DAY 6

1)
```
#include<stdio.h>
int main()
{
   char a=1;
   while(a<=10);//halt the process//create logical errors
   {
      printf("a=%d",a);
      a++;
   }
   return 0;
}
```

2)//wap to display multiplication table from 1 to 10

```
#include<stdio.h>
int main()
{
  int i=1,j;
  while(i<=10){
    j=1;
    while(j<=10){
       printf("%d * %d=%d\t",i,j,i*j);
       j++;
    }
    printf("\n");
    i++;
  }
  return 0;
}
```

OUTPUT

```
[?2004l
1 * 1=1         1 * 2=2         1 * 3=3         1 * 4=4         1 * 5=5         1 * 6=6         1 *
7=7             1 * 8=8         1 * 9=9         1 * 10=10
2 * 1=2         2 * 2=4         2 * 3=6         2 * 4=8         2 * 5=10                2 * 6=12
                2 * 7=14                2 * 8=16                2 * 9=18                2 * 10=20

3 * 1=3         3 * 2=6         3 * 3=9         3 * 4=12                3 * 5=15                3 *
6=18            3 * 7=21                3 * 8=24                3 * 9=27                3 * 10=30
```

```
4 * 1=4         4 * 2=8         4 * 3=12                4 * 4=16                4 * 5=20
        4 * 6=24                4 * 7=28                4 * 8=32                4 * 9=36
        4 * 10=40
5 * 1=5         5 * 2=10                5 * 3=15                5 * 4=20                5 * 5=25
                5 * 6=30                5 * 7=35                5 * 8=40                5 * 9=45
                5 * 10=50
6 * 1=6         6 * 2=12                6 * 3=18                6 * 4=24                6 * 5=30
                6 * 6=36                6 * 7=42                6 * 8=48                6 * 9=54
                6 * 10=60
7 * 1=7         7 * 2=14                7 * 3=21                7 * 4=28                7 * 5=35
                7 * 6=42                7 * 7=49                7 * 8=56                7 * 9=63
                7 * 10=70
8 * 1=8         8 * 2=16                8 * 3=24                8 * 4=32                8 * 5=40
                8 * 6=48                8 * 7=56                8 * 8=64                8 * 9=72
                8 * 10=80
9 * 1=9         9 * 2=18                9 * 3=27                9 * 4=36                9 * 5=45
                9 * 6=54                9 * 7=63                9 * 8=72                9 * 9=81
                9 * 10=90
10 * 1=10               10 * 2=20               10 * 3=30               10 * 4=40               10
* 5=50          10 * 6=60               10 * 7=70               10 * 8=80               10 * 9=90
                10 * 10=100
[?2004h
```

3)//wap to display right angled star pattern

```
#include<stdio.h>
int main()
{
   int i=1,j;
   while(i<=5){
      j=1;
      while(j<=i){
          printf("*");
          j++;

      }
      printf("\n");
      i++;
   }
   return 0;
}
```

OUTPUT

```
*
**
***
****
*****
```

4)//wap to display right angled star pattern

```
#include<stdio.h>
int main()
{
   int i=1,j;
   while(i<=5){
      j=5;
      while(j>=i){
          printf("*");
          j--;

      }
      printf("\n");
      i++;
   }
   return 0;
}
```

OUTPUT

```
*****
****
***
**
*
```

5)#include <stdio.h>

```
int main() {
    int rows, i = 1, j, k;

    printf("Enter the number of rows: ");
    scanf("%d", &rows);

    while (i <= rows) {

        j = 1;
        while (j <= rows - i) {
            printf(" ");
            j++;
        }

        k = 1;
        while (k <= (2 * i - 1)) {
            printf("*");
            k++;
        }
        printf("\n");
        i++;
    }

    return 0;
}
```

OUTPUT

```
Enter the number of rows: 4
      *
    * * *
 * *  *  *  *
 * * * * * *  *
```

6)

```
//WAP TO PRINT NUMBERS BETWEEN 1 TO 10 USING DO while
#include<stdio.h>
int main()
{
    int i=1;
    do{
        printf("%d\n",i);
        i++;

    }while(i<=10);


    return 0;
}
```

OUTPUT

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

```
7
8
9
10
```

```
7)//WAP TO PRINT MULTIPLICATION TABLE upto 10
#include<stdio.h>
int main()
{
   int i=1,j;
   do{
      j=1;
      do{
      printf("%d*%d=%d\t",i,j,i*j);
      j++;
      }while(j<=10);
      i++;
      printf("\n");

   }while(i<=10);


   return 0;
}
```

Output

```
1*1=1  1*2=2  1*3=3  1*4=4  1*5=5  1*6=6  1*7=7  1*8=8  1*9=9  1*10=10
2*1=2  2*2=4  2*3=6  2*4=8  2*5=102*6=122*7=142*8=162*9=182*10=20
3*1=3  3*2=6  3*3=9  3*4=123*5=153*6=183*7=213*8=243*9=273*10=30
4*1=4  4*2=8  4*3=124*4=164*5=204*6=244*7=284*8=324*9=364*10=40
5*1=5  5*2=105*3=155*4=205*5=255*6=305*7=355*8=405*9=455*10=50
6*1=6  6*2=126*3=186*4=246*5=306*6=366*7=426*8=486*9=546*10=60
7*1=7  7*2=147*3=217*4=287*5=357*6=427*7=497*8=567*9=637*10=70
8*1=8  8*2=168*3=248*4=328*5=408*6=488*7=568*8=648*9=728*10=80
9*1=9  9*2=189*3=279*4=369*5=459*6=549*7=639*8=729*9=819*10=90
10*1=10         10*2=20         10*3=30         10*4=40         10*5=50         10*6=60
10*7=70         10*8=80         10*9=90         10*10=100
```

```
8)//WAP TO calculate sum of first n natural numbers
#include<stdio.h>
int main()
{
    int i,j,sum=0,n;
    printf("enter limit");
    scanf("%d",&n);
    for(i=1;i<=10; i++){
            sum=sum+i;
        }
    printf("sum=%d",sum);
    return 0;
}
```

Output

```
enter limit 10
sum=55
```

11)//WAP TO calculate reverse a number

```
#include<stdio.h>
int main()
{
    int n,reverse=0,rem;
    printf("enter value of the number to reverse");
    scanf("%d",&n);
    for(int i=1;n!=0; i++){
            rem=n%10;
            reverse=reverse*10+rem;
            n=n/10;
        }
    printf("reverse=%d",reverse);
    return 0;
}
```

OUTPUT

```
enter value of the number to reverse1563
reverse=3651
```

12)//WAP TO print fibinoccae series

```
#include<stdio.h>
int main()
{
    int n1=0,n2=1,n3,n;
    printf("enter limit=");
    scanf("%d",&n);
    if(n>=1){
        printf("%d\t",n1);
    }
    if(n>=2){
        printf("%d\t",n2);
    }
    for(int i=1;i<=n; i++){
            n3=n1+n2;
            n1=n2;
            n2=n3;
            printf("%d\t",n3);
        }
    return 0;
}
```

OUTPUT

```
enter limit=5
0       1       1       2       3       5       8
```

```
13)//infinite for loop
#include<stdio.h>
int main()
{
    int i=1;
    printf("prgm executed");
    for(;;);

    return 0;
}
```

```
14)#include<stdio.h>
int main()
{
  int rows;
  printf("enter no of rows");
  scanf("%d",&rows);
  for(int i=1;i<=rows;i++)
  {
     for(int j=1;j<=rows-i;j++)
     {
         printf(" ");
     }
     for(int k=1;k<=i;k++)
     {
         printf(" 1");
     }
     printf("\n");
  }



    return 0;
}
```

```
enter no of rows5
     1
    1 1
   1 1 1
  1 1 1 1
 1 1 1 1 1
```

```
15)#include<stdio.h>
#include<stdlib.h>
#include<time.h>

int main() {
    int lower, upper, guess;
    srand(time(NULL));
    printf("Enter upper limit: ");
    scanf("%d", &upper);
    printf("Enter lower limit: ");
    scanf("%d", &lower);
     int random_number = (rand() % (upper - lower + 1)) + lower;
    int attempts = 5;
    for (int i = attempts; i > 0; i--) {
        printf("You have %d tries left.\n", i);
        printf("Enter a guess: ");
        scanf("%d", &guess);
        if (guess < lower || guess > upper) {
            printf("Please enter a number between %d and %d.\n\n", lower, upper);
        }
        if (guess == random_number) {
            printf("Congratulations! You guessed it!\n");
            break;
        }

        else if (guess > random_number) {
            printf("Sorry, %d is too high. My number is lesser than that.\n\n", guess);
        } else {
            printf("Sorry, %d is too low. My number is greater than that.\n\n", guess);
        }
    }
     printf("The number was %d.\n", random_number);

    return 0;
}
```

OUTPUT

```
[?2004l
Enter upper limit: 20
Enter lower limit: 0
You have 5 tries left.
Enter a guess: 15
Sorry, 15 is too high. My number is lesser than that.

You have 4 tries left.
Enter a guess: 6
Sorry, 6 is too high. My number is lesser than that.

You have 3 tries left.
Enter a guess:
2
Sorry, 2 is too high. My number is lesser than that.

You have 2 tries left.
Enter a guess: 3
Sorry, 3 is too high. My number is lesser than that.

You have 1 tries left.
Enter a guess: 8
Sorry, 8 is too high. My number is lesser than that.

The number was 1.
```

CONTINUE

```
16)#include<stdio.h>
int main()
{
    int i;
    for(i=0;i<=10;i++)
    {
        if(i==5){
          continue;
        }
        printf("%d\n",i);
    }
    return 0;
}
```

OUTPUT

```
0
1
2
3
4
6
7
8
9
10
```

BREAK

```
#include<stdio.h>
int main()
{
    int i;
    for(i=0;i<=10;i++)
    {
        if(i==5){

          break;
        }
        printf("%d\n",i);
    }
    return 0;
}
```

OUTPUT

```
0
1
2
3
4
```

```
18)#include<stdio.h>
int main()
{
   int n,sum=0;
   printf("enter upto 20 integers(enter -1 to stop)");
   for(int i=1;i<=20;i++)
   {
       scanf("%d",&n);
       if(n==-1)
       {
           break;
       }
        else if(n<0 )
       {
           continue;

       }else if(n%2==0)
       {
          sum=sum+n;
       }
   }

    printf("sum of even numbers =%d",sum);

    return 0;
}
```

Output

```
enter upto 20 integers(enter -1 to stop) 4
7
-3
2
8
-5
10
-1
sum of even numbers =24
```

20)Problem Statement 1: Banking System Simulation

Description: Create a simple banking system simulation that allows users to create an account, deposit money, withdraw money, and check their balance. The program should

handle
multiple accounts and provide a menu-driven interface.

Requirements:
1. Use appropriate data types for account balance (e.g., float for monetary values) and user input (e.g., int for account numbers).

2. Implement a structure to hold account details (account number, account holder name, balance).

3. Use control statements to navigate through the menu options:

    i. Create Account
    ii. Deposit Money
    iii. Withdraw Money
    iv. Check Balance

4. Ensure that the withdrawal does not exceed the available balance and handle invalid inputs gracefully.

Example Input/Output:

```
 Welcome to the Banking System
 1. Create Account
 2. Deposit Money
 3. Withdraw Money
 4. Check Balance
 5. Exit
 Choose an option: 1
 Enter account holder name: John Doe
 Account created successfully! Account Number: 1001


Choose an option: 2
 Enter account number: 1001
 Enter amount to deposit: 500
 Deposit successful! New Balance: 500.0
```

```
Choose an option: 3
 Enter account number: 1001
 Enter amount to withdraw: 200
 Withdrawal successful! New Balance: 300.0


Choose an option: 4
 Enter account number: 1001
 Current Balance: 300.0


Choose an option: 5
 Exiting the system.
```

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

int main() {
    char name[30];
    char acc_num[30] = "";
    float acc_balance = 0.0;
    int op;

    while(1) {
        printf("\nWelcome to the Banking System\n");
        printf("1. Create Account\n2. Deposit Money\n3. Withdraw Money\n4. Check Balance\n5. Exit\n");
printf("Choose an option: ");
scanf("%d", &op);

switch(op) {
    case 1: {
        printf("Enter account holder name: ");
        getchar();
        scanf("%s", name);
        printf("Enter Account Number: ");
        scanf("%s", acc_num);

        printf("Account created successfully!\n");
        acc_balance = 0.0;
        break;
    }

    case 2: {
        char new_acc_num[30];
        float deposit_amount;

        printf("Enter account number: ");
        scanf("%s", new_acc_num);

        if(strcmp(acc_num, new_acc_num) == 0) {
            printf("Enter amount to deposit: ");
            scanf("%f", &deposit_amount);
        acc_balance += deposit_amount;
        printf("Deposit successful! New Balance: %.2f\n", acc_balance);
    } else {
        printf("Wrong account number! Try again!\n");
    }
    break;
}

case 3: {
    char new_acc_num[30];
    float withdraw_amount;

    printf("Enter account number: ");
    scanf("%s", new_acc_num);

    if(strcmp(acc_num, new_acc_num) == 0) {
        printf("Enter amount to withdraw: ");
        scanf("%f", &withdraw_amount);

        if(withdraw_amount > acc_balance) {
            printf("Insufficient balance! Your current balance is %.2f\n", acc_balance);
        } else {
            acc_balance -= withdraw_amount;
            printf("Withdrawal successful! New Balance: %.2f\n", acc_balance);
        }
    } else {
        printf("Wrong account number! Try again!\n");
    }
    break;
}

case 4: {
    char new_acc_num[30];

    printf("Enter account number: ");
    scanf("%s", new_acc_num);

    if(strcmp(acc_num, new_acc_num) == 0) {
        printf("Current Balance: %.2f\n", acc_balance);
    } else {
        printf("Wrong account number! Try again!\n");
    }
    break;
}

case 5: {
    printf("Exiting the system.\n");
    return 0;
}

default:
```

```
                printf("Invalid option! Please choose a valid option.\n");
                break;
        }
    }

    return 0;
}
```

OUTPUT

```
[?2004l

Welcome to the Banking System
1. Create Account
2. Deposit Money
3. Withdraw Money
4. Check Balance
5. Exit
Choose an option: 1
Enter account holder name: NAVYA
Enter Account Number: 1234
Account created successfully!

Welcome to the Banking System
1. Create Account
2. Deposit Money
```

```
3. Withdraw Money
4. Check Balance
5. Exit
Choose an option: 2
Enter account number: 1234
Enter amount to deposit: 500
Deposit successful! New Balance: 500.00

Welcome to the Banking System
1. Create Account
2. Deposit Money
3. Withdraw Money
4. Check Balance
5. Exit
Choose an option: 3
Enter account number:
1234
Enter amount to withdraw: 200
Withdrawal successful! New Balance: 300.00

Welcome to the Banking System
1. Create Account
2. Deposit Money
3. Withdraw Money
4. Check Balance
5. Exit
```

```
Choose an option: 4

Enter account number: 1234

Current Balance: 300.00


Welcome to the Banking System

1. Create Account

2. Deposit Money

3. Withdraw Money

4. Check Balance

5. Exit

Choose an option: 5

Exiting the system.
```

17)TEMPERATURE SYSTEM

```
#include <stdio.h>


int main()

{

    float temp[40],highest,lowest;
```

```
int sum=0,count=0;
printf("enter temperature for each day of month(30 days)\n");
for (int i = 0; i < 30; i++) {
    printf("Day %d temperature: ", i + 1);
    scanf("%f", &temp[i]);
    if (temp[i] == -1) {
        break;
    }
   sum += temp[i];
}
float average=sum/30;
for(int i=0;i<=30;i++)
{
    if(temp[i]>average)
    {
     count=count+1;
    }
}
 highest=temp[0];
 lowest=temp[0];
for(int i=0;i<=30;i++)
{

    if(temp[i]>highest)
    {
        highest=temp[i];
        }
        if(temp[i]<lowest){
            lowest=temp[i];
        }
    }
    printf("average temp of the month is %f\n",average);
    printf("highest temperature is %f\n",highest);
    printf("lowest temperature is %f\n",lowest);
    printf("no of days above average temperature is %d\n",count);
    return 0;
}
```

OUTPUT

```
enter temperature for each day of month(30 days)
Day 1 temperature: 25
Day 2 temperature: 56
Day 3 temperature: 45
Day 4 temperature: 30
Day 5 temperature: -1
average temp of the month is 5.000000
highest temperature is 56.000000
lowest temperature is -1.000000
no of days above average temperature is 4
```

Problem Statement : Inventory Management System

Description: Create an inventory management system that allows users to manage products in a store. Users should be able to add new products, update existing product quantities, delete products, and view inventory details.

Requirements:

1. Use appropriate data types for product details (e.g., char arrays for product names, int for quantities, float for prices).

2. Implement a structure to hold product information.

3. Use control statements for menu-driven operations:
   i. Add Product
   ii. Update Product Quantity
   iii. Delete Product
   iv. View All Products in Inventory

4. Ensure that the program handles invalid inputs and displays appropriate error messages.

Example Input/Output:

```
Inventory Management System
 1. Add Product
 2. Update Product Quantity
 3. Delete Product
 4. View All Products in Inventory
 5. Exit


Choose an option: 1
 Enter product name: Widget A
 Enter product quantity: 50
 Enter product price: 19.99


Choose an option: 4
 Product Name: Widget A, Quantity: 50, Price: $19.99


Choose an option: 5
 Exiting the system.
```

```
#include<stdio.h>

#include<string.h>

int main()

{

    char prdct_names[30][30];

    char quantity[20][20];

    int option;

    float price[40];

    char i=0;

    printf("INVENTORY MANAGEMNET SYSTEM\n");

    while(1){

      printf("1.Add a product\n2.Update product Quantity\n3.Delete product\n4.View all products in invenetory\n5.Exit\n");

       printf("choose an option\n");

       scanf("%d",&option);

       switch(option){
```

```
case 1:
      printf("enter product name:");
      scanf("%s",prdct_names[i]);
      printf("enter product quantity:");
      scanf("%s",quantity[i]);
      printf("enter price:");
      scanf("%f",&price[i]);
      i++;
      break;
 case 2:
       char new_prdctname[20];
       printf("enter product name you want to update its quantity\n");
       scanf("%s",new_prdctname);
       for(int j=0;j<i;j++){
         if(strcmp(new_prdctname,prdct_names[j])==0){
          printf("enter new quantity for %s:",new_prdctname);
          scanf("%s",quantity[j]);
       }

       }

      break;

 case 3:
       char delete_prdct[30];
       printf("enter the item you want to delete");
```

```
              scanf("%s",delete_prdct);
              for(int j=0;j<i;j++)
              {
                 if(strcmp(delete_prdct,prdct_names[i])==0){
                   for (int k = j; k < i - 1; k++) {
                     strcpy(prdct_names[k], prdct_names[k + 1]);
                     strcpy(quantity[k], quantity[k + 1]);
                     price[k] = price[k + 1];
               }
                 }
                 i--;
              }
              printf("%s is deleted",delete_prdct);
              break;

          case 4:
              for(int j=0;j<i;j++)
              {
                 printf("name:%s\tQuantity:%s\tPrice:%f\n",prdct_names[j],quantity[j],price[i]);
              }
              break;
      }
    }


}
[?2004l
```

```
INVENTORY MANAGEMNET SYSTEM
1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
choose an option
1
enter product name:apple
enter product quantity:4
enter price:56
1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
choose an option
3 2
enter product name you want to update its quantity
apple
enter new quantity for apple:2
1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
```

```
choose an option
1
enter product name:orange
enter product quantity:6
enter price:5
1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
choose an option
3
enter the item you want to deleteorange
orange is deleted1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
choose an option
4
name:apple    Quantity:2      Price:5.000000
1.Add a product
2.Update product Quantity
3.Delete product
4.View all products in invenetory
5.Exit
```

choose an option