Unit #: 1

Unit Name: Problem Solving Fundamentals

Topic: Problems on basic constructs in C and control structures

Course objectives:

The objective(s) of this course is to make students

CObj1: Acquire knowledge on how to solve relevant and logical problems using computing machine

CObj2: Map algorithmic solutions to relevant features of C programming language constructs

CObj3: Gain knowledge about C constructs and it's associated eco-system

CObj4: Appreciate and gain knowledge about the issues with C Standards and it's respective behaviors

CObj5: Get insights about testing and debugging C Programs

Course outcomes:

At the end of the course, the student will be able to

CO1: Understand and apply algorithmic solutions to counting problems using appropriate C Constructs

CO2: Understand, analyse and apply text processing and string manipulation methods using C Arrays, Pointers and functions

CO3: Understand prioritized scheduling and implement the same using C structures

CO4: Understand and apply sorting techniques using advanced C contructs

CO5: Understand and evaluate portable programming techniques using preprocessor directives and conditional compilation of C Programs

Team – PSWC,

Jan - May, 2022

Dept. of CSE,

PES University

List of problems

1. You are transporting some boxes through a tunnel whose height is only 41 feet. Given the length, width, and height of the box, calculate the volume of those boxes and check if they pass through the tunnel.

   Ex: Input = 5, 5, 5 Output = 125

   Input = 1, 2, 40 Output = 80

   Input = 10, 5, 41 Output = Can't Pass!

2. Given the number of rows, print a hollow diamond using star symbol: Ex: Input = 5

3. Check whether the given number is divisible by the sum of its digits. Display appropriate message
4. Write a C program to find the eligibility of admission for a professional course based on the following criteria: Eligibility Criteria : Marks in Maths >=65 and Marks in Physics >=50 and Marks in Chemistry>=55 and Total in all three subjects >=190 or Total in Maths and Physics >=140
5. Write a C program to print the prime numbers within the given input range. Ex. if user gives 10 and 50 as the input range, then the program must display all the prime numbers between the range 10 and 50 (i.e. 11 13 17 19 23 29 31 37 41 43 47)
6. Construct a menu-based calculator to perform arithmetic operations (Add, Subtract, Multiply,Divide) on complex numbers.
7. Given a number N, check whether it is a palindrome numbers or not. Ex. 121 , 3443 are palindromes whereas 123,4531 are not plaindromes
8. Write a program to print all odd numbers between the two limits "m" and "n" (both m and n is given as an integer inputs by the user and m<n) and print all odd numbers excluding those which are divisible by 3 and 5.

9. Implement a converter program to obtain an integer from a hexadecimal byte.
10. Given a number N, check whether the nth bit of a number, N is set to 1 or not. Input the N value from the user.

A few solutions

1. You are transporting some boxes through a tunnel whose height is only 41 feet. Given the length, width, and height of the box, calculate the volume of those boxes and check if they pass through the tunnel.

   Ex: Input = 5, 5, 5 Output = 125

   Input = 1, 2, 40 Output = 80

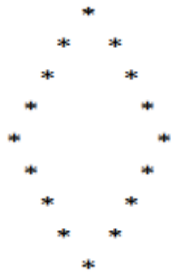
   Input = 10, 5, 41 Output = Can't Pass!

   Solution:

```
#include<stdio.h>
int main()
{
   int h, w, l;
   printf("Enter the length, width, height: ");
   scanf("%d %d %d", &l, &w, &h);
   if(h >= 41)
   {
      printf("Can't pass!");

   }
   else
   {
            int volume = l*w*h;
            printf("%d", volume);
   }
   return 0;
}
```

2. Given the number of rows, print a hollow diamond using star symbol: Ex: Input = 5



Solution:

```
#include<stdio.h>
int main()
{
  int n;
  printf("Enter number of rows: ");
  scanf("%d",&n);
  for(int i=1; i<= n; i++)
  {
    for(int j=i; j<= n; j++)
      printf(" ");
     for(int k = 1; k <= 2*i-1; k++)
    {
      if(k == 1 || k == (2*i-1))
          printf("*");
      else
          printf(" ");
    }
    printf("\n");
  }
  for(int i = n-1; i>=1; i--)
  {
```

```
    for(int j = n; j>=i; j--)
      printf(" ");

    for(int k = 1; k <= 2*i-1; k++)
    {
      if(k == 1 || k == 2*i-1)
          printf("*");
      else
          printf(" ");
    }
    printf("\n");
  }
  return 0;
}
```

3. Check whether the given number is divisible by the sum of its digits. Display appropriate message (Divisible or Not Divisible)

Solution:

```
#include<stdio.h>
 int main()
{
   int n;
   printf("Enter the number:");
   scanf("%d", &n);
   int temp = n;
   //finding the sum of digits of the number
   if(n<=0)
      printf("invalid");
   else
   {
   int sum = 0;
   while(n)
```

```
{
    int r = n % 10;
    sum = sum + r;
    n = n / 10;
}
//check if sum of digits divides the number
if(temp%sum == 0)
    printf("Divisible");
else
    printf("Not Divisible");
}
return 0;
}
```

4. Write a C program to find the eligibility of admission for a professional course based on the following criteria: Eligibility Criteria : Marks in Maths >=65 and Marks in Physics >=50 and Marks in Chemistry>=55 and Total in all three subjects >=190 or Total in Maths and Physics >=140

Solution:

```
#include <stdio.h>
int main()
{
   int p,c,m,total,math_phy;
  printf("Input the marks obtained in Physics, Chemistry, Math :");
  scanf("%d %d %d",&p, &c, &m);
  total = m + p + c;
  math_phy = m + p;
  printf("Total marks of Maths, Physics and Chemistry : %d\n",m+p+c);
  printf("Total marks of Maths and  Physics : %d\n",m+p);
  if (m>=65)
       if(p>=50)
          if(c>=55)
```

```
            if((total)>=190||(math_phy)>=140)
               printf("The  candidate is eligible for admission.\n");
            else
              printf("The candidate is not eligible.\n");
             else
          printf("The candidate is not eligible.\n");
           else
         printf("The candidate is not eligible.\n");
       else
        printf("The candidate is not eligible.\n");
        return 0;
    }
```

5. Write a C program to print the prime numbers within the given input range. Ex. if user gives 10 and 50 as the input range, then the program must display all the prime numbers between the range 10 and 50 (i.e. 11 13 17 19 23 29 31 37 41 43 47)

```
#include <stdio.h>
int main()
{
   int low, high, i, flag;
   printf("Enter two numbers(intervals): ");
   scanf("%d %d", &low, &high);

   printf("Prime numbers between %d and %d are: ", low, high);

   while (low < high)
   {
      flag = 0;

      for(i = 2; i <= low/2; ++i)
      {
         if(low % i == 0)
         {
            flag = 1;
            break;
         }
      }

      if (flag == 0)
         printf("%d ", low);
```

```
    ++low;
  }

  return 0;
}
```

6. Construct a menu-based calculator to perform arithmetic operations (Add, Subtract, Multiply,Divide) on complex numbers.

   Solution:

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
 int choice, a, b, c, d;
 do
 {
  printf("1. Addition\n");
  printf("2. Subtraction\n");
  printf("3. Multiplication\n");
  printf("4. Divide\n");
  printf("Enter your choice\n");
  scanf("%d", &choice);
  if (choice > 4 || choice < 1)
    exit(0);          // terminates the program
  printf("Enter a and b where a + ib is the first complex number.");
  scanf("%d %d", &a, &b);
  printf("Enter c and d where c + id is the second complex number.");
  scanf("%d %d", &c, &d);
  /*
  if (choice == 1)
  {
   int real = a+c;
```

```
  int img = b+d;
  if (img >= 0)
    printf("Sum = %d + %di\n", real, img);
  else
    printf("Sum = %d%di\n", real, img);
}
else if (choice == 2)
{
  int real = a-c;
  int img = b-d;

  if (img >= 0)
    printf("Difference = %d + %di", real, img);
  else
    printf("Difference = %d %di", real, img);
}
else if (choice == 3)
{
  int real = a*c - b*d;
  int img = b*c + a*d;

  if (img >= 0)
    printf("Product = %d + %di", real, img);
  else
    printf("Product = %d %di", real, img);
}
else if (choice == 4)
{
  if (c == 0 && d == 0)
    printf("Division by 0 + 0i isn't allowed.");
  else
```

```
{
 int x = a*c + b*d;
 int y = b*c - a*d;
 int z = c*c + d*d;

 if (x%z == 0 && y%z == 0)
 {
  if (y/z >= 0)
    printf("Division of the complex numbers = %d + %di", x/z, y/z);
  else
    printf("Division of the complex numbers = %d %di", x/z, y/z);
 }
 else if (x%z == 0 && y%z != 0)
 {
  if (y/z >= 0)
    printf("Division of two complex numbers = %d + %d/%di", x/z, y, z);
  else
    printf("Division of two complex numbers = %d %d/%di", x/z, y, z);
 }
 else if (x%z != 0 && y%z == 0)
 {
  if (y/z >= 0)
    printf("Division  = %d/%d + %di", x, z, y/z);
  else
    printf("Division = %d %d/%di", x, z, y/z);
 }
 else
 {
  if (y/z >= 0)
    printf("Division = %d/%d + %d/%di",x, z, y, z);
  else
```

```
        printf("Division = %d/%d %d/%di", x, z, y, z);
    }
  }
}*/
int real, img;
switch(choice)
{
case 1:
  real = a+c;
  img = b+d;
  if (img >= 0)
    printf("Sum = %d + %di\n", real, img);
  else
    printf("Sum = %d%di\n", real, img);
  break;

case 2:
  real = a-c;
  img = b-d;
  if (img >= 0)
    printf("Difference = %d + %di\n", real, img);
  else
    printf("Difference = %d %di\n", real, img);
  break;

case 3:
  real = a*c - b*d;
  img = b*c + a*d;
  if (img >= 0)
    printf("Product = %d + %di\n", real, img);
  else
```

```
    printf("Product = %d %di\n", real, img);
  break;
case 4:
  if (c == 0 && d == 0)
    printf("Division by 0 + 0i isn't allowed.");
  else
  {
    int x = a*c + b*d;
    int y = b*c - a*d;
    int z = c*c + d*d;
    if (x%z == 0 && y%z == 0)
    {
      if (y/z >= 0)
        printf("Division of the complex numbers = %d + %di", x/z, y/z);
      else
        printf("Division of the complex numbers = %d %di", x/z, y/z);
    }
    else if (x%z == 0 && y%z != 0)
    {
      if (y/z >= 0)
        printf("Division of two complex numbers = %d + %d/%di", x/z, y, z);
      else
        printf("Division of two complex numbers = %d %d/%di", x/z, y, z);
    }
    else if (x%z != 0 && y%z == 0)
    {
      if (y/z >= 0)
        printf("Division  = %d/%d + %di", x, z, y/z);
      else
        printf("Division = %d %d/%di", x, z, y/z);
    }
```

```
        else
        {
          if (y/z >= 0)
            printf("Division = %d/%d + %d/%di",x, z, y, z);
          else
            printf("Division = %d/%d %d/%di", x, z, y, z);
        }
       }
       break; Breaks come out of current loop
      }
  }while(choice<=4);
  return 0;
}
```

HAPPY CODING!

Solve other problems. If any issues, please contact sindhurpai@pes.edu