

Q.1 Write a program using conditional operator (?:) to determine whether a year entered through the keyboard is **leap year** or not.

```

1 #include <stdio.h>
2 int main() {
3     int year;
4     printf("Enter a year: ");
5     scanf("%d", &year);
6     (year%4==0 && year%100!=0) ? printf("%d is a leap year.", year) :
7     (year%400 ==0 ) ? printf("%d is a leap year.", year) : printf("%d is not a leap year.", year);
8     return 0;
9 }

```

input

Enter a year: 2021
2021 is not a leap year.

Q.2 An insurance company follows the following rules to calculate the premium to be paid by the client/customer:

- If a person's health is excellent and the person is between 18 and 30 years of age, lives in a city, and is a male then the premium is Rs.50 per 10,000 and his policy amount cannot exceed Rs. 3 lakhs INR.
- If a person satisfies all the above conditions except that the sex is female, then the premium is Rs 30 per 10,000 and her policy amount cannot exceed Rs 2 lakhs INR.
- If a person's health is poor and the person is between 18 and 30 years of age and lives in a village and is a male, then the premium is Rs. 70 per 10,000 and his policy cannot exceed Rs. 1 lakh.

Write a program to output whether the person should be insured or not, his/ her premium rate and maximum amount for which he/ she can be insured. Usage of if—else ladder.

```

1 #include<stdio.h>
2 int main(){
3     int age, premium, max_amount;
4     char health, location, sex;
5     scanf ("%c %d %c %c", &health, &age, &location, &sex);
6     if ((health=='e') && ((age>=18)&&(age<=30)) && (location=='c') && (sex=='m'))
7     {
8         premium=50;
9         max_amount=3;
10        printf("The payable premium is Rs.%d", premium);
11        printf("\tThe maximum policy amount is Rs. %d Lakhs INR", max_amount);
12    }
13    else if ((health=='e') && ((age>=18)&&(age<=30)) && (location=='c') && (sex=='f'))
14    {
15        premium=30;
16        max_amount=2;
17        printf("The payable premium is Rs.%d", premium);
18        printf("\t The maximum policy amount is Rs. %d Lakhs INR", max_amount);
19    }
20    else if ((health=='p') && ((age>=18)&&(age<=30)) && (location=='v') && (sex=='m'))
21    {
22        premium=70;
23        max_amount=1;
24        printf("The payable premium is Rs.%d", premium);
25        printf("\t The maximum policy amount is Rs. %d Lakhs INR", max_amount);
26    }
27    else
28    {
29        printf("This person is not insured.");
30    }
31 }

```

input

e 23 c m
The payable premium is Rs.50 The maximum policy amount is Rs. 3 Lakhs INR

p 23 v m
The payable premium is Rs70 The maximum policy amount is Rs. 1 Lakhs INR

Q.3 Write a program that tests a user entered character and displays its classification according to the ASCII classifications as shown in Fig.1. Write the program starting at the top of the classification tree and display all classifications that the character belongs to. For example, if the user enters a digit, you should display that it is printable, graphical, alphanumeric, and a digit as shown in the output for digit 2. Use Switch statement in c to write this program.

```

1 #include<stdio.h>
2 int main()
3 {
4     char ch;
5     printf("Enter a character\n");
6     scanf("%c", &ch);
7     switch (ch < 32)
8     {
9         case 1: //ascii for control sequences is between 0 and 31
10            printf("control\n");
11            break;
12         case 0:
13            printf("printable\n");
14            switch(ch)
15            {
16                case 32: //ascii for spacebar is 32
17                    printf("space\n");
18                    break;
19                default:
20                    printf("graphical\n");
21                    switch(((ch >= 65 && ch <= 90) || (ch >= 97 && ch <= 122) || (ch >= 48 && ch <= 57)))
22                    {
23                        case 1:
24                            printf("alphanumeric\n");
25                            switch(ch >= 48 && ch <= 57)
26                            {
27                                case 1://ascii for digits is between 48 and 57
28                                    printf("digit\n");
29                                    break;
30                                case 0:
31                                    printf("alphabetic\n");
32                                    switch (ch >= 65 && ch <= 90)
33                                    {
34                                        case 0://ascii for Lower case is between 97 and 122
35                                            printf("lower case\n");
36                                            break;
37                                        case 1://ascii for upper case is between 65 and 90
38                                            printf("upper case\n");
39                                            break;
40                                    }
41                                    break;
42                                }
43                            break;
44                        case 0:
45                            switch(((ch >= 33 && ch <= 47) || (ch >= 58 && ch <= 64) || (ch >= 91 && ch <= 96) || (ch >= 123 && ch <= 126)))
46                            {
47                                case 1:
48                                    printf("punctuation\n");
49                                    break;
50                                case 0:
51                                    printf("Unknown character printed");
52                                    break;
53                                }
54                            break; }
55                    }
56            break;}
57     return 0;
58 }

```

```

Enter a character
B
printable
graphical
alphanumeric
alphabetic
upper case

```

```

Enter a character
2
printable
graphical
alphanumeric
digit

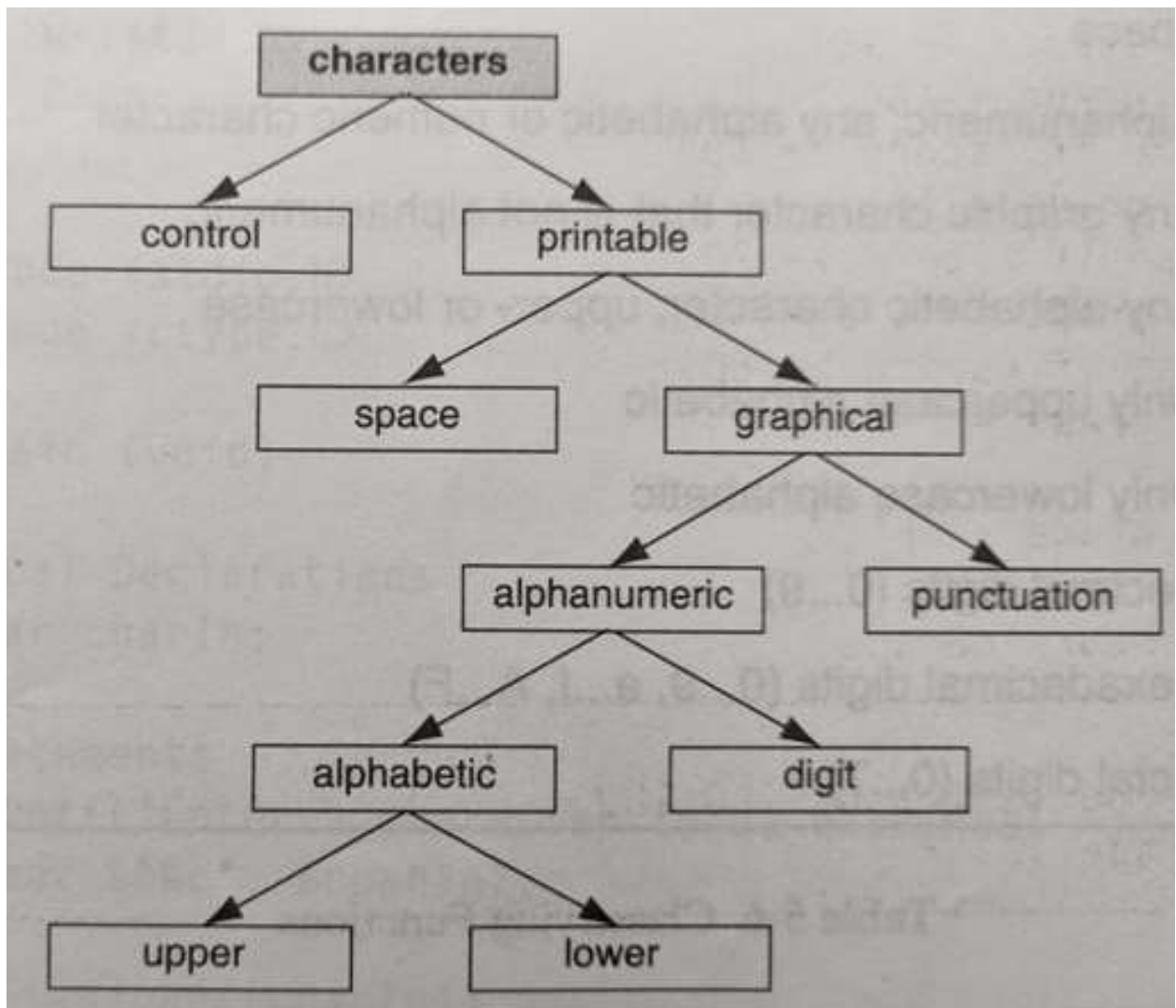
```

```

Enter a character

printable
space

```



(Figure 1)

Q.4 Adding a list of numbers from standard input as shown in the output window. Example of **sentinel** controlled loop.

```
1  #include <stdio.h>
2  int main ( )
3  {
4      int x;
5      int sum = 0;
6
7      printf ("Enter your numbers: <EOF> to stop\n");
8      while (scanf ("%d", &x) != EOF)
9          sum += x;
10     printf("The total is: %d\n", sum);
11     return 0;
12 }
```

Enter your numbers: <EOF> to stop
21 65 3 2 -6
The total is: 85

Q.5 Using if and while, print 10 numbers per line. Input given is a number between 1 and 50.

```
1  #include <stdio.h>
2  int main ( )
3  {
4      int num;
5      int lineCount;
6      printf("Enter an integer between 1 and 50:");
7      scanf("%d",&num);
8      if (num > 50)
9          num = 50;
10     lineCount = 0;
11     while (num > 0)
12     {
13         if (lineCount < 10)
14             lineCount++;
15         else
16         {
17             printf("\n");
18             lineCount = 1;
19         }
20         printf ("%3d", num--);
21     }
22     return 0;
23 }
```

```
Enter an integer between 1 and 50:47
 47 46 45 44 43 42 41 40 39 38
 37 36 35 34 33 32 31 30 29 28
 27 26 25 24 23 22 21 20 19 18
 17 16 15 14 13 12 11 10  9  8
  7  6  5  4  3  2  1
```

Tasks: (Do any two from the below tasks)

- (1) Write an equivalent program for Q.1 (checking leap year or not) using **if-else** statements.
- (2) Rewrite the following code fragment using one switch statement. Add any other needed declarations or assignments. Run the code and submit the screenshot of the run.

```
if (ch == 'A' || 'a')
    countA++;
else if (ch == 'C' || 'c')
    countC++;
else printf ("error...");
```

- (3) Print only the output up to one more than the half of the number given as input (in Q.5). Your modified code should give the below output: (Sample output):

```
Enter an integer between 1 and 50:34
 34 33 32 31 30 29 28 27 26 25
 24 23 22 21 20 19 18
```

```
Enter an integer between 1 and 50:13
 13 12 11 10  9  8  7
```

Q.1 Generate Pascal's triangle of any length. Usage of Conditionals and loops for generating the triangle.

```
1  #include<stdio.h>
2  int main(){
3      int bin,p,q,r,x;
4      bin=1;
5      q=0;
6      printf("Rows you want to input:");
7      scanf("%d",&r);
8      printf("\nPascal's Triangle:\n");
9
10     while(q<r) {
11         for(p=40-3*q; p>0; --p)
12             printf(" ");
13         for(x=0;x<=q;++x)
14         {
15             if((x==0) || (q==0))
16                 bin=1;
17             else
18                 bin=(bin*(q-x+1))/x;
19             printf("%6d",bin);
20         }
21         printf("\n");
22         ++q;
23     }
24 }
```

Rows you want to input:6

Pascal's Triangle:

```

                                     1
                                1   1
                           1   2   1
                       1   3   3   1
                   1   4   6   4   1
               1   5   10  10   5   1
```


Rows you want to input:3

Pascal's Triangle:

```
      1
     1 1
    1 2 1
```

Q.2 Write a program using nested for loops that generate a pattern of numbers and #s as shown in the output.

```
1  #include <stdio.h>
2  int main()
3  {
4      int n, row, col;
5      printf("enter a number between 1 and 9:");
6      scanf("%d", &n);
7      for (row = 1; row <= n; row++)
8      {
9          for (col = 1; col <= n; col++)
10             if (row >= col)
11                 printf ("%d", col);
12             else
13                 printf ("#");
14             printf ("\n");
15         }
16         return 0;
17     }
```

enter a number between 1 and 9:6

```
1#####
12####
123###
1234##
12345#
123456
```

Tasks:

a) Modify the above program to generate the pattern shown in first figure below. b) Modify the above code to generate the pattern shown in second figure.

```
enter a number between 1 and 9:6
123456
#23456
##3456
###456
####56
#####6
```

Pattern for task 2 (a)

Pattern
for task
2 (b) ➡

```
1#####
12#####
123#####
1234#####
12345#####
```

Q.3 Write a program using nested for loops that generate a pattern as shown in the output of the below program.

```

1  #include <stdio.h>
2  int main()
3  {
4      int x = 0,y = 0;
5      unsigned int rows = 0;
6      printf("Enter the number of rows = ");
7      scanf("%u",&rows);
8      for(x=1; x<=rows; ++x)
9      {
10         // Print spaces
11         for(y=x; y<=rows; ++y)
12         {
13             printf(" ");
14         }
15         // Print star/
16         for(y =1; y<=((2*x)-1); ++y)
17         {
18             printf("*");
19         }
20         // Print new line
21         for(y=x; y<=2*rows-x; ++y)
22         {
23             printf(" ");
24         }
25         // Print star/
26         for(y =1; y<=((2*x)-1); ++y)
27         {
28             printf("*");
29         }
30         printf("\n");
31     }
32     return 0;
33 }

```



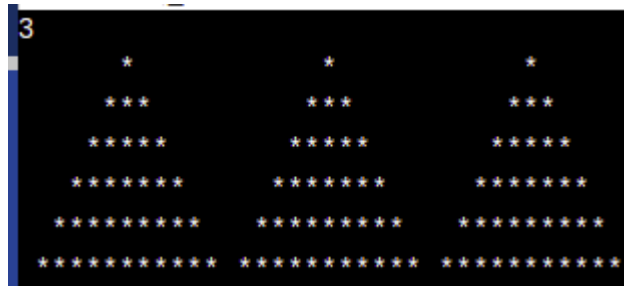
Enter the number of rows = 6

```

      *              *
    ***            ***
  *****        *****
*****          *****
*****          *****
*****          *****
*****          *****

```


Task 3 Modify the program in Question 3 to print 3 pyramids and the number of rows should be twice the input number as shown below.



Q.4 Write a program for a matchstick game being played between the computer and a user. Your program should ensure that the computer always wins. Rules for the game are as follows:

- There are 21 matchsticks,
- The computer asks the user to pick 1, 2, 3, or 4 matchsticks,
- After user picks, the computer does its picking. And this pattern repeats...
- Whoever is forced to pick up the last matchstick loses the game.

A sample run is as below:

```
Total Match Sticks remaining: 21
Pick up the match sticks between (1 to 4): 3
Computer picks up the 2 match sticks.
Total Match Sticks remaining: 16
Pick up the match sticks between (1 to 4): 2
Computer picks up the 3 match sticks.
Total Match Sticks remaining: 11
Pick up the match sticks between (1 to 4): 4
Computer picks up the 1 match sticks.
Total Match Sticks remaining: 6
Pick up the match sticks between (1 to 4): 3
Computer picks up the 2 match sticks.

You lost and computer won.
```

Program:

```
#include<stdio.h>
int main() {
    int match_sticks = 21, user_choice, computer_choice;
    while(match_sticks>=1)
    {
        printf("Total Match Sticks remaining: %d\n", match_sticks);
        printf("Pick up the match sticks between (1 to 4): "); scanf("%d", &user_choice);
        if(user_choice > 4) {
            printf("Invalid Entry: Game ends..."); break;
        }
        computer_choice = 5 - user_choice;
        printf("Computer picks up the %d match sticks.\n", computer_choice);
        match_sticks = match_sticks-user_choice-computer_choice;
        if(match_sticks==1) {
            printf("\nYou lost and computer won."); break;
        }
    }
    return(0);
}
```

Birla Institute of Technology & Science, Pilani Hyderabad Campus

Second Semester 2020-2021

Computer Programming [CS F111]

Lab 8

Q1. Write a program to convert a Binary number into its equivalent Decimal number. Do not use Arrays or any other data structures.

```
1  #include <stdio.h>
2  #include <math.h>
3
4  int convert (long long n);
5  int main ()
6  {
7      long long n;
8      printf ("Enter a binary number: ");
9      scanf ("%lld", &n);
10     printf ("%lld in binary = %d in decimal", n, convert (n));
11     return 0;
12 }
13
14 int convert (long long n)
15 {
16     int dec = 0, i = 0, rem;
17     while (n != 0) {
18         rem = n % 10;
19         n /= 10;
20         dec += rem * pow (2, i);
21         ++i;
22     }
23     return dec;
24 }
```

Output:

```
Enter a binary number: 11001
11001 in binary = 25 in decimal
```

Q2. Write a program to convert a DECIMAL number into its equivalent OCTAL number. Do not use Arrays or any other data structures.

```
1  #include <stdio.h>
2  #include <math.h>
3  long long convert(int n);
4  int main() {
5      int n;
6      printf("Enter a decimal number: ");
7      scanf("%d", &n);
8      printf("%d in decimal = %lld in octal", n, convert(n));
9      return 0;
10 }
11
12 long long convert(int n) {
13     long long bin = 0;
14     int rem, i = 1, step = 1;
15     while (n != 0) {
16         rem = n % 8;
17         n /= 8;
18         bin += rem * i;
19         i *= 10;
20     }
21     return bin;
22 }
```

Output1:

```
Enter a decimal number: 12
12 in decimal = 14 in octal
```

Output2:

```
Enter a decimal number: 337
337 in decimal = 521 in octal

...Program finished with exit code 0
Press ENTER to exit console.
```

Q3. Write a program to find out 1's complement of a Decimal integer. Do not use Arrays or any other data structures.

```
1  #include <stdio.h>
2  long long convert(long long int n) {
3      long long bin = 0;
4      long long int rem, i = 1, step = 1;
5      while (n != 0) {
6          rem = n % 2;
7          n /= 2;
8          bin += rem * i;
9          i *= 10;
10     }
11     return bin;
12 }
13 int main() {
14     long long int num1, num2 = 5, no, counter;
15     long long fab = 1, count = 0;
16     int power = 1, rem;
17     printf("Enter a decimal integer:");
18     scanf("%lld", &no);
19     num1 = convert(no);
20     printf("Binary equivalent:%lld\n", num1);
21     while (num1 != 0) {
22         counter = num1 % 10;
23         if (counter == 0)
24             num2 = num2 * 10 + 1;
25         else
26             num2 *= 10;
27         num1 /= 10;
28     }
29     while (num2 % 10 != 5) {
30         rem = num2 % 10;
31         num1 = num1 * 10 + rem;
32         num2 /= 10;
33     }
34     printf("1's Complement with leading 0s' suppressed: %lld", num1);
35 }
```

Output:

```
Enter a decimal integer:23
Binary equivalent:10111
1's Complement with leading 0s' suppressed: 1000
```

(**Note:** As integers are used, the binary output gets shortened, because leading zeroes will have no significance for an int, e.g. 01000 will be output as 1000, which is acceptable).

Q4. Write a program to do binary addition. Take two binary numbers as input and add them to get their binary sum. Take your input in long long int format. Do not use Arrays or any other data structures.

```
1  #include <stdio.h>
2  #include <math.h>
3  long long convert(long long int n){
4      int dec = 0, i = 0, rem;
5      while (n != 0) {
6          rem = n % 10;
7          n /= 10;
8          dec += rem * pow(2, i);
9          ++i;
10     }
11     return dec;
12 }
13 long long convertB(long long int n) {
14     long long bin = 0;
15     long long int rem, i = 1, step = 1;
16     while (n != 0) {
17         rem = n % 2;
18         n /= 2;
19         bin += rem * i;
20         i *= 10;
21     }
22     return bin;
23 }
24 void main() {
25     long long int num1, num2 = 5, no, counter, fab = 1, count = 0;
26     printf("Enter both the numbers:\n");
27     scanf("%lld %lld", &num1, &num2);
28     num1 = convert(num1);
29     num2 = convert(num2);
30     no = num1 + num2;
31
32     printf("Sum: %lld", convertB(no));
33 }
34 }
```

Output:

```
Enter both the numbers:
1101
1101
Sum: 11010
```

TASKS:

1. Write a program to PRINT the hexadecimal equivalent of a input decimal number. Do not use any arrays.

2. Take 2 binary numbers as input and subtract the second number from the first one. For this find the 1's complement of the second number and add it to the first number. The steps are as follows. [Do not use Arrays or any other data structures]

i. Find the 1's complement of the second number.

ii. Add it to the first number.

iii. If the result of addition has a carry over then it is ignored and an 1 is added in the last bit of the result.

iv. If there is no carry over, then 1's complement of the result of addition is obtained to get the final result and it is negative.

Birla Institute of Technology & Science, Pilani Hyderabad Campus

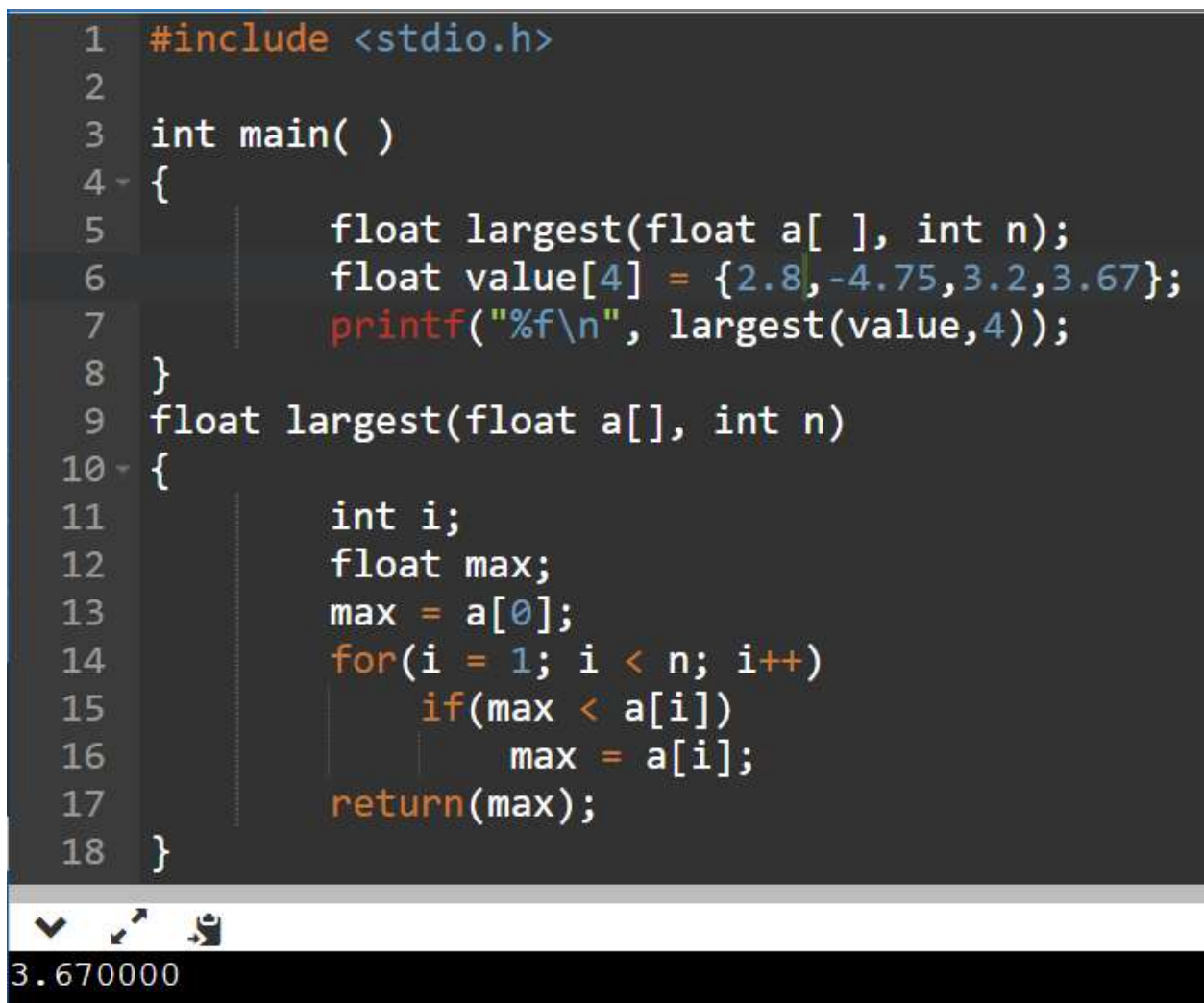
Second Semester 2020-2021

Computer Programming [CS F111]

Lab 9 (Arrays)

Q1. Write a program to find out largest number from a list of numbers stored in an Array.

```
1  #include <stdio.h>
2
3  int main( )
4  {
5      float largest(float a[ ], int n);
6      float value[4] = {2.8,-4.75,3.2,3.67};
7      printf("%f\n", largest(value,4));
8  }
9  float largest(float a[], int n)
10 {
11     int i;
12     float max;
13     max = a[0];
14     for(i = 1; i < n; i++)
15         if(max < a[i])
16             max = a[i];
17     return(max);
18 }
```



3.670000

Q2. Write a program to input a set of characters, words, lines into an Array terminated by ~. Find out how many words, lines and characters are there in the input.

```

1  #include<stdio.h>
2  int main()
3  {
4      char str[200];
5      int line, word, ch;
6
7      line = word = ch = 0;
8      printf("Enter string terminated with ~ :\n");
9      scanf("%[^\n]", str);
10     // check every character
11     for(int i=0; str[i]!='\0'; i++)
12     {
13         // if it is new line then one line and one word completed
14         if(str[i]=='\n')
15         {
16             line++;
17             word++;
18         }
19         // else it is a character
20         else
21         {
22             // if character is space or tab then one word is also completed
23             if(str[i]==' '||str[i]=='\t')
24             {
25                 word++;
26                 ch++;
27             }
28             // it was not '\n', space or tab
29             // it is a normal character
30             else {
31                 ch++;
32             }
33         }
34     }
35     printf("\nCharacter counts = %d\n", ch);
36     printf("Word counts = %d\n", word);
37     printf("Line counts = %d\n", line);
38     return 0;
39 }

```

```

Enter string terminated with ~ :
hello, how are you?
i am fine.
~

Character counts = 29
Word counts = 7
Line counts = 2

```

Task: Modify the above program to count the number of special characters additionally along with other output.

Q.3 Write a program to input a binary number from keyboard. Using Arrays, generate the 2's complement representation of this number.

```

1  #include <stdio.h>
2  int main() {
3      int n;
4      int carry = 1;
5      printf("Enter the number of bits:");
6      scanf("%d",&n);
7      char binary[n+1];
8      char onecomplement [n+1];
9      char twoscomplement [n+1];
10     printf("\nEnter the binary number : ");
11     scanf("%s", binary);
12     printf("\nThe 1's complement of the number is:");
13     for(int i=0;i<n;i++) {
14         if(binary[i]=='0')
15             onecomplement[i]='1';
16         else if(binary[i]=='1')
17             onecomplement[i]='0';
18     }
19     onecomplement[n]='\0';
20     printf("%s",onecomplement);
21     printf("\nThe 2's complement of the number is:");
22     for(int i=n-1; i>=0; i--) {
23         if(onecomplement[i] == '1' && carry == 1)
24         {
25             twoscomplement[i] = '0';
26         }
27         else if(onecomplement[i] == '0' && carry == 1)
28         {
29             twoscomplement[i] = '1';
30             carry = 0;
31         }
32         else
33         {
34             twoscomplement[i] = onecomplement[i];
35         }
36     }
37     twoscomplement[n]='\0';
38     printf("%s",twoscomplement);
39     return 0;
40 }

```

Output:

```
Enter the number of bits:4

Enter the binary number : 1100

The 1's complement of the number is:0011
The 2's complement of the number is:0100
```

Q.4 Assume that Admission division at BITS, Pilani wants you to write a C program to admit students into its 3 engineering streams (Civil, EEE and CSE) based on the following logic for the year 2021. Assume that there are 2 seats in each of these branches. Your program should receive as input student application number (an integer), his scores as per the below requirement and allot him/ her a branch according to his/ her preferences and as per the admission rules given below:

Admission logic:

1. A student should have a minimum aggregate of 75% marks in Phy, Chem and Maths; and 60% in each of the PCM subjects at 12th standard. He/ she should have passed 12th standard either in 2021 or in 2020.
2. He/ she should have scored a minimum of 240 marks in the current year's (2021) BITSAT.
3. Seats are allotted to students on the basis of their BITSAT scores while PCM scores in 12th are considered as mere eligibility, i.e. if a student with a valid BITSAT score wants first a seat in a particular stream, then he/ she is allotted the seat provided his MSN (merit serial number) is higher than others.
4. Receive 7 students BITSAT score and other scores (which are valid scores), and store these into Arrays. Generate MSNs for each and allot streams/ branches based on their BITSAT scores and preferences.

Program:

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

```

int main() {
    int n = 7;

    int bitsat[14], boards_phy[n], boards_chem[n], boards_math[n], appln[n];
    int passyr[n], result[n];
    int seats[] = {2, 2, 2};
    int preference[3 * n];

    // 0 means CS, 1 means Civil, 2 means EEE, 3 Not Allotted
    int eligibility[7] = {};

    for (int i = 0; i < 7; i++) {
        printf("Please Enter Student %d's application number: ", i + 1);
        scanf("%d", &appln[i]);

        printf("Please Enter Student %d's Physics score in boards: ", i + 1);
        scanf("%d", &boards_phy[i]);

        printf("Please Enter Student %d's Chemistry score in boards: ", i + 1);
        scanf("%d", &boards_chem[i]);

        printf("Please Enter Student %d's Maths score in boards: ", i + 1);
        scanf("%d", &boards_math[i]);

        printf("Please Enter Student %d's Board passing year: ", i + 1);
        scanf("%d", &passyr[i]);

        printf("Please Enter Student %d's BITSAT score: ", i + 1);
        scanf("%d", &bitsat[2 * i]);

        bitsat[2 * i + 1] = i; //2i+1 is index

        printf("Please Enter Student %d's Preference ", i + 1);
        for (int j = 3 * i; j < 3 * (i + 1); j++) {
            scanf("%d", &preference[j]);
        }
    }
}

```



```

        printf("\n");
    }
    for (int i = 0; i < 7; i++) {

        printf("\n");

        if (boards_phy[i]>=60 && boards_chem[i]>=60 && boards_math[i]>=60)
{

        if (boards_phy[i] + boards_chem[i] + boards_math[i] >= 225) {

            if (bitsat[2 * i] >= 240) {

                if (passyr[i] == 2021 || passyr[i] == 2020)

                    eligibility[i] = 1;

            }

        }

    }

}

for (int i = 0; i < 14; i += 2) {
    for (int j = i + 2; j < 14; j += 2) {
        if (bitsat[i] < bitsat[j]) {
            int tempMark = bitsat[i], tempIndex = bitsat[i + 1];
            bitsat[i] = bitsat[j];
            bitsat[i + 1] = bitsat[j + 1];
            bitsat[j] = tempMark;
            bitsat[j + 1] = tempIndex;
        }
    }
}

```

```

    }
}
for (int i = 0; i < 7; i++) {
    int index = bitsat[2 * i + 1];
    if (eligibility[index] == 0)
        result[index] = 3;
    else {
        int pref1 = preference[3 * index];
        int pref2 = preference[(3 * index) + 1];
        int pref3 = preference[(3 * index) + 2];
        if (seats[pref1] != 0) {
            result[index] = pref1;
            seats[pref1] = seats[pref1] - 1;
        }
        else if (seats[pref2] != 0) {
            result[index] = pref2;
            seats[pref2] = seats[pref2] - 1;
        }
        else if (seats[pref3] != 0) {
            result[index] = pref3;
            seats[pref3] = seats[pref3] - 1;
        }
        else
            result[index] = 3;
    }
}
printf("Allotment: \n");

```

```
for (int i = 0; i < 7; i++) {  
  
    printf("Student%d: Application no.: %d ", i + 1, appln[i]);  
  
    switch (result[i]) {  
  
        case 0:  
  
            printf("CS");  
  
            break;  
  
        case 1:  
  
            printf("Civil");  
  
            break;  
  
        case 2:  
  
            printf("EEE");  
  
            break;  
  
        case 3:  
  
            printf("-");  
  
            break;  
  
    }  
  
    printf("\n");  
  
    }  
}
```

Task:

Extend the above program to allot 15 students competing for 3 seats each in CSE, Civil and EEE departments.

Please Enter Student 1's application number: 1001
Please Enter Student 1's Physics score in boards: 89
Please Enter Student 1's Chemistry score in boards: 96
Please Enter Student 1's Maths score in boards: 79
Please Enter Student 1's Board passing year: 2021
Please Enter Student 1's BITSAT score: 357
Please Enter Student 1's Preference 0 1 2

Please Enter Student 2's application number: 1002
Please Enter Student 2's Physics score in boards: 86
Please Enter Student 2's Chemistry score in boards: 76
Please Enter Student 2's Maths score in boards: 90
Please Enter Student 2's Board passing year: 2020
Please Enter Student 2's BITSAT score: 378
Please Enter Student 2's Preference 0 1 2

Please Enter Student 3's application number: 1003
Please Enter Student 3's Physics score in boards: 89
Please Enter Student 3's Chemistry score in boards: 76
Please Enter Student 3's Maths score in boards: 68
Please Enter Student 3's Board passing year: 2021
Please Enter Student 3's BITSAT score: 320
Please Enter Student 3's Preference 0 2 1

Please Enter Student 4's application number: 1004
Please Enter Student 4's Physics score in boards: 88
Please Enter Student 4's Chemistry score in boards: 77
Please Enter Student 4's Maths score in boards: 99
Please Enter Student 4's Board passing year: 2021
Please Enter Student 4's BITSAT score: 335
Please Enter Student 4's Preference 0 1 2

Please Enter Student 5's application number: 1005
Please Enter Student 5's Physics score in boards: 88
Please Enter Student 5's Chemistry score in boards: 86
Please Enter Student 5's Maths score in boards: 97
Please Enter Student 5's Board passing year: 2021
Please Enter Student 5's BITSAT score: 376
Please Enter Student 5's Preference 0 2 1

Please Enter Student 6's application number: 1006
Please Enter Student 6's Physics score in boards: 89
Please Enter Student 6's Chemistry score in boards: 97
Please Enter Student 6's Maths score in boards: 88
Please Enter Student 6's Board passing year: 2020
Please Enter Student 6's BITSAT score: 320
Please Enter Student 6's Preference 1 2 0

Please Enter Student 7's application number: 1007
Please Enter Student 7's Physics score in boards: 88
Please Enter Student 7's Chemistry score in boards: 87
Please Enter Student 7's Maths score in boards: 86
Please Enter Student 7's Board passing year: 2020
Please Enter Student 7's BITSAT score: 356
Please Enter Student 7's Preference 0 2 1

Allotment:

Student1: Application no.: 1001 Civil
Student2: Application no.: 1002 CS
Student3: Application no.: 1003 -
Student4: Application no.: 1004 Civil
Student5: Application no.: 1005 CS
Student6: Application no.: 1006 EEE
Student7: Application no.: 1007 EEE

-----#-----