Some sample C program for practice

(Solve the lab assignments sheet 3 and 4)

- ✓ Read the all topics of syllebus from reference books
- ✓ Solve the previous semesters questions
- ✓ Follow the all class lectures taken by me
- ✓ Solve the all problems discuss in class lectures

Saifur Rahaman
Assistant Professor
Dept. of CSE, IIUC

1. Factorial Number calculation:

```
#include <stdio.h>
int main()
{
  int c, n, fact = 1;
  printf("Enter a number to calculate it's factorial\n");
  scanf("%d", &n);
  for (c = 1; c <= n; c++)
    fact = fact * c;
  printf("Factorial of %d = %d\n", n, fact);
  return 0;
}
        ////
#include <stdio.h>
long factorial(int);
int main()
{
  int number;
  long fact = 1;
  printf("Enter a number to calculate it's factorial\n");
  scanf("%d", &number);
  printf("%d! = %ld\n", number, factorial(number));
  return 0;
}

long factorial(int n)
{
  int c;
  long result = 1;
  for (c = 1; c <= n; c++)
    result = result * c;
  return result;
}

//recursion
#include<stdio.h>
long factorial(int);
int main()
{
  int n;
  long f;
  printf("Enter an integer to find factorial\n");
  scanf("%d", &n);
  if (n < 0)
    printf("Negative integers are not allowed.\n");
  else
  {
    f = factorial(n);
    printf("%d! = %ld\n", n, f);
  }
  return 0;
}

long factorial(int n)
{
  if (n == 0)
    return 1;
  else
    return(n * factorial(n-1));
}
```

2. Fibonacci Number generation:

```
#include<stdio.h>
int main()
{
   int n, first = 0, second = 1, next, c;
   printf("Enter the number of terms\n");
   scanf("%d",&n);
   printf("First %d terms of Fibonacci series are :-\n",n);
   for ( c = 0 ; c < n ; c++ )
   {
      if ( c <= 1 )
         next = c;
      else
      {
         next = first + second;
         first = second;
         second = next;
      }
      printf("%d\n",next);
   }
   return 0;
}

int Fibonacci(int n)
{
   if ( n == 0 )
      return 0;
   else if ( n == 1 )
      return 1;
   else
      return ( Fibonacci(n-1) + Fibonacci(n-2) );
}
```

```
#include <stdio.h>
int main()
{
    int t1=0, t2=1, nextTerm = 0, n;
    printf("Enter a positive integer: ");
    scanf("%d",&n);
    // displays the first two terms which is always 0 and 1
    printf("Fibonacci Series: %d, %d, ", t1, t2);
    nextTerm = t1+t2;
    while(nextTerm < n)
    {
        printf("%d, ",nextTerm);
        t1 = t2;
        t2 = nextTerm;
        nextTerm = t1+t2;
    }
    return 0;
}
```

3. Piglatin generator: Class lecture and book example

```
#include <stdio.h>
#include <string.h>
int main()
{
    char w[100]={0};
    while((scanf("%s",w))){
        if(strcmp(w,"end")==0)
        break;
        int l,i;
        char p[100]={0};
        l=strlen(w);
       // strcpy(p,w);
        for(i=0;i<l-1;i++){
            p[i]=w[i+1];
        }
        p[l-1]=w[0];
        p[l]='a';
        printf("%s ",p);
    }
      printf("\n");
    return 0;
}
```

4. Prime Number generator:

C program for prime number or not

```
#include<stdio.h>
int main()
{
   int n, c = 2;
   printf("Enter a number to check if it is prime\n");
   scanf("%d",&n);
   for ( c = 2 ; c <= n - 1 ; c++ )
   {
      if ( n%c == 0 )
      {
         printf("%d is not prime.\n", n);
         break;
      }
   }
   if ( c == n )
      printf("%d is prime.\n", n);
   return 0;
}

int check_prime(int a)
{
   int c;
// for ( c = 2 ; c <= (int)sqrt(n) ; c++ )
   for ( c = 2 ; c <= a - 1 ; c++ )
   {
      if ( a%c == 0 )
         return 0;
   }
   if ( c == a )
      return 1;
}
```

5. GCD and LCM:

```
#include <stdio.h>
int main() {
  int a, b, x, y, t, gcd, lcm;
  printf("Enter two integers\n");
  scanf("%d%d", &x, &y);
  a = x;
  b = y;
  while (b != 0) {
    t = b;
    b = a % b;
    a = t;
  }
  gcd = a;
  lcm = (x*y)/gcd;
  printf("Greatest common divisor of %d and %d = %d\n", x, y, gcd);
  printf("Least common multiple of %d and %d = %d\n", x, y, lcm);
  return 0;
}

//// Recursion

long gcd(long a, long b) {
  if (b == 0) {
    return a;
  }
  else {
    return gcd(b, a % b);
  }
}
```

6. Number convertion:

Integer to binary convert function :

```
int int_to_bin(unsigned int a,int *bin)
{
 int saifur=0;
 unsigned int c=1;
 int i ;
 for(i=31;i>=0;i--)
 {
  saifur = saifur + (a&c);
  bin[i] = (a&c)?1:0;
  c<<=1;
 }
 return saifur;
}
///
#include <stdio.h>
int main()
{
  int n, c, k;
  printf("Enter an integer in decimal number system\n");
  scanf("%d", &n);
  printf("%d in binary number system is:\n", n);
  for (c = 31; c >= 0; c--)
  {
    k = n >> c;
    if (k & 1)
      printf("1");
    else
      printf("0");
  }
  printf("\n");
  return 0;
}
```

Binary to decimal convert function

```
long binary(char s1[])
{
 long int s,a=0,i,j,m;
 s=strlen(s1);
for(j=0,i=s-1;0<=i;i--,j++)
      {
        m=s1[j]-48;
       a+=(long)(m*pow(2,i));
      }
      return a;
}
```

7. Palindrome:

C program for palindrome without using string functions

```
#include <stdio.h>
#include <string.h>
int main()
{
   char text[100];
   int begin, middle, end, length = 0;
   gets(text);
   while (text[length] != '\0')
      length++;
   end = length - 1;
   middle = length/2;
   for (begin = 0; begin < middle; begin++)
   {
      if (text[begin] != text[end])
      {
         printf("Not a palindrome.\n");
         break;
      }
      end--;
   }
   if (begin == middle)
      printf("Palindrome.\n");
   return 0;
}
```

C program for palindrome

```
#include <stdio.h>
#include <string.h>
int main()
{
   char a[100], b[100];
   printf("Enter the string to check if it is a palindrome\n");
   gets(a);
   strcpy(b,a);
   strrev(b);
   if (strcmp(a,b) == 0)
      printf("Entered string is a palindrome.\n");
   else
      printf("Entered string is not a palindrome.\n");
   return 0;
}
```

Palindrome number program c

```
#include <stdio.h>
int main()
{
   int n, reverse = 0, temp;
   printf("Enter a number to check if it is a palindrome or not\n");
   scanf("%d",&n);
   temp = n;
   while( temp != 0 ) {
      reverse = reverse * 10;
      reverse = reverse + temp%10;
      temp = temp/10;
   }
   if ( n == reverse )
      printf("%d is a palindrome number.\n", n);
   else
      printf("%d is not a palindrome number.\n", n);
   return 0;
}
```

8.   Reverse String and Number:

```
void reverse(char s[])
{
       int sa,i,fur;
       char t;
       fur=strlen(s);
       for(i=0,sa=fur-1;i<fur/2;i++,sa--)
       {      t=s[sa];
              s[sa]=s[i];
              s[i]=t;
       }
}

#include <stdio.h>
#include <string.h>
int main()
{
   char s[100], r[100];
   int n, c, d;
   printf("Input a string\n");
   gets(s);
   n = strlen(s);
   for (c = n - 1, d = 0; c >= 0; c--, d++)
      r[d] = s[c];
   r[d] = '\0';
   printf("%s\n", r);
   return 0;
}

// string function
int main()
{
   char arr[100];
   printf("Enter a string to reverse\n");
   gets(arr);
   strrev(arr);
   printf("Reverse of entered string is \n%s\n",arr);
   return 0;
}

//// Reverse Number
#include <stdio.h>
```

```
int main()
{
int n, reverse = 0;
printf("Enter a number to reverse\n");
scanf("%d", &n);
while (n != 0)
{
reverse = reverse * 10;
reverse = reverse + n%10;
n       = n/10;
}
printf("Reverse of entered number is = %d\n", reverse);
return 0;
}
```

9. File input/output: This program stores a sentence entered by user in a file.

```
#include <stdio.h>
#include <stdlib.h>  /* For exit() function */
int main()
{
   char c[1000];
   FILE *fptr;
   fptr=fopen("program.txt","w");
   if(fptr==NULL){
      printf("Error!");
      exit(1);
   }
   printf("Enter a sentence:\n");
   gets(c);
   fprintf(fptr,"%s",c);
   fclose(fptr);
   return 0;
}
```

///This program reads a string of text from a file.

```
/* Source Code to read a text of string from a file. */
#include <stdio.h>
#include <stdlib.h> /* For exit() function*/
int main()
{
   char c[1000];
   FILE *fptr;
   if ((fptr=fopen("program.txt","r"))==NULL){
       printf("Error! opening file");
       exit(1);         /* Program exits if file pointer returns NULL. */
   }
   fscanf(fptr,"%[^\n]",c);
   printf("Data from file:\n%s",c);
   fclose(fptr);
   return 0;
}
```

C program to display its own source code using __FILE__

```
#include <stdio.h>
int main() {
FILE *fp;
char c;
fp = fopen(__FILE__,"r");
do {

     c = getc(fp);
     putchar(c);

}
while(c != EOF);
fclose(fp);
return 0;
}
```

C program to open a file

C programming code to open a file and to print it contents on screen.

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
char ch, file_name[25];
FILE *fp;
printf("Enter the name of file you wish to see\n");
gets(file_name);
fp = fopen(file_name,"r"); // read mode
if( fp == NULL )
{
      perror("Error while opening the file.\n");
      exit(EXIT_FAILURE);
}
printf("The contents of %s file are :\n", file_name);
while( ( ch = fgetc(fp) ) != EOF )
      printf("%c",ch);

fclose(fp);
return 0;
}
// File Copy

while( ( ch = fgetc(source) ) != EOF )
      fputc(ch, target);
printf("File copied successfully.\n");
fclose(source);
fclose(target);
```

10. Pyramid: class lecture
11. Diamond

```
#include <stdio.h>
int main()
{
  int n, c, k, space = 1;
  printf("Enter number of rows\n");
  scanf("%d", &n);
  space = n - 1;
  for (k = 1; k <= n; k++)
  {
    for (c = 1; c <= space; c++)
      printf(" ");
    space--;
    for (c = 1; c <= 2*k-1; c++)
      printf("*");
    printf("\n");
  }
  space = 1;
  for (k = 1; k <= n - 1; k++)
  {
    for (c = 1; c <= space; c++)
      printf(" ");
    space++;
    for (c = 1 ; c <= 2*(n-k)-1; c++)
      printf("*");
    printf("\n");
  }
  return 0;
}
```

12. Loop: Class lecture and Book examples

13. **Swaping using pointer**

```
#include <stdio.h>
int main()
{
   int x, y, *a, *b, temp;
   printf("Enter the value of x and y\n");
   scanf("%d%d", &x, &y);
   printf("Before Swapping\nx = %d\ny = %d\n", x, y);
   a = &x;
   b = &y;

   temp = *b;
   *b   = *a;
   *a   = temp;
   printf("After Swapping\nx = %d\ny = %d\n", x, y);
   return 0;
}
```

Swapping numbers using call by reference : In this method we will make a function to swap numbers.

```
#include <stdio.h>
void swap(int*, int*);
int main()
{
   int x, y;
   printf("Enter the value of x and y\n");
   scanf("%d%d",&x,&y);
   printf("Before Swapping\nx = %d\ny = %d\n", x, y);
   swap(&x, &y);
   printf("After Swapping\nx = %d\ny = %d\n", x, y);
   return 0;
}

void swap(int *a, int *b)
{
   int temp;
   temp = *b;
   *b   = *a;
   *a   = temp;
}
```

C programming code to swap using bitwise XOR

```
#include <stdio.h>
int main()
{
  int x, y;
  scanf("%d%d", &x, &y);
  printf("x = %d\ny = %d\n", x, y);
  x = x ^ y;
  y = x ^ y;
  x = x ^ y;
  printf("x = %d\ny = %d\n", x, y);
  return 0;
}
```

14. **Array and Matrix:**

Matrix multiplication in c language

```
#include <stdio.h>
int main()
{
  int m, n, p, q, c, d, k, sum = 0;
  int first[10][10], second[10][10], multiply[10][10];
  printf("Enter the number of rows and columns of first matrix\n");
  scanf("%d%d", &m, &n);
  printf("Enter the elements of first matrix\n");
  for (c = 0; c < m; c++)
    for (d = 0; d < n; d++)
      scanf("%d", &first[c][d]);
  printf("Enter the number of rows and columns of second matrix\n");
  scanf("%d%d", &p, &q);
  if (n != p)
    printf("Matrices with entered orders can't be multiplied with each other.\n");
  else
  {
    printf("Enter the elements of second matrix\n");

    for (c = 0; c < p; c++)
      for (d = 0; d < q; d++)
        scanf("%d", &second[c][d]);

    for (c = 0; c < m; c++) {
      for (d = 0; d < q; d++) {
        for (k = 0; k < p; k++) {
          sum = sum + first[c][k]*second[k][d];
        }
        multiply[c][d] = sum;
        sum = 0;
      }
    }

    printf("Product of entered matrices:-\n");
    for (c = 0; c < m; c++) {
      for (d = 0; d < q; d++)
        printf("%d\t", multiply[c][d]);

      printf("\n");
    }
  }
  return 0;
}

// Add matrix

#include <stdio.h>
int main()
{
   int m, n, c, d, first[10][10], second[10][10], sum[10][10];
   printf("Enter the number of rows and columns of matrix\n");
   scanf("%d%d", &m, &n);
   printf("Enter the elements of first matrix\n");
   for (c = 0; c < m; c++)
      for (d = 0; d < n; d++)
         scanf("%d", &first[c][d]);
   printf("Enter the elements of second matrix\n");
   for (c = 0; c < m; c++)
      for (d = 0 ; d < n; d++)
            scanf("%d", &second[c][d]);

   printf("Sum of entered matrices:-\n");
   for (c = 0; c < m; c++) {
      for (d = 0 ; d < n; d++) {
         sum[c][d] = first[c][d] + second[c][d];
         printf("%d\t", sum[c][d]);
      }
      printf("\n");
   }
   return 0;
}
```

15. Pointer: Class lecture and book examples

16. Structure:

Source Code to Store Information of 10 students Using Structure

```
#include <stdio.h>
struct student{
    char name[50];
    int roll;
    float marks;
};
int main(){
    struct student s[10];
    int i;
    printf("Enter information of students:\n");
    for(i=0;i<10;++i)
    {
        s[i].roll=i+1;
        printf("\nFor roll number %d\n",s[i].roll);
        printf("Enter name: ");
        scanf("%s",s[i].name);
        printf("Enter marks: ");
        scanf("%f",&s[i].marks);
        printf("\n");
    }
    printf("Displaying information of students:\n\n");
    for(i=0;i<10;++i)
    {
     printf("\nInformation for roll number %d:\n",i+1);
     printf("Name: ");
     puts(s[i].name);
     printf("Marks: %.1f",s[i].marks);
   }
   return 0;
}
```

C Program to Calculate Difference Between Two Time Period

```
#include <stdio.h>
struct TIME{
  int seconds;
  int minutes;
  int hours;
};
void Difference(struct TIME t1, struct TIME t2, struct TIME *diff);
int main(){
    struct TIME t1,t2,diff;
    printf("Enter start time: \n");
    printf("Enter hours, minutes and seconds respectively: ");
    scanf("%d%d%d",&t1.hours,&t1.minutes,&t1.seconds);
    printf("Enter stop time: \n");
    printf("Enter hours, minutes and seconds respectively: ");
    scanf("%d%d%d",&t2.hours,&t2.minutes,&t2.seconds);
    Difference(t1,t2,&diff);
    printf("\nTIME DIFFERENCE: %d:%d:%d - ",t1.hours,t1.minutes,t1.seconds);
    printf("%d:%d:%d ",t2.hours,t2.minutes,t2.seconds);
    printf("= %d:%d:%d\n",diff.hours,diff.minutes,diff.seconds);
    return 0;
}
void Difference(struct TIME t1, struct TIME t2, struct TIME *differ){
    if(t2.seconds>t1.seconds){
        --t1.minutes;
        t1.seconds+=60;
    }
    differ->seconds=t1.seconds-t2.seconds;
    if(t2.minutes>t1.minutes){
        --t1.hours;
        t1.minutes+=60;
    }
    differ->minutes=t1.minutes-t2.minutes;
    differ->hours=t1.hours-t2.hours;
}
```

17.   Sorting:

```
void sort(int saifur[])
{
        int i,j,t;
        char t2;
for(i=0;i<*s;i++)
{
        for(j=*s-1;j>0;j--)
        {
                if(a[j]<a[j-1])
                {
                        t=a[j];
                        t2=t1[j];
                        a[j]=a[j-1];
                        t1[j]=t1[j-1];
                        a[j-1]=t;
                        t1[j-1]=t2;
}}   }
}

/* Bubble sort code */

#include <stdio.h>

int main()
{
  int array[100], n, c, d, swap;

  printf("Enter number of elements\n");
  scanf("%d", &n);

  printf("Enter %d integers\n", n);

  for (c = 0; c < n; c++)
    scanf("%d", &array[c]);

  for (c = 0 ; c < ( n - 1 ); c++) {
    for (d = 0 ; d < n - c - 1; d++) {
      if (array[d] > array[d+1]) /* For decreasing order use < */
      {
        swap       = array[d];
        array[d]   = array[d+1];
        array[d+1] = swap;
      }
```

```c
    }
  }
 printf("Sorted list in ascending order:\n");
  for ( c = 0 ; c < n ; c++ )
     printf("%d\n", array[c]);
  return 0;
}

void bubble_sort(long list[], long n)
{
  long c, d, t;
  for (c = 0 ; c < ( n - 1 ); c++){
    for (d = 0 ; d < n - c - 1; d++){
      if (list[d] > list[d+1])
      {
        /* Swapping */
        t          = list[d];
        list[d]    = list[d+1];
        list[d+1] = t;
      }
    }
  }
}
```

18.   Left Shifting and Right Shifting: Class lecture and book examples

19.   String Function All string function:

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

   int main()
   {
      char a[100];
      char b[100];
      int length;

      printf("Enter a string to calculate it's length\n");
      gets(a);
      gets(b);

   length = strlen(a);
   if (strcmp(a,b) == 0)
   printf("Entered strings are equal.\n");
   else
   printf("Entered strings are not equal.\n");

      printf("Length of entered string is = %d\n",length);

   //String Copy

   char source[1000], destination[1000];

      printf("Input a string\n");
      gets(source);

      strcpy(destination, source);

      printf("Source string:      \"%s\"\n", source);
      printf("Destination string: \"%s\"\n", destination);
   strcat(a,b);
      printf("String obtained on concatenation is %s\n",a);
   char string[1000];
      printf("Input a string to convert to upper case\n");
```

```
    gets(string);
    printf("Input string in upper case: \"%s\"\n",strupr(string));
    return 0;
}

int compare_strings(char a[], char b[])
{
int c = 0;
while (a[c] == b[c]) {
        if (a[c] == '\0' || b[c] == '\0')
                break;
        c++;
}

if (a[c] == '\0' && b[c] == '\0')
        return 0;
else
        return -1;
}

void concatenate(char p[], char q[]) {
int c, d;
c = 0;
while (p[c] != '\0') {
        c++;
}
d = 0;
while (q[d] != '\0') {
        p[c] = q[d];
        d++;
        c++;
}
p[c] = '\0';
}

void find_frequency(char s[], int count[]) {
int c = 0;
while (s[c] != '\0') {
        if (s[c] >= 'a' && s[c] <= 'z' )
                count[s[c]-'a']++;
        c++;
}
}

void upper_string(char s[]) {
int c = 0;
while (s[c] != '\0') {
        if (s[c] >= 'a' && s[c] <= 'z') {
                s[c] = s[c] - 32;
        }
        c++;
}
}

void lower_string(char s[]) {
int c = 0;
while (s[c] != '\0') {
        if (s[c] >= 'A' && s[c] <= 'Z') {
                s[c] = s[c] + 32;
        }
        c++;
}
}
```

C program to change case from upper to lower and lower to upper

```
#include <stdio.h>

int main ()
{
int c = 0;
char ch, s[1000];
printf("Input a string\n");
gets(s);
while (s[c] != '\0') {
       ch = s[c];
       if (ch >= 'A' && ch <= 'Z')
               s[c] = s[c] + 32;
       else if (ch >= 'a' && ch <= 'z')
               s[c] = s[c] - 32;
       c++;
}
printf("%s\n", s);
return 0;
}
```

20.   Graphic:

This c graphics program draws basic shapes such as circle, line, rectangle, ellipse and display text on screen using c graphics. This can be a first graphics program for a beginner-

```
#include<graphics.h>
#include<conio.h>
int main()
{
   int gd=DETECT,gm,left=100,top=100,right=200,
       bottom=200,x=300,y=150,radius=50;

   initgraph(&gd, &gm, "C:\\TC\\BGI");

   rectangle(left, top, right, bottom);
   circle(x, y, radius);
   bar(left + 300, top, right + 300, bottom);
   line(left - 10, top + 150, left + 410, top + 150);
   ellipse(x, y + 200, 0, 360, 100, 50);
   outtextxy(left + 100, top + 325, "My First C Graphics Program");

   getch();
   closegraph();
   return 0;
}
```

```
//C graphics program moving car

#include <graphics.h>
#include <dos.h>
int main()
{
   int i, j = 0, gd = DETECT, gm;
   initgraph(&gd,&gm,"C:\\TC\\BGI");
   settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
   outtextxy(25,240,"Press any key to view the moving car");
   getch();
   for( i = 0 ; i <= 420 ; i = i + 10, j++ )
   {
      rectangle(50+i,275,150+i,400);
      rectangle(150+i,350,200+i,400);
      circle(75+i,410,10);
      circle(175+i,410,10);
      setcolor(j);
      delay(100);
      if( i == 420 )
         break;
      if ( j == 15 )
         j = 2;

      cleardevice(); // clear screen
   }
   getch();
   closegraph();
   return 0;
}
```

21. Allocate memory dynamically using malloc() function.

```
#include <stdio.h>
#include <stdlib.h>
int main(){
    int n,i,*ptr,sum=0;
    printf("Enter number of elements: ");
    scanf("%d",&n);
    ptr=(int*)malloc(n*sizeof(int));  //memory allocated using malloc
    //ptr=(int*)calloc(n,sizeof(int));
    // ptr=realloc(ptr,n2);
   //free(ptr);
    if(ptr==NULL)
    {
        printf("Error! memory not allocated.");
        exit(0);
    }
    printf("Enter elements of array: ");
    for(i=0;i<n;++i)
    {
        scanf("%d",ptr+i);
        sum+=*(ptr+i);
    }
    printf("Sum=%d",sum);
    free(ptr);
   return 0;
}
```

Saifur Rahaman
Lecturer
Dept. of CSE
IIUC