Computer Programming File

Name: Prajwal Krishnamurthy Bhat

Roll no.: 2021UCS1609

Class: CSE-1

1. Write a program in C to show that right shift effectively divides a number by 2 and a left shift effectively multiplies a number by 2

Code:

#include <stdio.h>

#include <conio.h>

int main()

{

    int a;

    int multi;

    printf("Enter the number: ");

    scanf("%d", &a);

    printf("The number multiplied by 2: %d\n",a\*2);

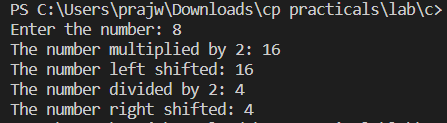
    printf("The number left shifted: %d\n",a<<1);

    printf("The number divided by 2: %d\n",a/2);

    printf("The number right shifted: %d",a>>1);

}

Output:



2. Using if else statements, write a program in ‘C’ to read two integers from the user and display the quotient. Your program should be able to detect divide by zero.

Code:

#include <stdio.h>

#include <conio.h>

int main()

{

    int a;

    int b;

    printf("Enter the dividend: ");

    scanf("%d", &a);

    printf("Enter the divisor: ");

    scanf("%d",&b);

    if(b==0)

    {

        printf("\nCannot divide by zero");

    }

    else

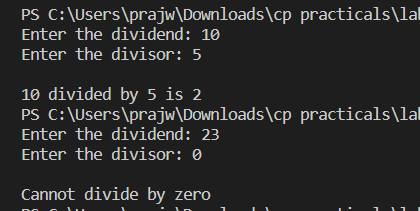
    {

        printf("\n%d divided by %d is %d",a,b,a/b);

    }

}

Output:



3. Write down a function in C to implement bitwise AND, OR, XOR and NOT operations.

Code:

#include <stdio.h>

#include <conio.h>

int main()

{

    int a;

    int b;

    printf("Enter the first number: ");

    scanf("%d", &a);

    printf("\nEnter the second number: ");

    scanf("%d", &b);

    printf("\n\n%d&%d is %d",a,b,a&b);

    printf("\n%d|%d is %d",a,b,a|b);

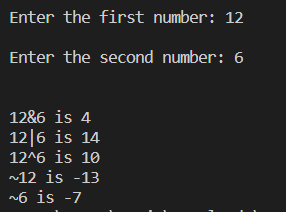
    printf("\n%d^%d is %d",a,b,a^b);

    printf("\n~%d is %d",a,~a);

    printf("\n~%d is %d",b,~b);

}

Output:



4. Write a program in C that uses a two-dimensional array to store the numeric grade for each student (n) in a multiple teacher’s class (m). The program assumes that the teacher has three classes and a maximum of 30 students per class. Both the variable M and N should be user defined.

Code:

#include <stdio.h>

int main()

{

    int m,n,i,j;

    printf("Enter the number of classes: ");

    scanf("%d",&m);

    printf("Enter the number of students: ");

    scanf("%d",&n);

    int arr[n][m];

    if (m < 30){

        for(i=0; i<n; i++)

        {

            for(j=0; j<m; j++)

            {

                printf("Enter the grade of student %d in class %d: ",i+1,j+1);

                scanf("%d",&arr[i][j]);

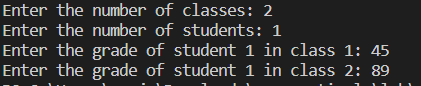
            }

        }

    }

}

Output:



5. User has given a random size string to input, you have to calculate the length of the string using pointer. You cannot use predefined function strrev.

Code:

#include <stdio.h>

int main()

{

    int i=0;

    int length=0;

    char string[100];

    char \*given\_string=&string[i];

    printf("Enter the string: ");

    scanf("%s",given\_string);

    for(i=0;string[i]!=0;i++);

    printf("Length of input string: %d\n",i);

}

Output:



6. Write the program to input the value of age of employees in the company. You have to calculate the average age of the employee in the company using pointer of array.

Code:

#include <stdio.h>

int main()

{

    int n, \*nptr=&n;

    float num[100],sum=0.0,avg;

    printf("Enter the number of employees: ");

    scanf("%d",nptr);

    for (int i=0; i<n; i++)

    {

        float \*numptr=&num[i];

        printf("Enter the age of employee no. %d: ", i+1);

        scanf("%f",numptr);

        sum+=\*numptr;

    }

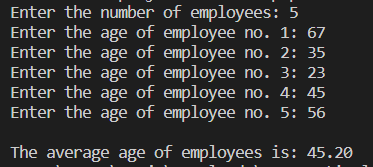
    avg=sum/n;

    printf("\nThe average age of employees is: %.2f",avg);

    return 0;

}

Output:



7. Write a program in C using pointers to implement insertion and deletion in a queue.

Code:

#include <stdio.h>

#include <stdlib.h>

#define MAX 50

void insert();

void del();

void display();

int queue[MAX];

int front=-1,rear=-1;

int main()

{

    int choice,\*chptr=&choice;

    while (1)

    {

        printf("\n1.Insert an element.\n2.Delete an element.\n3.Display the queue\n4.Quit.\nEnter your choice: ");

        scanf("%d",chptr);

        switch(choice)

        {

            case 1:

                insert();

                break;

            case 2:

                del();

                break;

            case 3:

                display();

                break;

            case 4:

                exit(1);

            default:

                printf("\nYou chose unwisely.");

        }

    }

}

void insert()

{

    int item;

    if(rear==MAX-1)

        printf("\nQueue overflow\n");

    else

    {

        if(front==-1)

            front=0;

        printf("\nEnter the element to be inserted: ");

        scanf("%d",&item);

        rear++;

        int \*qptr=&queue[rear];

        \*qptr=item;

    }

}

void del()

{

    if(front == - 1 || front > rear)

    {

        printf("\nQueue Underflow\n");

        return;

    }

    else

    {

        printf("\nElement %d was deleted\n",queue[0]);

        for (int i=0; i<rear; i++)

            {

                queue[i]=queue[i+1];

            }

        rear--;

        if(rear==-1)

        {

            front=-1;

        }

    }

}

void display()

{

    int i;

    if(front==-1)

        printf("\nQueue is empty");

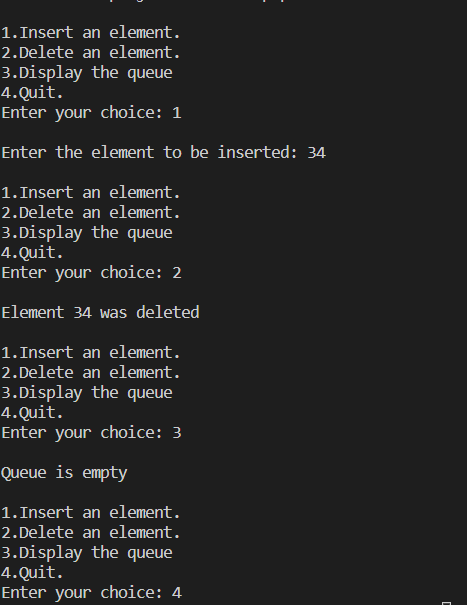
    printf("\n");

    for(i==front; i<=rear;i++)

        printf("%d\n", queue[i]);

}

Output:



8. A start-up owner is interested in maintaining a dataset of the newly recruited employees. She is interested in storing the Emp\_Name (Str), Emp\_Age (int), Emp\_Degree (Str), Emp\_Exp (Float), Emp\_add (Structure). Emp\_add needs one user defined data to store street no., city, district and state for the employee address. You have to design a database where we can store all the information for at least 20 employees.

Code:

#include <stdio.h>

#include <string.h>

int main()

{

    int intph;

    float floatph;

    char charph[100];

    int n=0;

    int x;

    struct employee

    {

        char emp\_name[50];

        int emp\_age;

        char emp\_degree[100];

        float emp\_exp;

        struct emp\_add

        {

            int street\_no;

            char city[50];

            char district[50];

            char state[50];

        }add;

    };

    printf("Enter the number of employees: ");

    scanf("%d", &x);

    struct employee e[x];

    for(int i=0; i<x; i++)

    {

        printf("Enter the employee name: ");

        scanf("%s",&charph);

        strcpy(e[i].emp\_name,charph);

        printf("Enter the employee age: ");

        scanf("%d",&intph);

        e[i].emp\_age=intph;

        printf("Enter the employee degree: ");

        scanf("%s",&charph);

        strcpy(e[i].emp\_degree,charph);

        printf("Enter the employee experience: ");

        scanf("%f",&floatph);

        e[i].emp\_exp=floatph;

        printf("Enter the employee street number: ");

        scanf("%d",&intph);

        e[i].add.street\_no=intph;

        printf("Enter the employee city: ");

        scanf("%s",&charph);

        strcpy(e[i].add.city,charph);

        printf("Enter the employee district: ");

        scanf("%s",&charph);

        strcpy(e[i].add.district,charph);

        printf("Enter the employee state: ");

        scanf("%s",&charph);

        strcpy(e[i].add.state,charph);

        printf("\n");

        n++;

    }

    if(n!=0)

    {

        for(int j=0; j<n; ++j)

        {

            printf("\nNo: %d\nName: %s\nAge: %d\nDegree: %s\nExperience: %.2f\nStreet No: %d\nCity: %s\nDistrict: %s\nState: %s\n",

                   j+1,e[j].emp\_name,e[j].emp\_age,e[j].emp\_degree,e[j].emp\_exp,

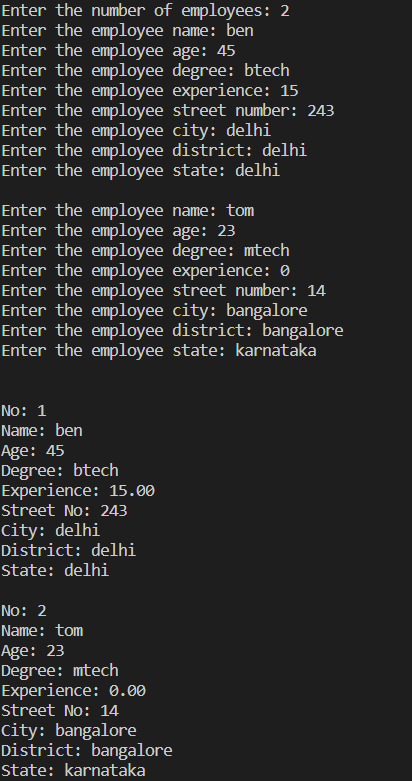
                   e[j].add.street\_no,e[j].add.city,e[j].add.district,e[j].add.state);

        }

    }

}

Output:



9. Develop a mailing list program in ‘C’ to hold the address information that includes name, street, city, state and zip code. Your program should include the following functionalities a. Add mailing address b. Delete mailing address

Code:

#include <stdio.h>

#include <stdlib.h>

#include <conio.h>

void add();

void del();

void display();

int counter=0;

int pos;

char choice;

struct address

    {

        char name[50];

        char street[50];

        char city[50];

        char state[50];

        char pin\_code[100];

    }a[100];

int main()

{

    while(1)

    {

        printf("\n1. Add a record.");

        printf("\n2. Delete a record.");

        printf("\n3. Display all records.");

        printf("\n4. Quit");

        printf("\nEnter your choice: ");

        choice=getche();

        switch(choice)

        {

        case '1':

            add();

            counter++;

            break;

        case '2':

            del();

            counter--;

            break;

        case '3':

            display();

            break;

        case '4':

            exit(0);

        default:

            printf("\nInvalid Choice");

        }

    }

}

void add()

{

    printf("\n\nEnter the name: ");

    scanf("%s",a[counter].name);

    printf("\nEnter the street: ");

    scanf("%s",a[counter].street);

    printf("\nEnter the city: ");

    scanf("%s",a[counter].city);

    printf("\nEnter the state: ");

    scanf("%s",a[counter].state);

    printf("\nEnter the pin code: ");

    scanf("%s",a[counter].pin\_code);

}

void del()

{

    printf( "\n\nDefine the position of the array element where you want to delete: ");

    scanf ("%d",&pos);

    if (pos>counter){

        printf ("\nInvalid Position");

    }

    else{

        for (int i=pos-1; i<counter-1; i++){

            a[i]=a[i+1];

        }

        printf ("\nThe record has been deleted\n");

    }

}

void display()

{

    if(counter==0){

        printf("\n\nThere are no addresses");

    }

    else{

        for(int i=0; i<counter;i++){

            printf("\n\nNo: %d\nName: %s\nStreet: %s\nCity: %s\nState: %s\nPin Code: %s\n",

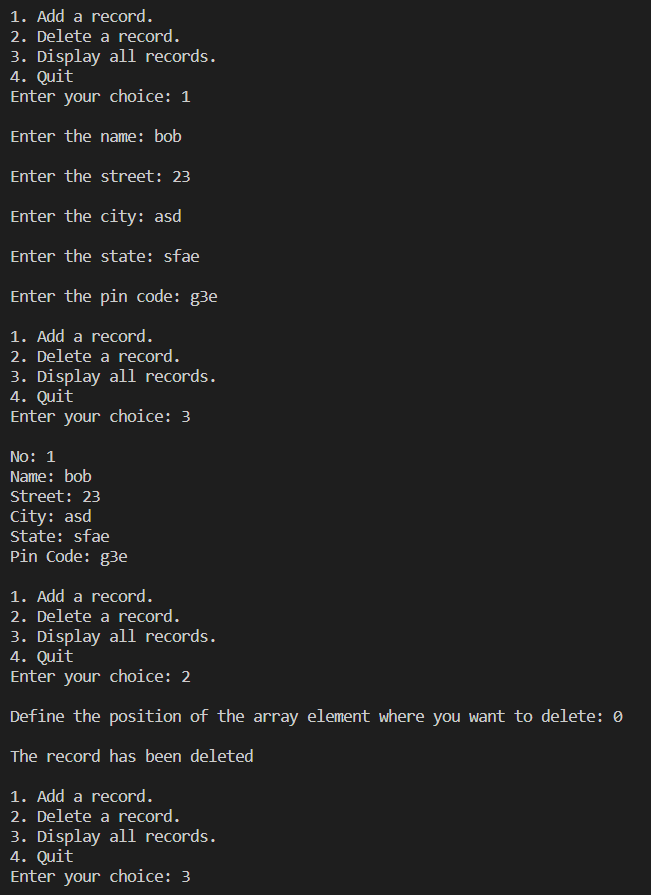
                   i+1,a[i].name,a[i].street,a[i].city,a[i].state,a[i].pin\_code);

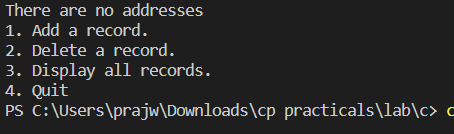
        }

    }

}

Output:





10. Write a program in ‘C’ using functions that reads in a line of text on a character-by-character basis, and then displays the characters in reverse order.

Code:

#include <stdio.h>

#include <string.h>

void reverse(char str[]){

    int len, i;

    len =strlen(str);

    for (int i =len-1; i>=0; --i){

        printf("%c", str[i]);

    }

}

int main() {

    char str[50];

    printf("Enter the string: ");

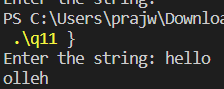
    gets(str);

    reverse(str);

    return 0;

}

Output:



11. Write a function which implements a pascal triangle for a specified number (Using recursion).

Code:

#include <stdio.h>

long pascal(int, int);

int main()

{

   int n, m, k, s;

   printf ("Enter number of rows: ");

   scanf("%d", &n);

   n-=1;

   for(k = 0; n >= k; k++) {

            for(s = 0; s < n-k; s++)

                    printf(" ");

            for(m = 0; k >= m; m++) {

                    long f = pascal(k, m);

                    printf("%ld ", f);

            }

            printf("\n");

    }

    return 0;

}

long pascal(int n, int i)

{

        if(n == i || i == 0)

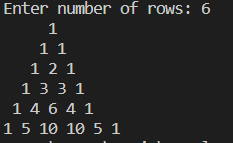
                return 1;

        else

                return pascal(n-1, i) + pascal(n-1, i-1);

}

Output:



12. Write a program in ‘C’ that reads files and displays them on the screen.

Code:

#include <stdio.h>

#include <stdlib.h>

int main()

{

    FILE \* fptr;

    char c;

    fptr = fopen("file.txt", "r");

    while(c != EOF){

        c = getc(fptr);

        printf("%c", c);

    }

    fclose(fptr);

    return 0;

}

Output:



13. Using fopen(), fgetc(), fputc() and fclose() functions write a program that reads characters from the keyboard and writes them to a disk file until the user types a dollar sign. The filename is specified from the command line.

Code:

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

void main()

{

    FILE \*fptr;

    char ch,file[10];

    printf("Enter file name: ");

    scanf("%s",file);

    fptr=fopen(file,"w");

    if(fptr==NULL)

    {

        printf("Unable to open file");

        exit(0);

    }

    else{

        while(1)

        {

            scanf("%c", &ch);

            if(ch=='$')

                exit(0);

            putc(ch,fptr);

        }

    }

    fclose(fptr);

}

Output:



14. Build a collection of employee records, You may use data structures with any combination of built in object types (lists, tuples, dictionaries, strings, numbers). Then access and modify the individual records and individual components of data structure using indexing.

Code:

employee={}

def add():

    counter=1

    employee[counter]={}

    employee[counter]['name']=input("Enter the employee name: ")

    employee[counter]['age']=input("Enter the employee age: ")

    employee[counter]['salary']=input("Enter the employee salary: ")

    employee[counter]['degree']=input("Enter the employee degree: ")

    print("")

    counter+=1

def delete():

    while True:

        try:

            e\_delete=int(input("Enter the ID of the employee to be deleted: "))

        except ValueError:

            print("That's not a valid ID")

        else:

            try:

                del employee[e\_delete]

            except KeyError:

                print("That's not a valid ID")

            else:

                print("The employee with ID "+str(e\_delete)+" was deleted!")

                print("")

                break

def display():

    if len(employee)==0:

        print("There are no employees\n")

    else:

        for e\_id, e\_info in employee.items():

            print("Employee ID:", e\_id)

            for key in e\_info:

                print(key + ':', e\_info[key])

        print("")

def modify():

    while True:

        try:

            id\_choice=int(input("Enter the id of the employee whose data has to be modified: "))

        except ValueError:

            print("That's not a valid ID")

        else:

            try:

                x=employee[id\_choice]

            except KeyError:

                print("That's not a valid ID")

            else:

                print("1. Modify the name")

                print("2. Modify the age")

                print("3. Modify the salary")

                print("4. Modify the degree")

                try:

                    modify\_choice=int(input("Enter your choice: "))

                except ValueError:

                    print("Invalid Choice")

                else:

                    if modify\_choice==1:

                        employee[id\_choice]['name']=input("Enter the employee name: ")

                        break

                    elif modify\_choice==2:

                        employee[id\_choice]['age']=input("Enter the employee age: ")

                        break

                    elif modify\_choice==3:

                        employee[id\_choice]['salary']=input("Enter the employee salary: ")

                        break

                    elif modify\_choice==4:

                        employee[id\_choice]['degree']=input("Enter the employee degree: ")

                        break

                    else:

                        print("Invalid Choice")

while True:

    print("1. Add an employee")

    print("2. Delete an employee")

    print("3. Display all employee")

    print("4. Modify data")

    print("5. Quit")

    try:

        choice=int(input("Enter your choice: "))

    except ValueError:

        print("Enter a number between 1 and 5")

    else:

        if choice==1:

            add()

        elif choice==2:

            delete()

        elif choice==3:

            display()

        elif choice==4:

            modify()

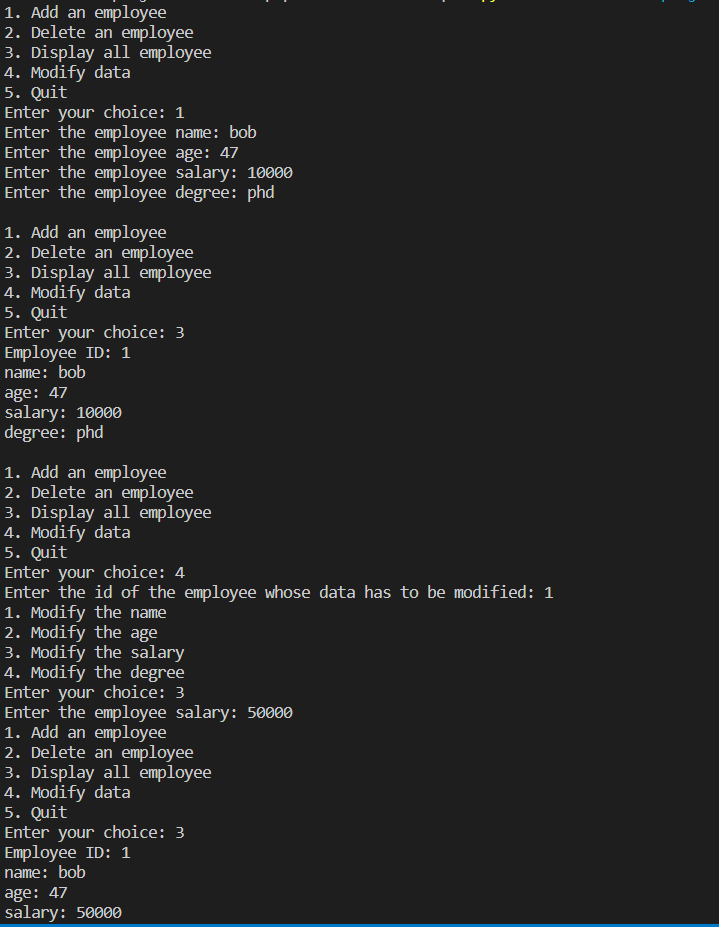
        elif choice==5:

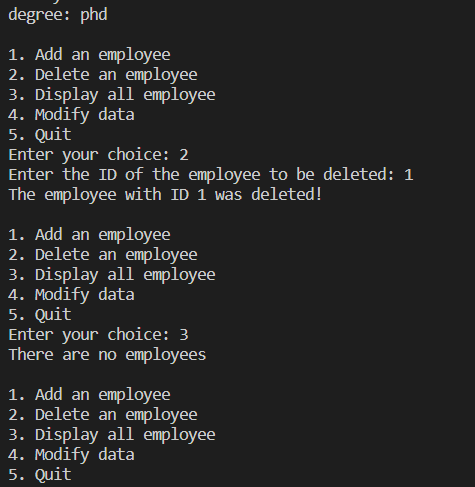
            break

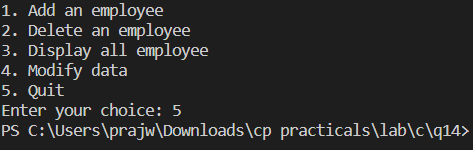
        else:

            print("Enter a number between 1 and 5")

Output:







15. Use lambda, map and filter functions to find the sum of cubes of first 5 even numbers.

Code:

numbers=[i for i in range(1,10)]

result=list(map(lambda n:n\*\*3,filter(lambda x:x%2==0,numbers)))

print(sum(result))

Output:



16. Write a program in Python to calculate percentage of marks in three subjects. Raise exceptions if marks entered are greater than the maximum marks.

Code:

i=0

for \_ in range(3):

    marks=int(input("Enter the marks obtained: "))

    maxmarks=int(input("Enter the maximum marks: "))

    result=(marks/maxmarks)\*100

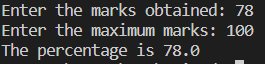
    if result>100:

        print("The marks entered are higher than maximum marks")

    else:

        print("The percentage is "+str(result))

Output:



17. Create a Package “Fun\_NSUT” with two Modules “Sports” and “Cultural”. In each of these two modules write at least two functions to just display information about a particular sport or cultural event. In your file “My\_interest”, import one function from each of these two Modules.

Code:

def video\_games():

    print('''Video games are defined based on their platform, which include arcade video games, console games,

    and personal computer (PC) games. More recently, the industry has expanded onto mobile gaming through

    smartphones and tablet computers, virtual and augmented reality systems, and remote cloud gaming.

    Video games are classified into a wide range of genres based on their type of gameplay and purpose''')

def hiking():

    print("Hiking is a long, vigorous walk, usually on trails or footpaths in the countryside.")

    video\_games()

    hiking()

def easter():

    print("Easter is a Christian festival and cultural holiday commemorating the resurrection of Jesus from the dead.")

def halloween():

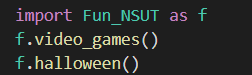
    print('''Halloween is a celebration observed in many countries on 31 October,

            the eve of the Western Christian feast of All Hallows' Day.''')

    easter()

    halloween()

Output:



18. Write down a Python code to copy from an existing file named “input.txt” to another file “output.txt”.

Code:

with open('input.txt','r') as f1:

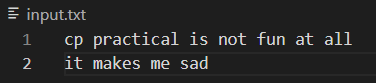
    words=f1.read()

with open('output.txt','w') as f2:

    f2.write(words)

Output:

Input.txt



Output.txt

