#include<iostream>

#include<string.h>

using namespace std;

class Person

{

string name, city;

int age;

public:

Person()

{

cout<<"Default Constructor called..."<<endl;

}

Person(int age, string name, string city)

{

this->age=age;

this->name=name;

this->city=city;

}

void setName(string name )

{

this->name=name;

}

string getName()

{

return name;

}

void setAge(int age)

{

this->age=age;

}

int getAge()

{

return age;

}

void setCity(string city)

{

this->city=city;

}

string getCity()

{

return city;

}

void display()

{

cout<<name<<" "<<age<<" "<<city<<endl;

}

};

int main()

{

Person p;

p.setName("jiva");

p.getName();

p.setAge(23);

p.getAge();

p.setCity("Pune");

p.getCity();

p.display();

return 0;

}

/\*1. Write a C program to input basic salary of an employee and calculate its Gross salary according to

following: Basic Salary <= 10000 : HRA = 20%, DA = 80% Basic Salary <= 20000 : HRA = 25%, DA = 90%

Basic Salary > 20000 : HRA = 30%, DA = 95% \*/

#include<stdio.h>

int main()

{

float gross,basic,hra,da;

printf("Please enter basic salary to calculate gross salary :");

scanf("%f",&basic);

if(basic<=10000)

{

hra=basic\*0.2;

da=basic\*0.8;

}

else if(basic<=20000)

{

hra=basic\*0.9;

da=basic\*0.25;

}

else

{

da=basic\*0.95;

hra=basic\*0.3;

}

gross=basic+hra+da;

printf("Gross salary for provided salary will be: %.2f",gross);

return 0;

}

/\*2. Create a class Date with data members as dd, mm, yy. Write getters and setters for all the data members. Also add the display function. Create Default and Parameterized constructors. Create the

object of this class in main method and invoke all the methods in that class.\*/

#include<iostream>

using namespace std;

class Date

{

int dd, mm, yy;

public:

//Default Constructor

Date()

{

cout<<"Default Constructor..."<<endl;

}

//Parameterized Constructor

Date(int dd, int mm, int yy)

{

this->dd=dd;

this->mm=mm;

this->yy=yy;

}

void setDay(int dd )

{

this->dd=dd;

}

int getDay()

{

return dd;

}

void setMonth(int mm)

{

this->mm=mm;

}

int getMonth()

{

return mm;

}

void setYear(int yy)

{

this->yy=yy;

}

int getYear()

{

return yy;

}

void display()

{

cout<<dd<<" / "<<mm<<" / "<<yy<<endl;

}

};

int main()

{

Date d;

d.setDay(01);

d.getDay();

d.setMonth(11);

d.getMonth();

d.setYear(1997);

d.getYear();

d.display();

return 0;

}

. Write a C program to input angles of a triangle and check whether triangle is valid or not.

#include<stdio.h>

int main()

{

int angle1,angle2,angle3,c;

printf("enter the angles of triangle :\n");

scanf("%d %d %d",&angle1,&angle2,&angle3);

if(angle1!=0)

{

if(angle2!=0)

{

if(angle3!=0)

{

c=angle1+angle2+angle3;

if(c==180)

printf("The triangle is valid.",c);

else

printf("The triangle is not valid.",c);

}

}

}

}

2. Find all prime no.'s in runtime array?

#include <stdio.h>

int main()

{

int n, i,flag;

flag = 0;

printf("Enter a positive integer: ");

scanf("%d", &n);

for (i = 2; i <= n / 2; ++i)

{

if (n % i == 0)

{

flag = 1;

break;

}

}

if (n == 1)

{

printf("1 is neither prime nor composite.");

}

else

{

if (flag == 0)

printf("%d is a prime number.", n);

else

printf("%d is not a prime number.", n);

}

}

#include <stdio.h>

int main()

{

int n, i,flag;

flag = 0;

printf("Enter a positive integer: ");

scanf("%d", &n);

for (i = 2; i <= n / 2; ++i)

{

if (n % i == 0)

{

flag = 1;

break;

}

}

if (n == 1)

{

printf("1 is neither prime nor composite.");

}

else

{

if (flag == 0)

printf("%d is a prime number.", n);

else

printf("%d is not a prime number.", n);

}

}

. Write a program to create an array of integers and perform following operations on that array like finding the sum, average, maximum and minimum number in that array. Accept the numbers of the array from user

#include<stdio.h>

int sumavg(int a[],int n)

{

int i,total=0,avg;

for (i = 0; i < n; i++)

{

total+=a[i];

}

avg = total / n;

printf("\nSum : %d\n", total);

printf("Average : %d\n", avg);

}

int minmax(int a[],int n)

{

int min,max,i;

min=max=a[0];

for(i=1; i<n; i++)

{

if(min>a[i])

min=a[i];

if(max<a[i])

max=a[i];

}

printf("Minimum : %d",min);

printf("\nMaximum : %d",max);

}

int main()

{

int a[10],i,n;

printf("\nEnter size of the array : ");

scanf("%d", &n);

printf("\nEnter elements in array : ");

for(i=0; i<n; i++)

{

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

}

minmax(a,n);

sumavg(a,n);

return 0;

}

. Accept a number and display it multiplication table.

#include<stdio.h>

int main()

{

int number,i;

printf("Please enter the number to print it's multiplication table: ");

scanf("%d",&number);

printf("Table of %d",number);

for(i=1;i<=10;i++)

printf("\n%d",number\*i);

}

3. Create a class Book with data members as bname,id,author,price. Write getters and setters for all the

data members. Also add the display function. Create Default and Parameterized constructors. Create

the object of this class in main method and invoke all the methods in that class.\*/

#include<iostream>

#include<string.h>

using namespace std;

class Book

{

string bname, author;

int id;

float price;

public:

//Default Constructor

Book()

{

cout<<"Default Constructor..."<<endl;

}

//Parameterized Constructor

Book(int id, string bname, string author,float price)

{

this->id=id;

this->bname=bname;

this->author=author;

this->price=price;

}

void setId(int id)

{

this->id=id;

}

int getId()

{

return id;

}

void setBookName(string bname )

{

this->bname=bname;

}

string getBookName()

{

return bname;

}

void setAuthor(string author)

{

this->author=author;

}

string getAuthor()

{

return author;

}

void setPrice(float price)

{

this->price=price;

}

float getPrice()

{

return price;

}

void display()

{

cout<<id<<" "<<bname<<" "<<author<<" "<<price<<endl;

}

};

int main()

{

Book b;

b.setId(1);

b.getId();

b.setBookName("Let Us C");

b.getBookName();

b.setAuthor("Yashvant Kanetkar");

b.getAuthor();

b.setPrice(450.50);

b.getPrice();

b.display();

return 0;

}

#include <stdio.h>

#include <stdlib.h>

int find\_length(char string[])

{

int len = 0, i;

for (i = 0; string[i] != '\0'; i++)

{

len++;

}

return len;

}

void join\_strings(char string1[], char string2[])

{

int i, len1, len2;

len1 = find\_length(string1);

len2 = find\_length(string2);

for (i = len1; i < len1 + len2; i++)

{

string1[i] = string2[i - len1];

}

string1[i] = '\0'; //adding null character at the end of input

}

int compare\_strings(char string1[], char string2[])

{

int len1, len2, i, count = 0;

len1 = find\_length(string1);

len2 = find\_length(string2);

if (len1 != len2)

return 1;

for (i = 0; i < len1; i++)

{

if (string1[i] == string2[i])

count++;

}

if (count == len1)

return 0;

return 1;

}

void copy\_string(char destination[], char source[])

{

int len, i;

len = find\_length(source);

for (i = 0; i < len; i++)

{

destination[i] = source[i];

}

destination[i] = '\0';

}

int main()

{

char string1[20], string2[20]; //string variables declaration with size 20

int choice;

while (1)

{

printf("\n1. Find Length \n2. Concatenate \n3. Compare \n4. Copy \n5. Exit\n");

printf("Enter your choice: ");

scanf("%d", & choice);

switch (choice)

{

case 1:

printf("Enter the string: ");

scanf("%s", string1);

printf("The length of string is %d", find\_length(string1));

break;

case 2:

printf("Enter two strings: ");

scanf("%s%s", string1, string2);

join\_strings(string1, string2);

printf("The concatenated string is %s", string1);

break;

case 3:

printf("Enter two strings: ");

scanf("%s%s", string1, string2);

if (compare\_strings(string1, string2) == 0)

{

printf("They are equal");

} else {

printf("They are not equal");

}

break;

case 4:

printf("Enter a string: ");

scanf("%s", string1);

printf("String1 = %s\n");

printf("After copying string1 to string 2\n");

copy\_string(string2, string1);

printf("String2 = %s", string2);

break;

case 5:

exit(0);

}

}

return 0;

}

4. Create a class Point with data members as x,y. Create Default and Parameterized constructors. Write

getters and setters for all the data members. Also add the display function. Create the object of this

class in main method and invoke all the methods in that class.\*/

#include<iostream>

using namespace std;

class Point

{

int x,y;

public:

//Default Constructor

Point()

{

cout<<"Default Constructor..."<<endl;

}

//Parameterized Constructor

Point(int x, int y)

{

this->x=x;

this->y=y;

}

void setXvalue(int x )

{

this->x=x;

}

int getXvalue()

{

return x;

}

void setYvalue(int y)

{

this->y=y;

}

int getYvalue()

{

return y;

}

void display()

{

cout<<"("<<x<<","<<y<<")"<<endl;

}

};

int main()

{

Point p;

p.setXvalue(5);

p.getXvalue();

p.setYvalue(4);

p.getYvalue();

p.display();

return 0;

}

4. Accept a number and display its sum of digits.

# include <stdio.h>

main()

{

int a,b,c,n, sum;

printf (" Enter a Three Digit Number:");

scanf("%d",&n);

a=n/100;

b=( (n%100)/10);

c=n%10;

sum=a+b+c;

printf("sum of 3 digit is %d",sum);

}

5. Accept a number and display whether its an Armstrong number

#include<stdio.h>

void main()

{

int a,b,sum,num;

printf("Enter the number:");

scanf("%d",&a);

num=a;

sum=0;

while(a>0)

{

b=a%10;

sum=sum+(b\*b\*b);

a=a/10;

}

if(sum==num)

printf("\nThe number %d is an armstrong number.",num);

else

printf("\nThe number %d is not an armstrong number.",num);

}

/\*5. Create a class ComplexNumber with data members real, imaginary. Create Default and

Parameterized constructors. Write getters and setters for all the data members. Also add the display

function. Create the object of this class in main method and invoke all the methods in that class.\*/

#include<iostream>

using namespace std;

class ComplexNumber

{

int real, imaginary;

public:

//Default Constructor

ComplexNumber()

{

cout<<"Default Constructor..."<<endl;

}

//Parameterized Constructor

ComplexNumber(int real, int imaginary)

{

this->real=real;

this->imaginary=imaginary;

}

void setReal(int real )

{

this->real=real;

}

int getReal()

{

return real;

}

void setImaginary(int imaginary)

{

this->imaginary=imaginary;

}

int getImaginary()

{

return imaginary;

}

void display()

{

cout<<real<<" + "<<imaginary<<"i"<<endl;

}

};

int main()

{

ComplexNumber c;

c.setReal(2);

c.getReal();

c.setImaginary(5);

c.getImaginary();

c.display();

return 0;

}

5. Enter data for two matrices. Multiply them to store result in third matrix & display result.

#include <stdio.h>

int main()

{

int row,col,i,j,a[row][col],transpose[col][row],k,l;

printf("Enter row size of 2D array to perform transpose operation: ");

scanf("%d",&row);

printf("Enter column size of 2D array to perform transpose operation: ");

scanf("%d",&col);

for(i=0;i<row;i++)

for(j=0;j<col;j++)

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

printf("\nTranspose Matrix: \n");

for(k=0;k<row;k++)

{

for(l=0;l<col;l++)

printf("\t%d",transpose[k][l]=a[l][k]);

printf("\n");

}

}

#include <stdio.h>

int main()

{

int OCTALVALUES[] = {0, 1, 10, 11, 100, 101, 110, 111};

long long octal, tempOctal, binary, place;

char hex[65] = "";

int rem;

place = 1;

binary = 0;

printf("Enter any octal number: ");

scanf("%lld", &octal);

tempOctal = octal;

while(tempOctal > 0)

{

rem = tempOctal % 10;

binary = (OCTALVALUES[rem] \* place) + binary;

tempOctal /= 10;

place \*= 1000;

}

while(binary > 0)

{

rem = binary % 10000;

switch(rem)

{

case 0:

strcat(hex, "0");

break;

case 1:

strcat(hex, "1");

break;

case 10:

strcat(hex, "2");

break;

case 11:

strcat(hex, "3");

break;

case 100:

strcat(hex, "4");

break;

case 101:

strcat(hex, "5");

break;

case 110:

strcat(hex, "6");

break;

case 111:

strcat(hex, "7");

break;

case 1000:

strcat(hex, "8");

break;

case 1001:

strcat(hex, "9");

break;

case 1010:

strcat(hex, "A");

break;

case 1011:

strcat(hex, "B");

break;

case 1100:

strcat(hex, "C");

break;

case 1101:

strcat(hex, "D");

break;

case 1110:

strcat(hex, "E");

break;

case 1111:

strcat(hex, "F");

break;

}

binary /= 10000;

}

strrev(hex);

printf("Octal number: %lld\n", octal);

printf("Hexadecimal number: %s", hex);

return 0;

}

#include <bits/stdc++.h>

using namespace std;

void towerOfHanoi(int n, char from\_rod,

char to\_rod, char aux\_rod)

{

if (n == 1)

{

cout << "Move disk 1 from rod " << from\_rod <<

" to rod " << to\_rod<<endl;

return ;

}

towerOfHanoi(n - 1, from\_rod, aux\_rod, to\_rod);

cout << "Move disk " << n << " from rod " << from\_rod <<

" to rod " << to\_rod << endl;

towerOfHanoi(n - 1, aux\_rod, to\_rod, from\_rod);

}

int main()

{

int n = 4;

towerOfHanoi(n, 'A', 'C', 'B');

return 0;

}

9. Write a C program to count total number of notes in given amount.

#include<stdio.h>

int main()

{

long int num;

int count = 0, rem;

printf("Enter a number: ");

scanf("%ld", &num);

while (num != 0)

{

rem = num % 10; // get the last digit of num

num = num / 10; // remove the last digit from num

count++; // increment count by 1

}

printf("%d", count);

return 0;

}

/\*10. Write a C program to invert the case of alphabet. \*/

#include<stdio.h>

#include<stdlib.h>

char get();

void change(char);

int main()

{

change(get());

return 0;

}

char get()

{

char c;

printf("\nPlease enter a character: ");

scanf("%c",&c);

return c;

}

void change(char c)

{

if(c>='A' && c<='Z')

c = 'a' + (c - 'A');

else if(c >= 'a' && c <='z')

c = 'A' + (c - 'a');

else

{

printf("\nInvalid input");

exit(0);

}

printf("\nChanged case: %c ",c);

}

1. Write a C program to print all natural numbers from 1 to n. - using while loop

#include<stdio.h>

int main()

{

int n,i;

printf("Enter the number :");

scanf("%d",&n);

i=1;

printf("\nThe natural numbers from %d to %d is",i,n);

while(n)

{

printf("\n%d",i++);

n--;

}

}

1. Write a C program to print all natural numbers revers from 1 to n. - using while loop

#include<stdio.h>

int main()

{

int n;

printf("enter the n value:");

scanf("%d",&n);

while(n>=1)

{

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

n--;

}

return 0;

}

13. Write a C program to find sum of all even and odd numbers between 1 to n.

#include<stdio.h>

int main()

{

int n,i,sumeven,sumodd;

sumeven=0;

sumodd=0;

printf("enter the number:");

scanf("%d",&n);

for(i=1;i<=n;i++)

{

if(i%2==0)

sumeven+=i;

else

sumodd+=i;

}

printf("\nthe sum of even number is %d",sumeven);

printf("\nthe sum of odd number is %d",sumodd);

}

15. Write a C program to find first and last digit of a number.

#include<stdio.h>

int main()

{

int n,a;

printf("enter the number:");

scanf("%d",&n);

a = n%10 ;

while(n >= 10)

{

n = n / 10;

}

printf("\nThe first digit of the number is %d ",n);

printf("\nThe last digit of the number is %d ",a);}

16. Write a C program to enter a number and print its reverse.

#include<stdio.h>

void main()

{

int n,sum=0,r;

printf("Enter a number:");

scanf("%d",&n);

while (n>0)

{

r=n%10;

sum=sum\*10+r;

n=n/10;

}

printf("\nsum is %d",sum);

printf("\n reverse number is %d",sum);

}

7. Write a C program to check whether a number is palindrome or not.

#include<stdio.h>

int main()

{

int n,a,reverse,remainder;

printf("enter the number:");

scanf("%d",&n);

a=n;

while (n != 0)

{

remainder = n % 10;

reverse = reverse \* 10 + remainder;

n = n/10;

}

if(a==reverse)

printf("the number is palindrome");

else

printf("the number is not palindrome");

}

18. Write a C program to find power of a number using for loop.

#include <stdio.h>

int main()

{

int base, exponent,power,i;

power = 1;

printf("Enter base: ");

scanf("%d", &base);

printf("Enter exponent: ");

scanf("%d", &exponent);

for(i=1; i<=exponent; i++)

{

power = power \* base;

}

#include <stdio.h>

typedef struct Time{

int hours;

int minutes;

int seconds;

}Time;

void updateMin(int,Time\*);

void updateSec(int,Time\*);

void setTime(Time\*);

void setHr(Time\*\*);

void setMin(Time\*\*);

void setSeconds(Time\*\*);

int main(){

int hr = 0,mm = 0, ss = 0;

printf("enter hrs\n");

scanf("%d",&hr);

printf("enter min\n");

scanf("%d",&mm);

printf("enter second\n");

scanf("%d",&ss);

Time t1 = {hr,mm,ss};

setTime(&t1);

printf("before update\n");

printf("%d % d %d \n",t1.hours,t1.minutes,t1.seconds );

updateMin(1,&t1);

updateSec(30,&t1);

printf("%d % d %d \n",t1.hours,t1.minutes,t1.seconds );

}

void setTime(Time \*temp){

setHr(&temp);

setMin(&temp);

setSeconds(&temp);

}

void setHr(Time \*\*temp){

int hr = (\*temp) -> hours;

if(hr >=0 && hr <24){

(\*temp) -> hours = hr;

}else{

(\*temp) -> hours = 0;

}

}

void setMin(Time \*\*temp){

int mm = (\*temp) -> minutes;

int r1 = 0;

if(mm >=0 && mm <60){

(\*temp)->minutes = mm;

// printf("%s\n", );

}else{

r1 = mm/60;

if((\*temp)->hours < 24){

(\*temp)->hours += r1;

(\*temp)->minutes = mm % 60;

}

if((\*temp)->hours > 23){

(\*temp)->hours = 0;

// temp.minutes = mm % 60;

}

}

}

void setSeconds(Time \*\*temp){

int ss = (\*temp) ->seconds;

int r2 = 0;

if(ss >=0 && ss < 60){

(\*temp)->seconds = ss;

}else{

r2 = ss /60;

if((\*temp)->minutes <= 59){

(\*temp)->minutes += r2;

(\*temp)->seconds = ss % 60;

}

if((\*temp)->minutes > 59){

(\*temp)->hours+=((\*temp)->minutes/60);

(\*temp)->minutes = 0;

}

if((\*temp)->hours > 23){

(\*temp)->hours = 0;

}

}

}

void updateMin(int mm, Time \*temp){

int iTemp = (temp) -> minutes;

(temp) -> minutes = iTemp+ mm;

if((temp)->minutes > 59){

setHr(&temp);

}

}

void updateSec(int ss, Time \*temp){

int iTemp = temp -> seconds;

temp -> seconds = iTemp + ss;

int iMin = (temp -> seconds) / 60;

if(temp -> seconds > 59){

updateMin(iMin,temp);

}

}

printf("%d ^ %d = %d", base, exponent, power);

}

19. Write a C program to find all factors of a number.

#include<stdio.h>

int main()

{

int n,i;

printf("enter the number:");

scanf("%d",&n);

printf("The factor of the number %d is\n",n);

for(i=1;i<=n;i++)

{

if(n%i==0)

{

printf("%d\t",i);

}

}

}

19. Write a C program to find all factors of a number.

#include<stdio.h>

int main()

{

int n,i;

printf("enter the number:");

scanf("%d",&n);

printf("The factor of the number %d is\n",n);

for(i=1;i<=n;i++)

{

if(n%i==0)

{

printf("%d\t",i);

}

}

}

21. Write a C program to find HCF (GCD) of two numbers.

#include<stdio.h>

int main()

{

int a,b,i,HCF;

printf("enter two numbers : \n");

scanf("%d %d",&a,&b);

for(i=1;i<=a && i<=b;i++)

{

if(a%i==0 && b%i==0)

HCF=i;

}

printf("HCF of the number is: %d",HCF);

}

#include <stdio.h>

int main()

{

int n1, n2, i, gcd;

printf("Enter two integers: ");

scanf("%d %d", &n1, &n2);

for(i=1; i<= n1 &&i<= n2; ++i)

{

// Checks if i is factor of both integers

if(n1%i==0 && n2%i==0)

gcd = i;

}

printf("G.C.D of %d and %d is %d", n1, n2, gcd);

return 0;

}

27. Write a C program to print Fibonacci series up to n terms.

#include <stdio.h>

int main()

{

int a, b, c, i, terms;

printf("Enter number of terms: ");

scanf("%d", &terms);

a = 0;

b = 1;

c = 0;

printf("Fibonacci terms: \n");

for(i=1; i<=terms; i++)

{

printf("%d, ", c);

a = b;

b = c;

c = a + b;

}

}

28.

#include<stdio.h>

#include<Math.h>

int main()

{

int i, j, k;

int z=1;

int n=6; //size

for(i=1;i<=n;i++)

{

for(j=n-1;j>=i;j--)

{

printf(" ");

}

for(k=1;k<=z;k++)

{

printf("%d",abs(k-i));

}

z=z+2;

printf("\n");

}

return 0;

}

. Find all odd and even no.'s in runtime array?

#include <stdio.h>

void main()

{

int i, num,o= 0,e = 0;

printf("Enter the value of num\n");

scanf("%d", &num);

for (i = 1; i <= num; i++)

{

if (i % 2 == 0)

e = e + i;

else

o =o+ i;

}

printf("\nSum of all even numbers = %d\n", e);

printf("\nSum of all odd numbers = %d\n", o);

}

// Write a program to sort a number of strings using bubble sort. Input is a number of strings and

// the output is the sorted list based on the length of strings.

// For e.g.: If input is jyoti, sareeka, anisha, sangita, savita, suja

#include<stdio.h>

#include <stdlib.h>

#include<string.h>

void swap(char\*\*,char\*\*);

int main(){

printf("enter the no of string \n");

int no;

scanf("%d",&no);

char\* arr[no];

char a[20];

int l,i;

char\* x;

printf("enter the names\n");

for (i = 0; i < no; i++) {

scanf(" %[^'\n']s", a); // taking values in char array

l = strlen(a);

// used malloc to allocate dynamic memory. l+1 to store "\0".

x = (char \*)malloc(l + 1);

strcpy(x, a);

arr[i] = x;

}

for(int i = 0; i < no -1 ; i++){

for(int j = 0; j < no -1; j++){

int size1 = (int) strlen(arr[j]);

int size2 = (int) strlen(arr[j+1]);

if(size1> size2){

swap(&arr[j],&arr[j+1]);

}

}

}

for(int i =0; i < no; i++){

printf("%s\n",arr[i]);

}

}

void swap(char\*\* ptr1, char\*\* ptr2)

{

char\* temp = \*ptr1;

\*ptr1 = \*ptr2;

\*ptr2 = temp;

}

#include<string.h>

#include<stdio.h>

#include<stdlib.h>

int main()

{

char \*\* ptop;

int noofnames;

char name[20];

int i;

printf("Enter no of strings: ");

scanf("%d",&noofnames);

ptop=(char \*\*)malloc(noofnames\*sizeof(char\*));

for(i=0;i<noofnames;i++)

{

printf("enter your name:");

scanf(" %[^'\n']s",name);

ptop[i]=(char \*)malloc(strlen(name)+1);

strcpy(ptop[i],name);

}

return 0;

}

#include <stdio.h>

int main()

{

unsigned int week;

printf("Enter week number (1-7): ");

scanf("%u", &week);

if(week == 1)

{

printf("Monday");

}

else if(week == 2)

{

printf("Tuesday");

}

else if(week == 3)

{

printf("Wednesday");

}

else if(week == 4)

{

printf("Thursday");

}

else if(week == 5)

{

printf("Friday");

}

else if(week == 6)

{

printf("Saturday");

}

else if(week == 7)

{

printf("Sunday");

}

else

{

printf("Invalid Input! Please enter week number between 1-7.");

}

return 0;

}