list

   iii. circular double linked

   iv. none of the above

(n) What would be the output of the following code?

```
#include <stdio.h>

int main()
{
    int arr[5];

    // Assume that base address of arr is 2000
    //and size of integer is 32 bit

    arr++;
    printf("%u", arr);

    return 0;
}
```

   i. 2002

   ii. 2004

   iii. 2020

   iv. None of the above

(o) Whats the meaning of following declaration in C language?

$$int\ (*p)[5];$$

   i. It will result in compile error because there shouldn't be any parenthesis i.e. "int *p[5]" is valid.

   ii. p is a pointer to 5 integers.

   iii. p is a pointer to integer array.

   iv. p is an array of 5 pointers to integers.

   v. p is a pointer to an array of 5 integers.

(p) Which of the following traversals is sufficient to construct BST from given traversals 1) Inorder 2) Preorder 3) Postorder.

   i. Any one of the given three traversals is sufficient

   ii. Either 2 or 3 is sufficient

   iii. Both 2 and 3 are required

   iv. Both 1 and 3 are required

(q) The preorder traversal sequence of a binary search tree is 30, 20, 10, 15, 25, 23, 39, 35, 42. Which one of the following is the postorder traversal sequence of the same tree?

   i. 10, 20, 15, 23, 25, 35, 42, 39, 30

   ii. 15, 10, 25, 23, 20, 42, 35, 39, 30

   iii. 15, 20, 10, 23, 25, 42, 35, 39, 30

   iv. 15, 10, 23, 25, 20, 35, 42, 39, 30

(r) Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28. How many heapify operations have been performed on root of heap?

   i. 1

   ii. 2

   iii. 3 or 4

   iv. 5 or 6

(s) Consider a binary max-heap implemented using an array. Which one of the following array represents a binary max-heap?

   i. 25,12,16,13,10,8,14

   ii. 25,12,16,13,10,8,14

   iii. 25,14,16,13,10,8,12

   iv. 25,14,12,13,10,8,16

(t) Consider any array representation of an n element binary heap where the elements are stored from index 1 to index n of the array. For the element stored at index i of the array (i <= n), the index of the parent is

   i. i - 1

   ii. floor(i/2)

   iii. ceiling(i/2)

   iv. (i+1)/2

(u) Consider a max heap, represented by the array: 40, 30, 20, 10, 15, 16, 17, 8, 4. Now consider that a value 35 is inserted into this heap. After insertion, the new heap is

   i. 40, 30, 20, 10, 15, 16, 17, 8, 4, 35

   ii. 40, 35, 20, 10, 30, 16, 17, 8, 4, 15

   iii. 40, 30, 20, 10, 35, 16, 17, 8, 4, 15

   iv. 40, 35, 20, 10, 15, 16, 17, 8, 4, 30

(v) In C, static storage class cannot be used with:

   i. Global variable

   ii. Function parameter

   iii. Local variable

iv. None of the above

(w) Choose the best statement with respect to following three program snippets.

```
/*Program Snippet 1 with for loop*/
for (i = 0; i < 10; i++)
{
    /*statement1*/
    continue;
    /*statement2*/
}


/*Program Snippet 2 with while loop*/
i = 0;
while (i < 10)
{
    /*statement1*/
    continue;
    /*statement2*/
    i++;
}


/*Program Snippet 3 with do-while loop*/
i = 0;
do
{
    /*statement1*/
    continue;
    /*statement2*/
    i++;
}while (i < 10);
```

  i. All the loops are equivalent i.e. any of the three can be chosen and they all will perform exactly same.
  ii. continue can't be used with all the three loops in C.
  iii. After hitting the continue; statement in all the loops, the next expression to be executed would be controlling expression (i.e. i < 10) in all the 3 loops.
  iv. None of the above is correct.

2. Write output/error. [33]

(a) Go through the following code and write the output [2]

```
#include<stdio.h>
struct complex{
    int real;
    int imaginary;};
typedef union{
int a;
    struct complex c;
} U;

int main(){
U x;
x.a=10;
x.c.real=15;
    x.c.imaginary=20;
```

```
printf("x.a=%d",x.a);
    return 0;
}
```

Output:_____

(b) Assume you are asked to find Huffman coding for the following text "I am Ms Mississippi". Assume that between two words there is exactly one space. Priority queue used for building the code is min-priority queue [Higher priority is given to lower priority value]. When there is a tie in priority value, it is resolved using following policy - priority('a') > priority('A') > priority('b') > priority('B') and so on. Priority of space is lowest when there is a tie in priority values. Priority of an older node is higher than priority of comparatively new node. Find code for all letters and cost (in terms of bit) to code the given text. Note: sp stands for space [8 X 0.5 +1 = 5]

Table 1: Huffman Code

Char	code	char	code
I		i	
a		s	
M		p	
m		sp	

Cost to encode=_____

(c) Write output when following code is executed and as input 4 1 2 3 4 are entered. [2]

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

struct s{
int x;
struct s *next;
};
typedef struct s *node;

node head=NULL;

node createnode(int x){
node n=(node)malloc(sizeof(struct s));
n->x=x;
n->next=NULL;
return n;
}

void insert(int x){
node n=createnode(x);
if (head==NULL){
        head=n;
        }
else {
        n->next=head;
        head=n;
}}
```

```
node find(){
node p,q;
p=head;
q=head;
if (p==NULL)
        return NULL;

while(1){
        if(q->next!=NULL){
                q=q->next;
                if(q->next!=NULL){
                        q=q->next;
                        p=p->next;
                }
                else {
                        return p;
                }
        } else {
                return p;
        }
}
}
```

```
int main(){
node p,q;
int i,n,x;
scanf("%d",&n);
for(i=0;i<n;i++){
        scanf("%d",&x);
        insert(x);
}
p=find();
printf("%d",p->x);
return 0;
}
```

Output=_____

(d) If following code snippet is executed on two singly
linked list i) $1 \to 2 \to 3 \to 4$ ii) $4 \to 3 \to 2 \to 1$
what would be output (final list)/error? Assume swap
function swaps two integers. And this will be called
from main function. [2+2=4]

```
struct s{
int x;
struct s *next;
};
typedef struct s *node;

void exchange(){
node p,q;
p=head;
q=head;
while(p->next!=NULL){
while(q->next!=NULL){
        q=q->next;
        if(q->x%p->x==0){
                swap(&q->x,&p->x);
                break;
```

```
        }
        else
                q=q->next;
}
p=p->next;
q==p;
}
}
```

Output=i)_____

ii)_____

(e) Write output/error [2]

```
int main( )
{
static int a[ ] = {0,1,2,3,4};
int *p[ ] = {a,a+1,a+2,a+3,a+4};
int **ptr = p;
ptr++;
printf(\n %d %d %d, ptr-p, *ptr-a, **ptr);
*ptr++;
printf(\n %d %d %d, ptr-p, *ptr-a, **ptr);
*++ptr;
printf(\n %d %d %d, ptr-p, *ptr-a, **ptr);
++*ptr;
printf(\n %d %d %d, ptr-p, *ptr-a, **ptr);
    return 0;
}
```

Output=_____

(f) Write output/error [2]

```
int main( )
{
void *vp;
char ch = g, *cp = goofy;
int j = 20;
vp = &ch;
printf(%c, *(char *)vp);
vp = &j;
printf(%d,*(int *)vp);
vp = cp;
printf(%s,(char *)vp + 3);
    return 0;
}
```

Output=_____

(g) On executing the below program what will be the
contents of 'target.txt' file if the source file contains
a line "To err is human"? [2]

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

```c
int main()
{
    int i, fss;
    char ch, source[20] = "source.txt",
    target[20]="target.txt", t;
    FILE *fs, *ft;
    fs = fopen(source, "r");
    ft = fopen(target, "w");
    while(1)
    {
        ch=getc(fs);
        if(ch==EOF)
            break;
        else
        {
            fseek(fs, 4L, SEEK_CUR);
            fputc(ch, ft);
        }
    }
    return 0;
}
```

Output= _____

(h) Write output/error [2]

```c
#include <stdio.h>
enum day {sunday = 1, monday, tuesday = 5,
          wednesday, thursday = 10, friday,
          saturday};

int main()
{
    printf("%d %d %d %d %d %d %d", sunday, monday,
    tuesday, wednesday, thursday, friday,
    saturday);

    return 0;
}
```

Output= _____

(i) Write output/error [2]

```c
int main()
{
char *str="A%%B";
printf("A%%B ");
printf("%s\n",str);
return 0;
}
```

Output= _____

(j) Write Output/error [2]

```c
#include <stdio.h>
int * gPtr;
int main()
{
    int * lPtr = NULL;
```

```c
    if(gPtr == lPtr)
    {
        printf("Equal!");
    }
    else
    {
        printf("Not Equal");
    }
    return 0;
}
```

Output= _____

(k) Write output/error [2]

```c
#include <stdio.h>
int main()
{
    static int i=5;
    if(--i){
        main();
        printf("%d ",i);
    }
    return 0;
}
```

Output= _____

(l) Write output/error [2]

```c
#include <stdio.h>
int fun()
{
    static int num = 16;
    return num--;
}

int main()
{
    for(fun(); fun(); fun())
        printf("%d ", fun());
    return 0;
}
```

Output= _____

(m) Write output/error [2]

```c
#include <stdio.h>
int fun(int n)
{
    static int s = 0;
    s = s + n;
    return (s);
}

int main()
{
    int i = 10, x;
    while (i > 0)
    {
```

6

```
        x = fun(i);
        i--;
    }
    printf ("%d ", x);
    return 0;
}
```

Output= _____

(n) Write output/error [2]

```
#include<stdio.h>

int main()
{
   int a[] = {1, 2, 3, 4, 5, 6};
   int *ptr = (int*)(&a+1);
   printf("%d ", *(ptr-1) );
   return 0;
}
```

Output= _____

3. Fill in the blank. Each blank carries equal marks [14*1.5=21]

(a) The following function reverse() is supposed to reverse a singly linked list. There is one line missing at the end of the function.

```
/* Link list node */
struct node
{
    int data;
    struct node* next;
};

/* head_ref is a double pointer which points to
head (or start) pointer   of linked list */
static void reverse(struct node** head_ref)
{
    struct node* prev    = NULL;
    struct node* current = *head_ref;
    struct node* next;
    while (current != NULL)
    {
        next    = current->next;
        current->next = prev;
        prev = current;
        current = next;
    }
    /*ADD A STATEMENT HERE*/

    ------------------------------
}
```

(b) Following code snippet tries to remove duplicate elements from a sorted list. However, program is damaged and you are asked to repair(fill in the blanks) so that it works.

```
struct s{
int x;
struct s *next;
};
typedef struct s *node;

void  remove_duplicate(){
node p,q;
p=head;
if(p==NULL)
        return;
while(p!=NULL){
        q=p->next;
____ (q!=NULL && q->x==p->x){
                q=q->next;
        }

        ------------------

        ------------------

}
}
```

(c) Find the following code snippet. It tries to remove multiplier from list. Like if it finds a node with value x at the starting of the list, it would scan the entire list and remove all elements with value $k \times x$ (where k is any integer constant). Then it checks for the next element. If the initial list is $2 \to 3 \to 4 \to 5 \to 7 \to 8 \to 9$, then it would become $2 \to 3 \to 5 \to 7$ after running following code. However, program is damaged and you are asked to repair(fill in the blanks) so that it works.

```
struct s{
int x;
struct s *next;
};
typedef struct s *node;

void remove_multiplier(){
node p,q,r;
p=head;
if(head==NULL)
        return;
else{
        while(p!=NULL){
                r=p;
                q=p->next;
                while(q!=NULL){
                        if(q->x%p->x==0){
                                r->next=q->next;
                                free(q);
                                q=r->next;
                        } else {

                        ------------------

                        ------------------
}
```

```
            }
                p=p->next;
    }
    }
    return;
    }
```

(d) Suppose a stack is implemented with an array of n elements. The condition of checking whether the stack is empty and full are _____ and

_____

(e) Suppose the numbers 7, 5, 1, 8, 3, 6, 0, 9, 4, 2 are inserted in that order into an initially empty binary search tree. The binary search tree uses the usual ordering on natural numbers. The in-order traversal sequence of the resultant tree is

_____

(f) Consider the frequency of characters as a, b, c, d, e, f are 5, 9, 12, 13, 16, 45. The Huffman codes of the characters are:_____.

(g) Fill in the blank .

```
/* C Program to Find the Number of Lines
 in a Text File */
#include <stdio.h>

int main()
{
    FILE *fileptr;
    int count_lines = 0;
    char filechar[40], chr;

    printf("Enter file name: ");
    scanf("%s", filechar);
    fileptr = fopen(filechar, "r");
    chr = getc(fileptr);
    while (chr != EOF)
    {

        if (_____)
        {
            --------------------;
        }

        --------------------;
    }
    fclose(fileptr); //close file.
    printf("There are %d lines \n", count_lines);
    return 0;
}
```

(h) Fill in the blank

```
/*
 * C Program to find HCF of a given Number
```

```
 using Recursion
 */
#include <stdio.h>

int hcf(int, int);

int main()
{
    int a, b, result;

    printf("Enter the two numbers to find
    their HCF: ");
    scanf("%d%d", &a, &b);
    result = hcf(a, b);
    printf("The HCF of %d and %d is %d.\n",
    a, b, result);
}

int hcf(int a, int b)
{
    while (a != b)
    {
        if (a > b)
        {
            return --------;
        }
        else
        {
            return ---------;
        }
    }
    return a;
}
```