**17/18, Engineering Software C++**

**IVO NYOH KUKA**

**3332598**

**CSN**

Week 1: Lab Exercises

1. Display two messages. Write a program that displays Introduction to computers and welcome to object- oriented programming.

**CODE**

#include <iostream>

Using namespace std;

int main ()

{

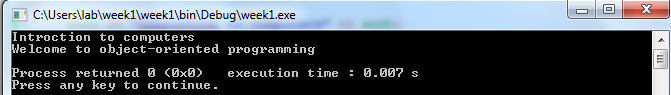
Cout<<”Introduction to computers”<<endl;

Cout <<”Welcome to object oriented programming”<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

using namespace std;

int main()

{

cout << " \* \*\*\*\*\*\*\*\*\*" << endl;

cout << " \*\*\* \*\*\*\*\*\*\*" << endl;

cout << " \*\*\*\*\* \*\*\*\*\*" << endl;

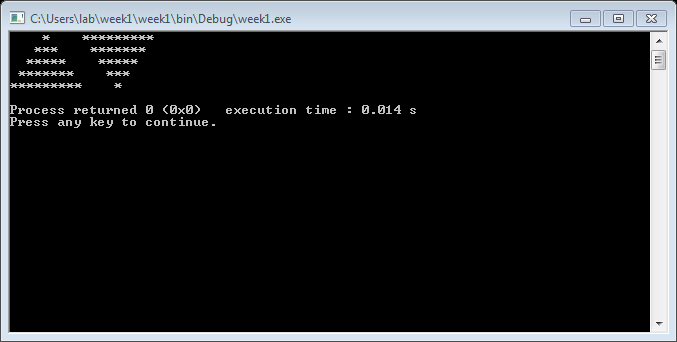
cout << " \*\*\*\*\*\*\* \*\*\*" << endl;

cout << "\*\*\*\*\*\*\*\*\* \*" << endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

using namespace std;

int main()

{

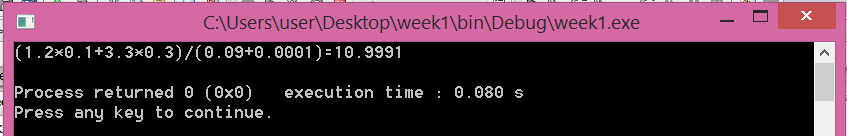
cout << "(1.2\*0.1+3.3\*0.3)/(0.09+0.0001)=";

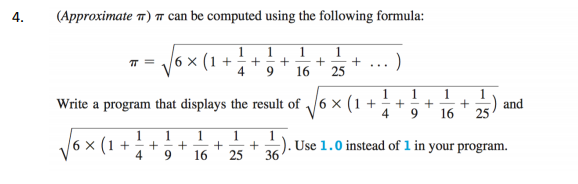
cout <<(1.2\*0.1)+(3.3\*0.3)/(0.09+0.001) << endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

#include <cstdlib>

#include <math.h>

using namespace std;

int main()

{

double n;

double r;

n=sqrt (6.0\*(1.0+1.0/4.0+1.0/9.0+1.0/16.0+1.0/25.0));

r=sqrt (6.0\*(1.0+1.0/4.0+1.0/9.0+1.0/16.0+1.0/25.0+1.0/36.0));

cout <<"sqrt (6.(1+1/4+1/9+1/16+1/25))=";

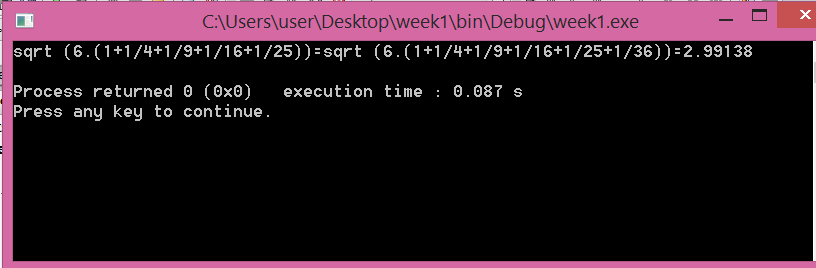
cout <<"sqrt (6.(1+1/4+1/9+1/16+1/25+1/36))=";

cout <<r<<endl;

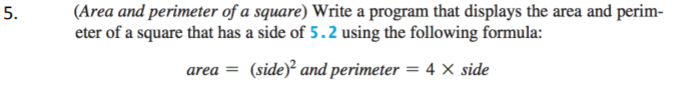
return 0;

}

**OUTPUT**



**CODE**



#include <cmath>

#include<conio.h>

using namespace std;

int main()

{

int side, peri, area;

cout <<"Enter lenght of side of square:";

cin >>side;

area =side\*side;

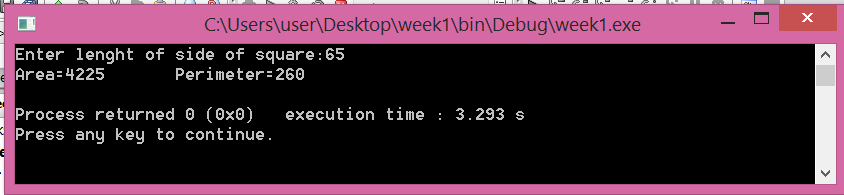
peri=4\*side;

cout<<"Area="<<area<<"\tPerimeter="<<peri<<endl;

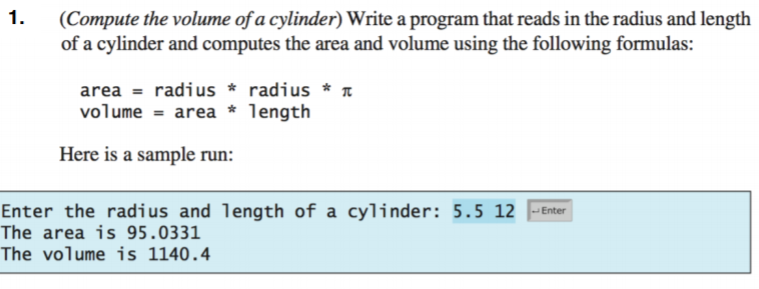
return 0;

}

**OUTPUT**



**Week 2: Lab Exercises**



**CODE**

#include <iostream>

#include <cstdlib>

#include <cmath>

#include<conio.h>

using namespace std;

int main()

{

double radius;

double lenght;

double area;

double volume;

cout <<"Enter a radius:";

cin>> radius;

cout <<"Enter a lenght:";

cin>>lenght;

area=radius\*radius\*3.14159;

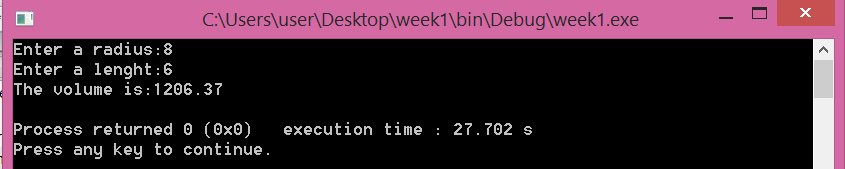
volume=(double)area\*lenght;

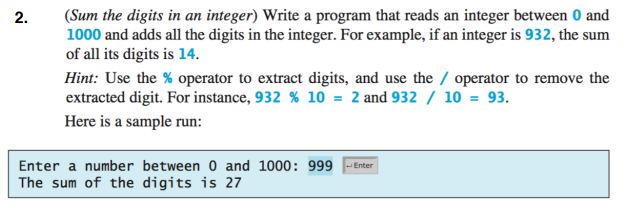
cout<<"The volume is:"<<volume<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

using namespace std;

int main()

{

int num;

cout << "Enter a number:";

cin >> num;

int sum = 0;

int a;

while (num != 0)

{

a = num % 10;

num = num / 10;

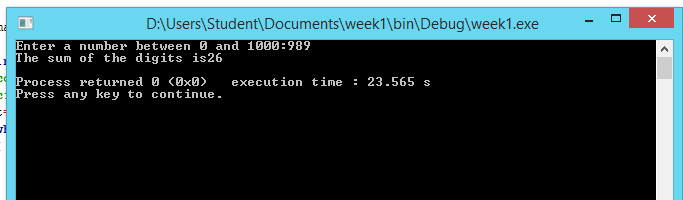
sum = sum + a;

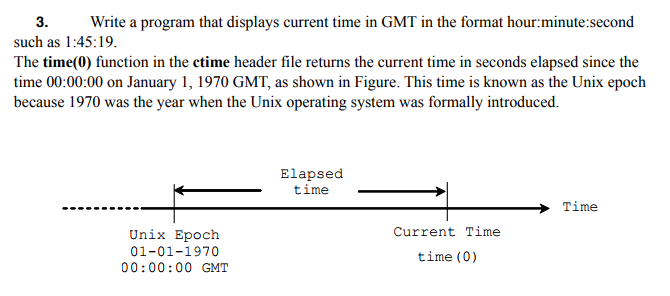
}

cout << "The sum of digits is " << sum << endl;

}

**OUTPUT**





**CODE**

#include <iostream>

#include<ctime>

using namespace std;

int main()

{

int totalSeconds = time (0);

int currentSecond = totalSeconds%60;

int totalMinutes = totalSeconds/60;

int currentMinute = totalMinutes%60;

int totalHours = totalMinutes/60;

int currentHour = totalHours%24;

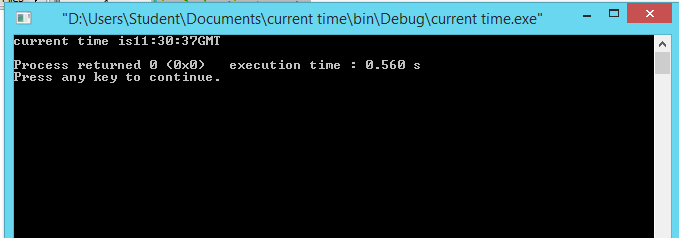
cout << "current time is" <<currentHour<<":"

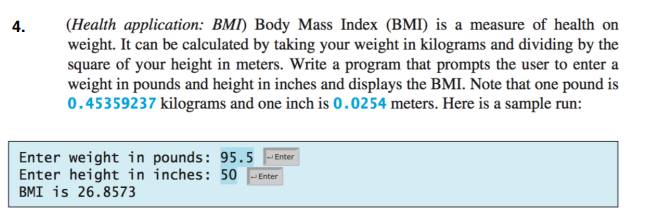
<<currentMinute<< ":"<<currentSecond << "GMT"<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

#include<ctime>

#include<stdlib.h>

using namespace std;

int main()

{

double pound, inch, kilo, meter, BMI;

cout <<"Enter weight in pounds";

cin>> pound;

cout<<"Enter height in inches";

cin>> inch;

kilo= pound\*0454;

meter= inch\*0.254;

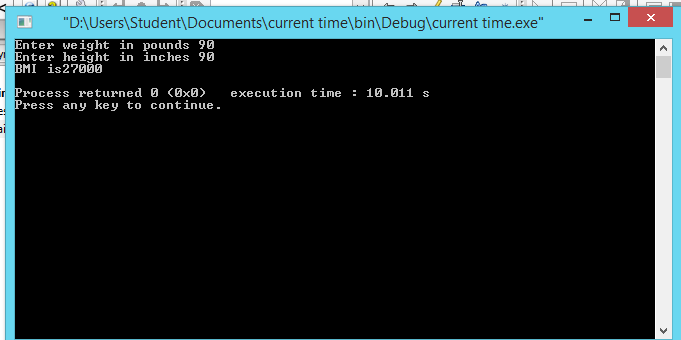
BMI =kilo/meter\*meter;

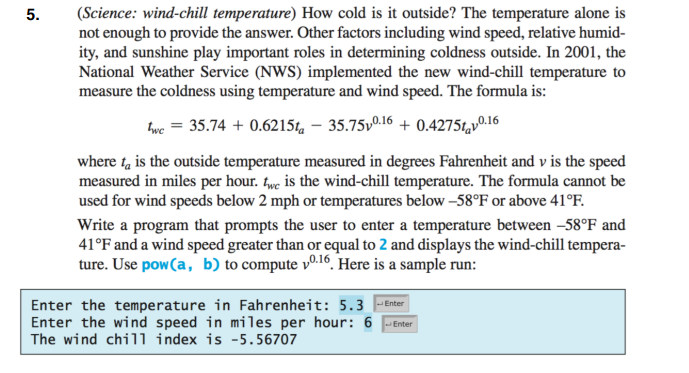
cout <<"BMI is" <<BMI<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

using namespace std;

int main()

{

double ta, v, twc;

cout << "Enter the temprature in farhenheit:";

cin >> ta;

cout << "Enter the velocity of the wind:";

cin >> v;

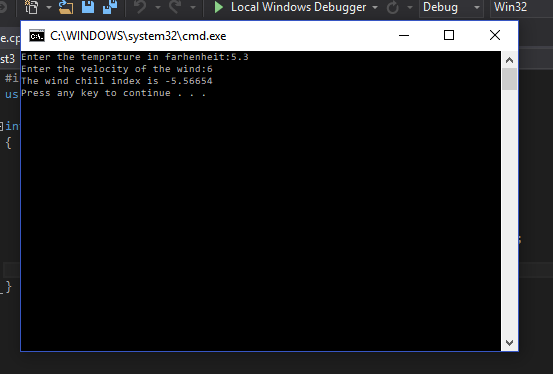
twc = 35.74 + 0.6216\*ta - 35.75\*pow(v, 0.16) + 0.4275\*ta\*pow(v, 0.16);

cout << "The wind chill index is " << twc << endl;

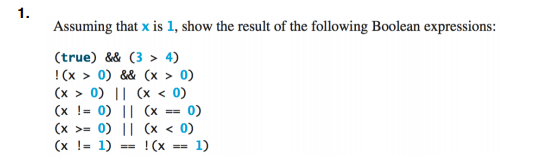
return 0;

}

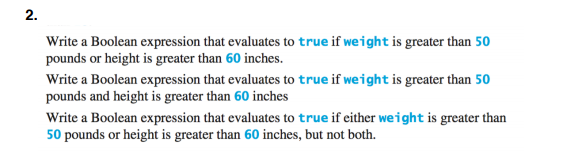
**OUTPUT**

****

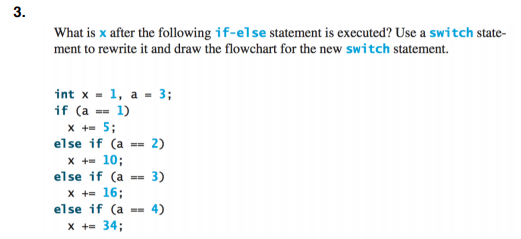
**Week 3 Lab exercises**



1. (true) && (3>4) ……………False. 3 is not greater than 4
2. !(x>0) && (x>0)…………….False.
3. (x!=0) || (x==0)……………….True
4. (x>0) || (x<0)…………………True
5. (X!=1) == !(x==1)…………….False



1. Weight >50 || Height >60
2. Weight >50 || Height >60
3. Weight >50 ^ Height >60



Switch (a) {

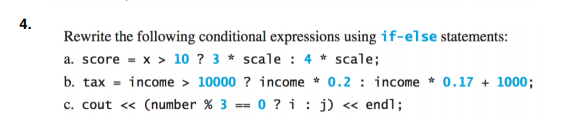
Case 1: x+5 = 5; break;

2: x+=10; break;

3: x+=16; break;

4: x+=34; break;}

X is 17



1. If (x>10

Score = 3\*scale;

else

score =4\*scale;

1. If (income>1000)

Tax=income\*0.2;

Else

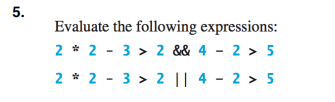
Tax=income \*0.17+1000;

1. If (number %3==)

Cout <<i<<endl;

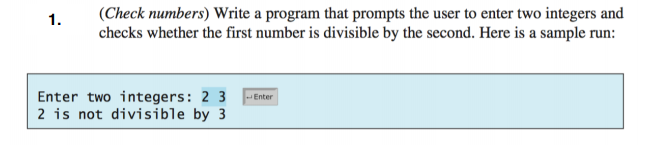
Else

Cout <<j<< endl;



1. False
2. False

**Programming Exercises**



**CODE**

#include <iostream>

using namespace std;

int main()

{

int a;

int b;

cout<<"Enter a value a"<<endl;

cin>> a;

cout<<"Enter a value b"<<endl;

cin>>b;

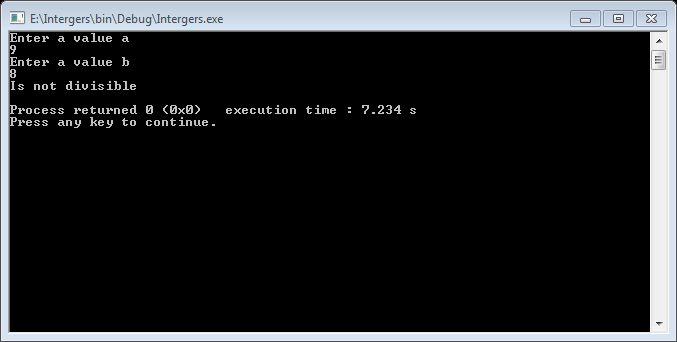
if (a%b==0) cout<<"Is Divisible"<<endl;

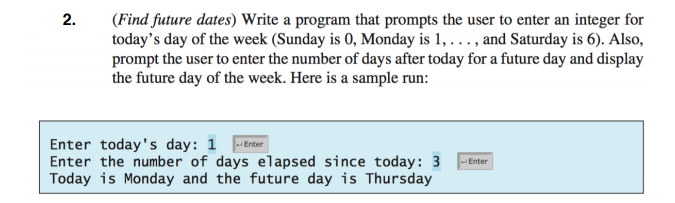
else cout<<"Is not divisible"<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

using namespace std;

int main()

{

int today = 0;

int futureDay = 0;

int daysElapsed = 0;

int sunday = 0,

monday = 1,

tuesday = 2,

wednesday = 3,

thursday = 4,

friday = 5,

saturday = 6;

cout << "Enter an integer value for day of week. \n"

<< "Week starting with Sunday as 0th day. \n";

cout << "Enter today's day: ";

cin >> today;

cout << "Enter the number of days elapsed since today: ";

cin >> daysElapsed;

futureDay = today + daysElapsed;

switch (today)

{

case 0 : cout << "Today is Sunday and the future day is ";

break;

case 1 : cout << "Today is Monday and the future day is ";

break;

case 2 : cout << "Today is Tuesday and the future day is ";

break;

case 3 : cout << "Today is Wednesday and the future day is ";

break;

case 4 : cout << "Today is Thursday and the future day is ";

break;

case 5 : cout << "Today is Friday and the future day is ";

break;

case 6 : cout << "Today is Saturday and the future day is ";

break;

}

switch (futureDay%7)

{

case 0 : cout << "Sunday";

break;

case 1 : cout << "Monday";

break;

case 2 : cout << "Tuesday";

break;

case 3 : cout << "Wednesday";

break;

case 4 : cout << "Thursday";

break;

case 5 : cout << "Friday";

break;

case 6 : cout << "Saturday";

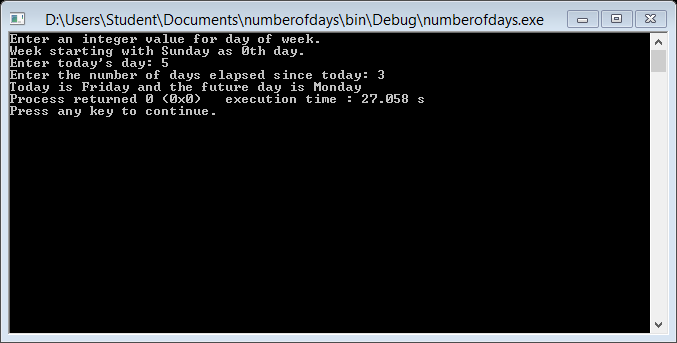
break;

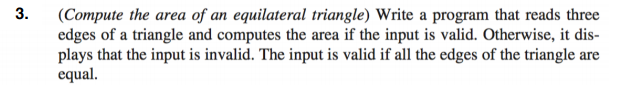
}

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

#include <cmath>

#include <ctime>

#include<cstdlib>

using namespace std;

int main()

{

float aside,bside,cside;

cout<<"Enter lenght of side a";

cin>>aside;

cout<<"Enter lenght of side b";

cin>>bside;

cout<<"Enter lenght of side c";

cin>>cside;

float three=3;

float area =sqrt(three)/4\*(aside\*aside);

if (aside==bside&&bside==cside)

{

cout<<"The area of the equiletaral triangle:"<<area<<endl;

}

else

{

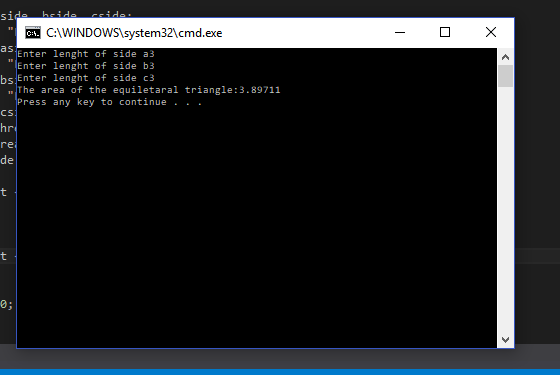
cout <<"Not valid:"<<endl;

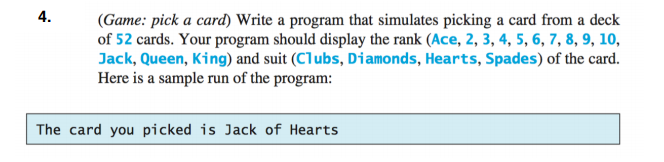
}

return 0;

}

**OUTPUT**

****



**CODE**

#include <iostream>

#include <ctime>

#include <cmath>

#include <cstdlib>

#include <string>

using namespace std;

const string values[13] = {"Ace", "2", "3", "4", "5", "6", "7", "8", "9", "10",

"Jack", "Queen", "King"};

const string suits[4] = {"Spades", "Hearts", "Clubs", "Diamonds"};

string StringRepresentation(unsigned int card)

{

if(card >= 52)

{

return "Invalid";

}

return values[card % 13] + " of " + suits[card/13]; // where card ranges from 0 to 51.

}

int main()

{

// Deals a random card

srand(time(NULL));

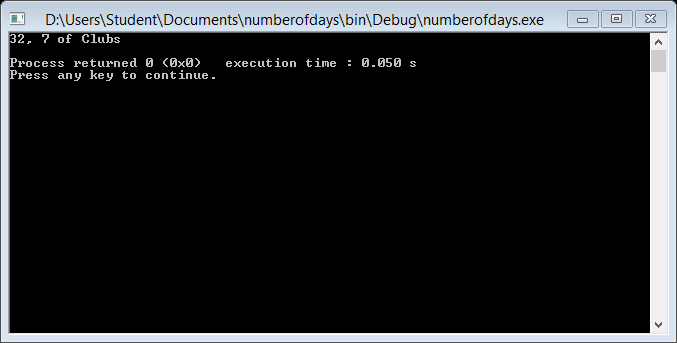
int card = rand() % 52;

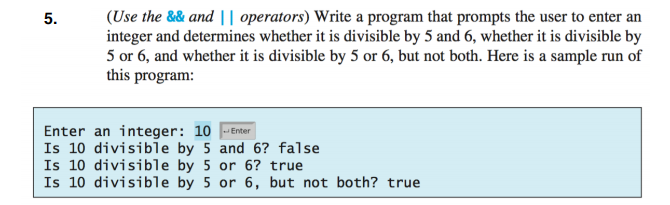
cout << card << ", " << StringRepresentation(card) << endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

#include<stdlib.h>

using namespace std;

int main()

{

int number;

cout << "Enter an integer:";

cin >> number;

if (number % 5 == 0 && number % 6 == 0)

cout << " Is " << number << " Divisible by 5 and 6? True" << endl;

else

cout << " Is " << number << " divisible by 5 and 6 ? False" << endl;

if (number % 5 == 0 || number % 6 == 0)

cout << " Is " << number << " divisible by 5 or 6? True" << endl;

else

cout << "is" << number << " divisible by 5 or 6? False" << endl;

if ((number % 5 == 0 || number % 6 == 0) && !(number % 5 == 0 && number % 6 == 0))

cout << "is" << number << " divisible by 5 and 6, but not both?True" << endl;

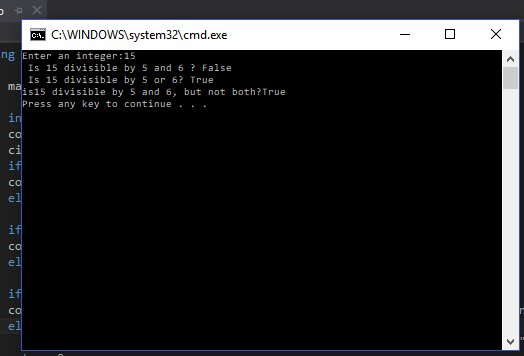
else

cout << " is " << number << " divisible by 5and 6, but not both?False" << endl;

return 0;

}

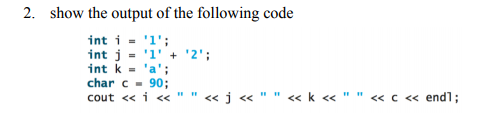
**OUTPUT**

****

**Week 4 Lab exercises**



**“double radians=math.toRadians(47)”**



**CODE**

#include <iostream>

#include<stdlib.h>

using namespace std;

int main()

{

int i ='1';

int j='1';

int k='a';

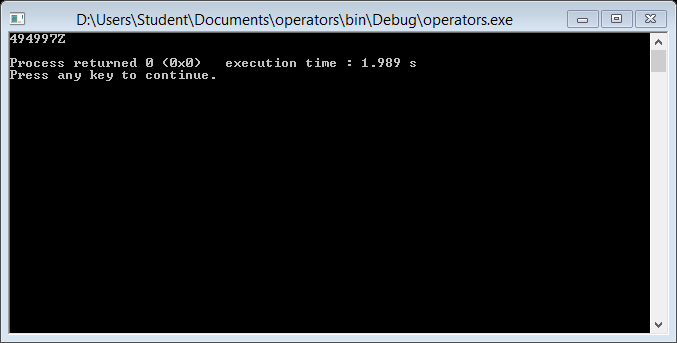
char c=90;

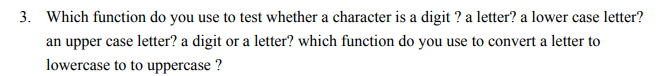
cout <<i<<""<<j<<""<<k<<""<<c<<endl;

return 0;

}

**OUTPUT**





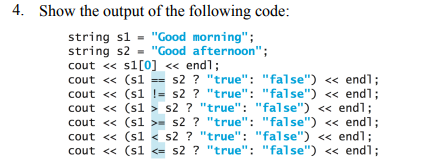
To check whether a character is a digit : isdigit(char)

To check whether a character is a letter : isalpha(char)

To check whether a character is a lowercase letter : islower(char)

To check whether a character is a uppercase letter : isupper(char)

To convert lowercase to uppercase : toupper(char)



False

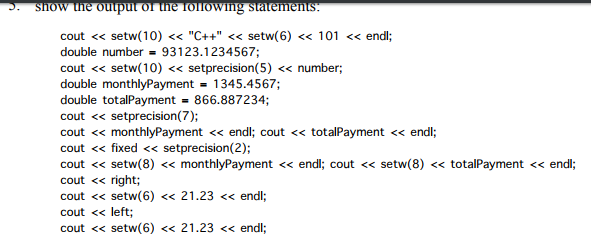
True

True

True

False

False



C++ 101

93123.123

866.8872

1345.46

866.89

21.23

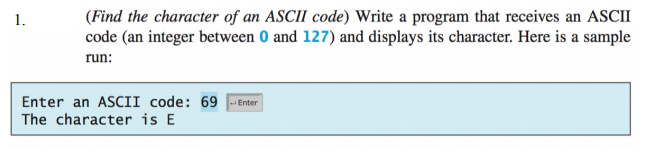
21.23



For reading : ifstream File(“test.txt”)

For writing : ofstream File(“test.txt”)

**Programming Exercises**



**CODE**

#include <iostream>

#include<stdlib.h>

using namespace std;

int main()

{

int n;

cout <<"Enter an ASCII code:";

cin>>n;

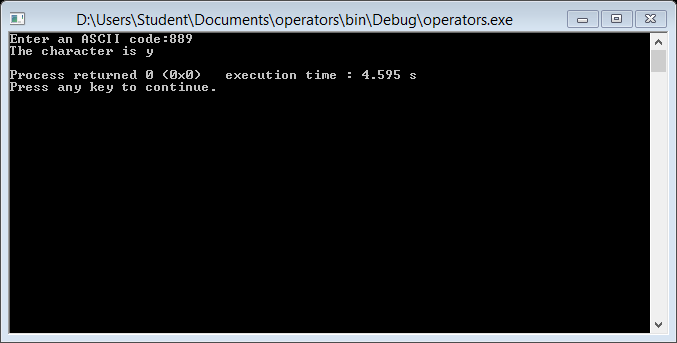
char ch(n);

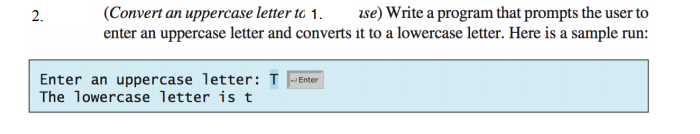
cout <<"The character is "<<ch<<endl;

return 0;

}

**OUTPUT**





**CODE**

#include <iostream>

#include<stdlib.h>

using namespace std;

int main()

{

char lowercaseletter;

cout << "Enter a lowercase letter:";

cin >> lowercaseletter;

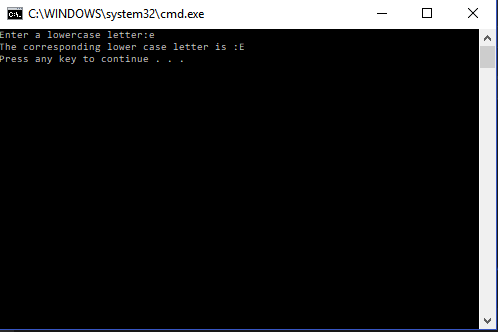
char uppercaseletter = toupper(lowercaseletter);

cout << "The corresponding lower case letter is :" << uppercaseletter << endl;

return 0;

}

**OUTPUT**

****



#include <iostream>

#include <string>

#include <cstdlib>

#include <ctime>

using namespace std;

int main()

{

char alpha[] =

"ABCDEFGHIJKLMNOPQRSTUVWXYZ"

"abcdefghijklmnopqrstuvwxyz";

int stringLength = sizeof(alpha) - 1;

char random[6];

srand(time(0));

for (int i = 0; i < 6; i++)

{

random[i] = alpha[rand() % stringLength];

}

for (int i = 0; i < 6; i++)

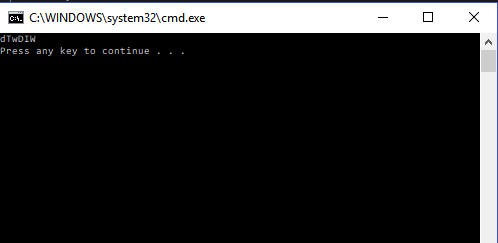
cout << random[i];

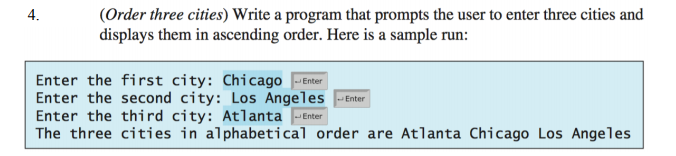
cout << endl;

return 0;

}

**OUTPUT**





#include <iostream>

#include<stdlib.h>

#include <string>

using namespace std;

int main()

{

string city1, city2, city3;

cout << "Enter the name of first city:";

cin >> city1;

cout << "Enter the name of second city:";

cin >> city2;

cout << "Enter the name of third city:";

cin >> city3;

if (city1[0] < city2[0] && city2[0] < city3[0])

cout << city1 << " " << city2 << " " << city3 << endl;

else if (city2[0] < city1[0] && city1[0] < city3[0])

cout << city2 << " " << city1 << " " << city3 << endl;

else if (city1[0] < city2[0] && city3[0] < city2[0])

cout << city1 << " " << city3 << " " << city2 << endl;

else if (city2[0] < city1[0] && city3[0] < city1[0])

cout << city2 << " " << city3 << " " << city1 << endl;

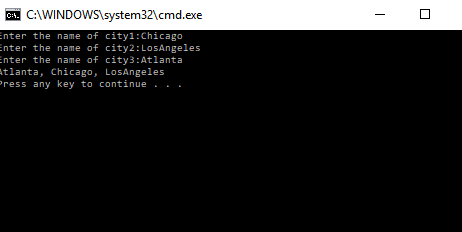
else if (city3[0] < city2[0] && city2[0] < city1[0])

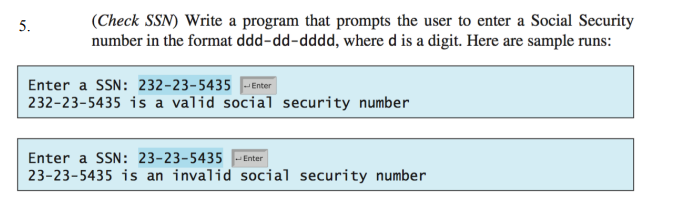
cout << city3 << " " << city2 << " " << city1 << endl;

else if (city3[0] < city2[0] && city1[0] < city2[0])

cout << city3 << " " << city1 << " " << city2 << endl;

}





#include <iostream>

#include<stdlib.h>

#include <string>

using namespace std;

int main()

{

string ssn;

cout << "Enter an ssn number:";

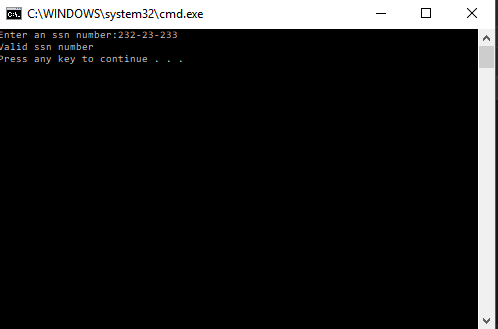
cin >> ssn;

if (ssn[3] == '-' && ssn[6] == '-')

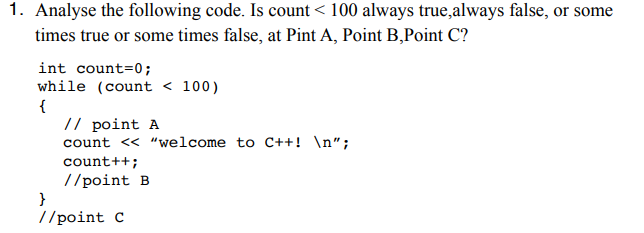
cout << "Valid ssn number" << endl;

else

cout << "Invalid ssn number" << endl;

}

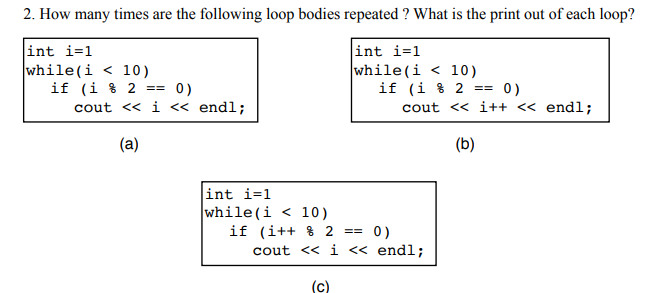
Screen Clipping



Point A : Always true

Point B : Sometimes true or sometimes false

Point C : Always false



**Part a:**

Infinite

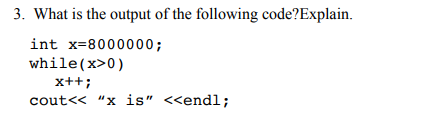
**Part b:**

Infinite

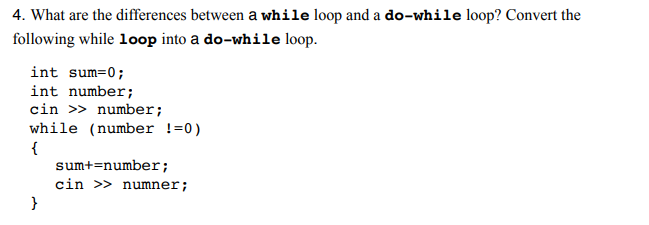
**Part c:**

9 times.

3, 5, 7, 9



Infinite loop.



int sum = 0;

int number;

cin >> number;

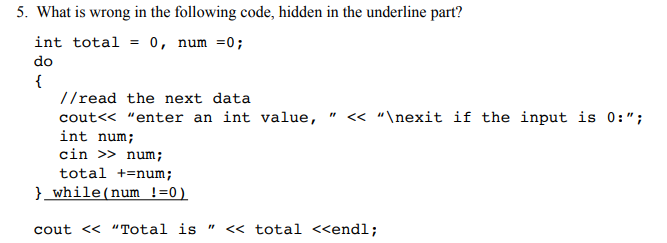
do

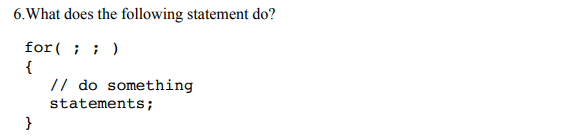
{

sum += number;

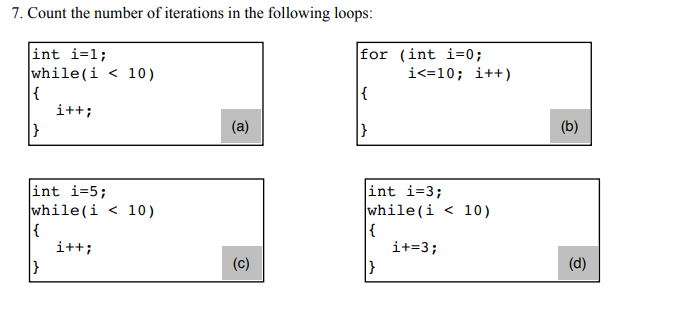
cin >> number;

}while (number != 0)





Infinite loop



**Part a:**

9 times.

**Part b:**

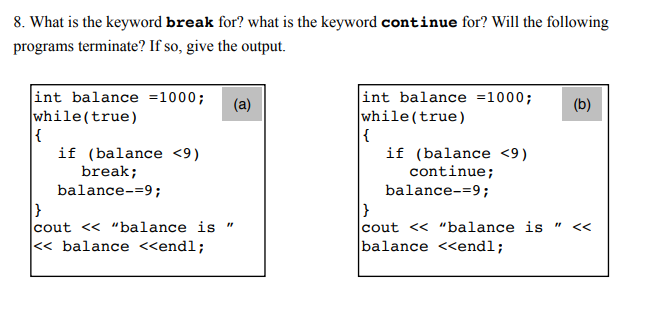
11 times.

**Part c:**

5 times.

**Part d:**

3 times.



Break condition is for exiting a loop.

Continue statement is for skipping a particular iteration of a loop.

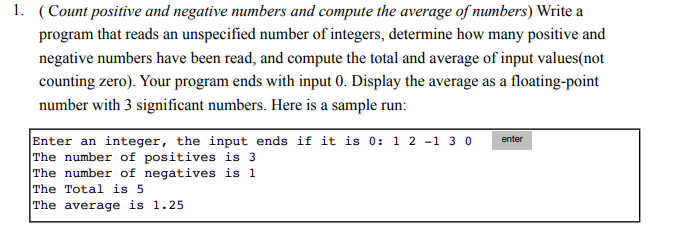
Part a:

Yes, program will terminate when the balance becomes less than 9.

Part b:

No, the loop is infinite. It will skip only the loop when the balance becomes less than 9.

Screen Clipping



**#include** <iostream>

**#include** <iomanip>

**using** **namespace** std;

**int** main() {

**int** sum = 0;

**int** count = 0;

**int** num = 0;

**float** avg = 0;

**int** positive = 0;

**int** negative = 0;

**while** (**true**)

{

cout << "Enter a number:";

cin >> num;

**if** (num == 0)

**break**;

**else** {

count++;

sum += num;

**if** (num > 0)

positive++;

**else**

negative++;

}

}

cout << "The sum of the numbers is " << sum << endl;

avg = **float**(sum)/count;

cout << setprecision(4);

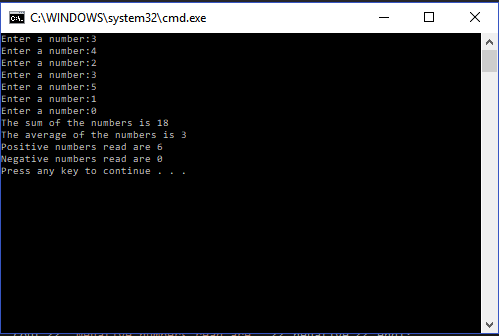
cout << "The average of the numbers is " << avg << endl;

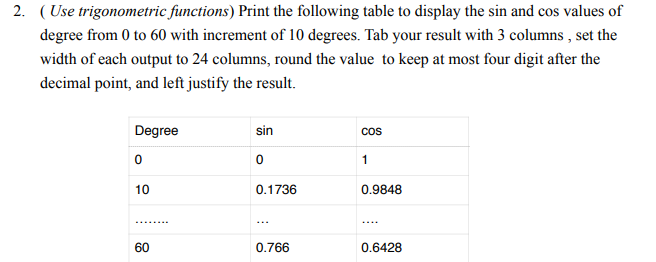
cout << "Positive numbers read are " << positive << endl;

cout << "Negative numbers read are " << negative << endl;

**return** 0;

}





#include <iostream>

#include <string>

#include <iomanip>

using namespace std;

int main()

{

cout << setw(24) << "Degree";

cout << setw(24) << "Sin";

cout << setw(24) << "Cos";

cout << endl;

int degree = 0;

while (degree <= 60)

{

cout << setw(24) << degree;

float s = sin(degree);

float c = cos(degree);

cout << setw(24) << setprecision(4) << s;

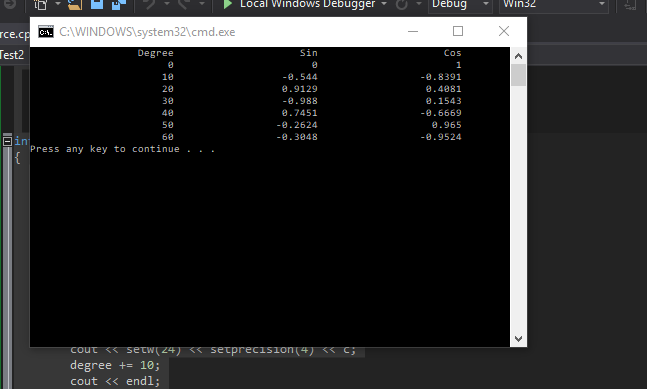
cout << setw(24) << setprecision(4) << c;

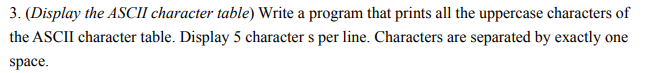
degree += 10;

cout << endl;

}

}





**#include** <iostream>

**using** **namespace** std;

**int** main() {

**char** ch = 65;

**int** count = 1;

**while** (ch <= 90)

{

cout << ch << " ";

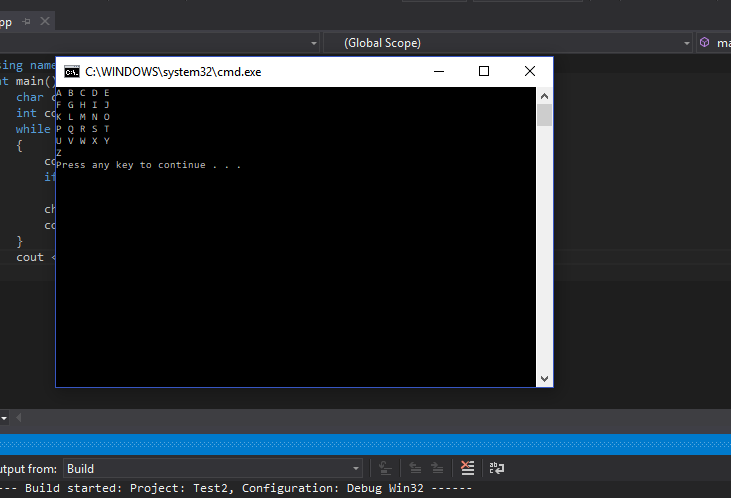
**if** (count % 5 == 0)

cout << endl;

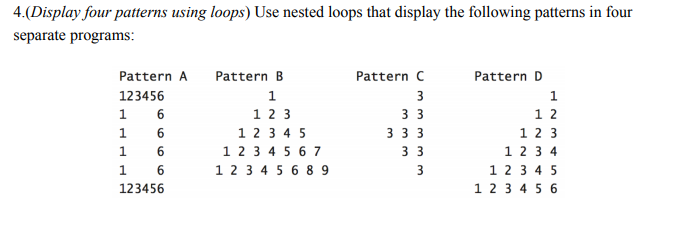
ch++;

count++;

}



}



**Part a:**

#include <iostream>

using namespace std;

int main() {

int num = 1;

while (num <= 6)

{

cout << num;

num++;

}

cout << endl;

for (int i=0; i<4; i++)

{

cout << 1;

for (int j=0; j<4; j++)

cout << " ";

cout << 6;

cout << endl;

}

num = 1;

while (num <= 6)

{

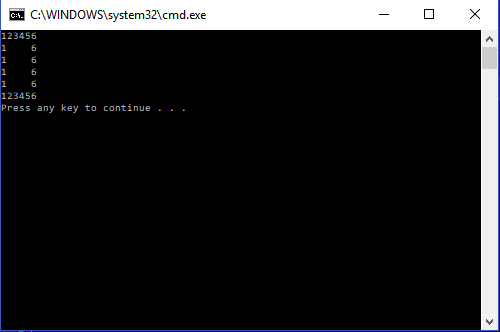
cout << num;

num++;

}

cout << endl;

}



**Part B:**

**#include** <iostream>

**using** **namespace** std;

**int** main() {

**int** spaces = 7;

**for** (**int** i=1; i<10; i+=2)

{

**for** (**int** k=0; k<spaces; k++)

{

cout << " ";

}

**for** (**int** j=1; j<=i; j++)

{

cout << j << " ";

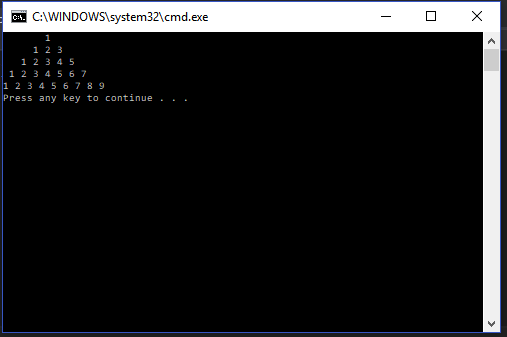
}

spaces -= 2;

cout << endl;

}

}



**Part c:**

**#include** <iostream>

**using** **namespace** std;

**int** main() {

**int** spaces = 4;

**for** (**int** i=1; i<=3; i++)

{

**for** (**int** j=0; j<spaces; j++)

{

cout << " ";

}

**for** (**int** k=1; k<=i; k++)

cout << 3 << " ";

spaces -= 2;

cout << endl;

}

spaces += 4;

**for** (**int** i=2; i>=1; i--)

{

**for** (**int** j=0; j<spaces; j++)

cout << " ";

**for** (**int** k=1; k<=i; k++)

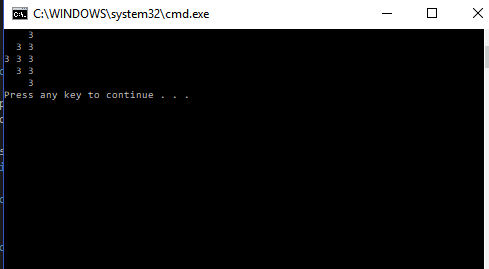
cout << 3 << " ";

cout << endl;

spaces += 2;

}

}



**Part d:**

**#include** <iostream>

**using** **namespace** std;

**int** main() {

**int** spaces = 10;

**int** times = 1;

**for** (**int** i=0; i<6; i++)

{

**int** num = 1;

**for** (**int** j=0; j<spaces; j++)

{

cout << " ";

}

**for** (**int** k=0; k<times; k++)

{

cout << num << " ";

num++;

}

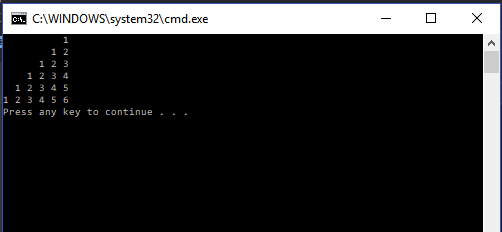
cout << endl;

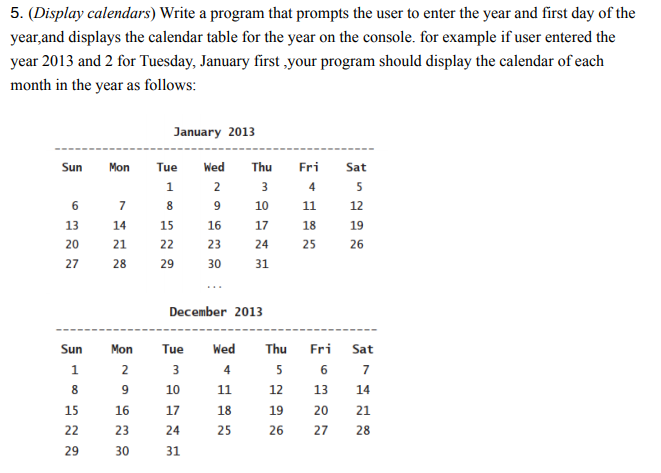
spaces -= 2;

times += 1;

}

}





#include <iostream>

#include <iomanip>

using namespace std;

/\*

A Function that returns the name of the month

with a given month number \*/

string getMonthName(int monthNumber)

{

string months[] = { "January", "February", "March",

"April", "May", "June",

"July", "August", "September",

"October", "November", "December"

};

return (months[monthNumber]);

}

/\* A Function to return the number of days in

a month\*/

int numberOfDays(int monthNumber, int year)

{

// January

if (monthNumber == 0 || monthNumber == 2 || monthNumber == 4 || monthNumber == 6 || monthNumber == 7 || monthNumber == 9 || monthNumber == 11)

return 31;

// February

if (monthNumber == 1)

{

// If the year is leap then February has

// 29 days

if (year % 400 == 0 ||

(year % 4 == 0 && year % 100 != 0))

return 29;

else

return 28;

}

else

return 30;

}

// Function to print the calendar of the given year

void printCalendar(int year, int dayNumber)

{

cout << " Calendar - " << year << "\n\n";

int days;

// Index of the day from 0 to 6

int current = dayNumber;

// i --> Iterate through all the months

// j --> Iterate through all the days of the

// month - i

for (int i = 0; i < 12; i++)

{

days = numberOfDays(i, year);

// Print the current month name

cout << "\n ------------" << getMonthName(i).c\_str() << "-------------\n";

// Print the columns

cout << " Sun Mon Tue Wed Thu Fri Sat\n";

// Print appropriate spaces

int k;

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

cout << " ";

for (int j = 1; j <= days; j++)

{

cout << setw(5) << j;

if (++k > 6)

{

k = 0;

cout << "\n";

}

}

if (k)

cout << "\n";

current = k;

}

return;

}

// Driver Program to check above funtions

int main()

{

int year, dayNumber;

cout << "Enter a year:";

cin >> year;

cout << "Enter the first date of that year:";

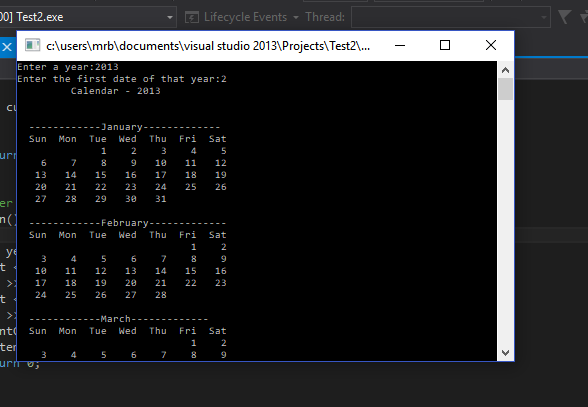
cin >> dayNumber;

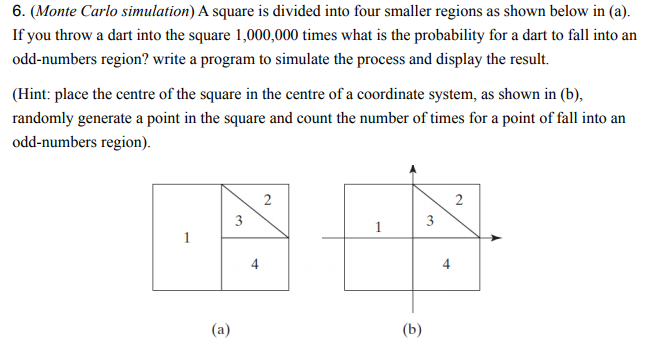
printCalendar(year, dayNumber);

system("pause");

return 0;

}





#include<iostream>

#include<ctime>

#include<cmath>

#include<iomanip>

#include<cstdlib>

using namespace std;

int main()

{

int nDarts;

cout << "\nHow many darts do you want to throw?: " << endl;

cin >> nDarts;

double prob;

if (nDarts > 0)

{

//Sets seed value for the random numbers

srand(time(0));

int hit = 0;

for (int i = 0; i < nDarts; ++i)

{

//generates a random value of x and

double xValue = static\_cast<double>(rand()) / RAND\_MAX;

double yValue = static\_cast<double>(rand()) / RAND\_MAX;

if (((xValue < 1) && (yValue > 1)) || ((xValue<1) && yValue < 1) || ((xValue>1) && (yValue>1)))

++hit;

}

prob = (double(hit) / nDarts) \* 100;

cout << "\nThere were " << hit << " hits in the odd numbered regions" << endl;

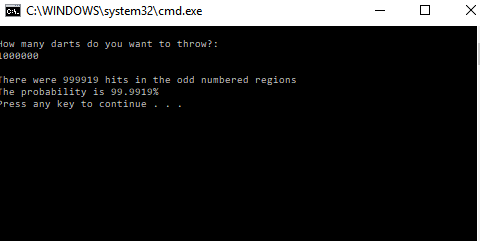
cout << "The probability is " << prob << "%" << endl;

}

return 0;

}

**OUTPUT**

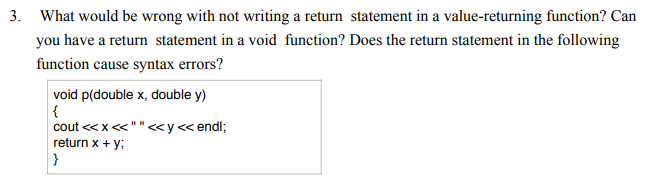


Screen Clipping

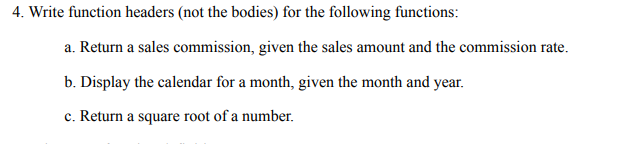
Screen Clipping

A function is a group of statements that together perform a task. We invoke a function by its name and entering the parameters.

Screen Clipping



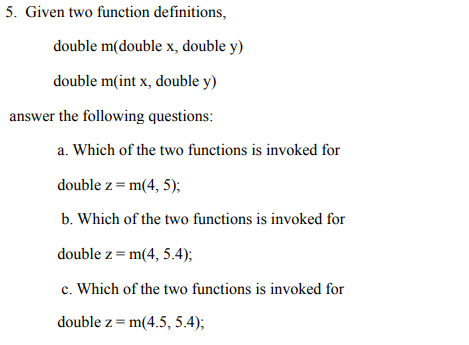
Yes, this statement will cause a syntax error.



double getSalesComission(int salesAmount, double commissionRate);

void DisplayCelendar(int month, int year);

float SquareRoot(float num);



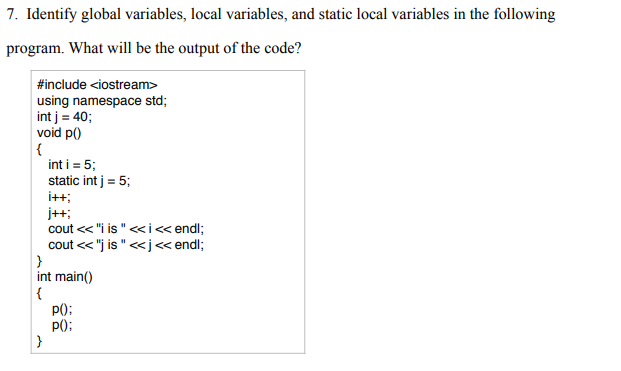
a. Second Function.

b. First Function

c. First Funtion.

Screen Clipping

The inline functions are a C++ enhancement feature to increase the execution time of a program. Functions can be instructed to compiler to make them inline so that compiler can replace those function definition wherever those are being called.



J is a global variable.

I is a local variable.

J is a local static variable.

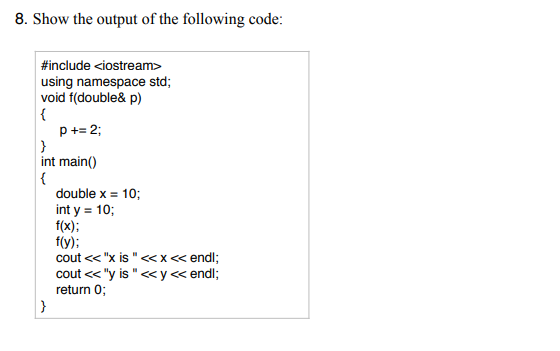
**Output:**

I is 6

J is 6

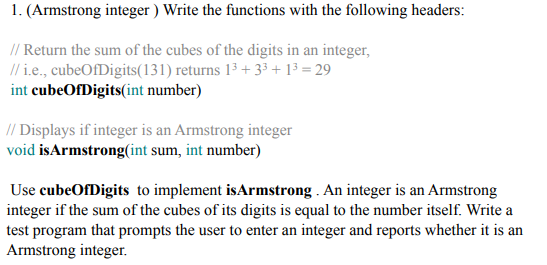
I is 6

J is 7



ERROR

Screen Clipping



#include <iostream>

using namespace std;

int cubeOfDigits(int num) {

int sum = 0;

while (num != 0)

{

int a = num % 10;

num = num / 10;

a = a\*a\*a;

sum += a;

}

return sum;

}

void Armstrong(int sum, int num)

{

if (cubeOfDigits(num) == num)

cout << num << " is an armstrong number" << endl;

else

cout << num << " is not an armstrong number" << endl;

}

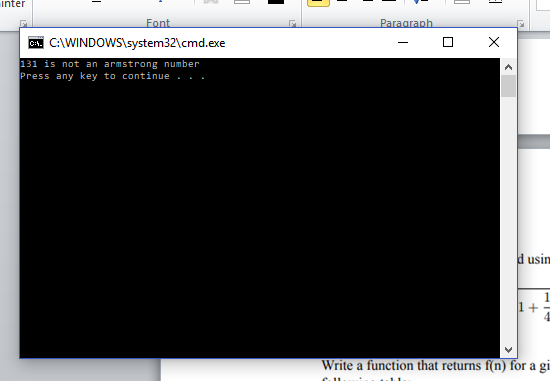
int main()

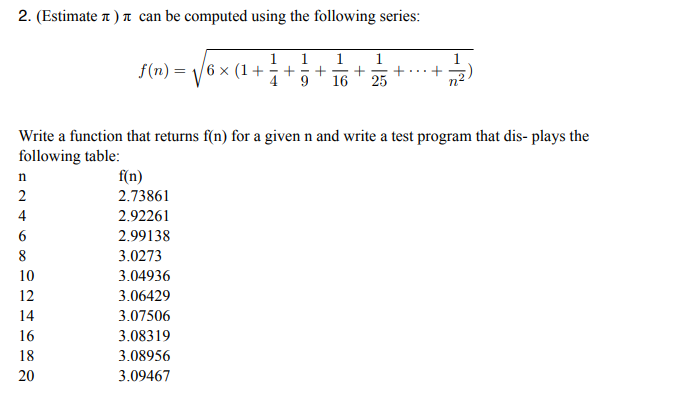
{

Armstrong(0, 371);

return 0;

}





#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

int n;

//cout << "Enter a number:";

//cin >> n;

cout << setw(10) << "n";

cout << setw(10) << "f(n)";

cout << endl;

for (int n = 0; n <= 20; n += 2)

{

float sum = 0;

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

{

float m = float(1) / (i\*i);

sum += m;

}

sum = sum \* 6;

sum = sqrt(sum);

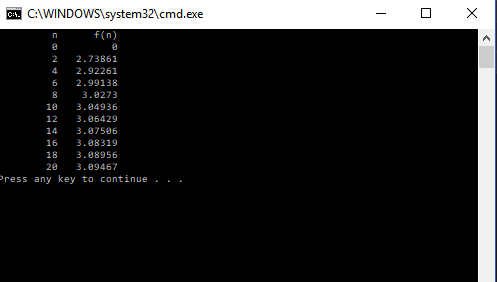
cout << setw(10) << n;

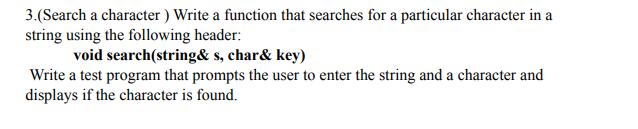
cout << setw(10) << sum;

cout << endl;

}

return 0;

}



#include <iostream>

#include <string>

using namespace std;

void Search(string &s, char &key)

{

bool flag = false;

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

{

if (s[i] == key)

flag = true;

}

if (flag)

cout << "Character found in the array" << endl;

else

cout << "Character not found in the array" << endl;

}

int main()

{

string str;

char key;

cout << "Enter a string:";

cin >> str;

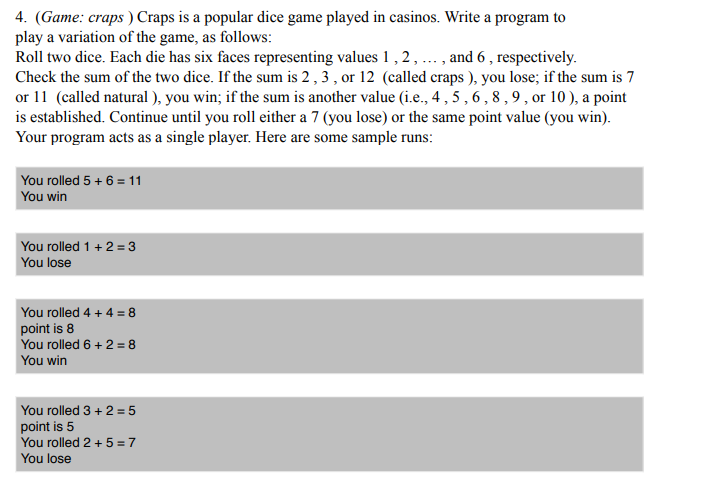
cout << "Enter the character you want to search in the array:";

cin >> key;

Search(str, key);

return 0;

}



#include <iostream>

#include <cstdlib>

#include <ctime>

using namespace std;

int main()

{

int die1;

int die2;

srand(time(NULL));

die1 = rand() % 6 + 1;

die2 = rand() % 6 + 1;

cout << "You rolled " << die1 << " + " << die2 << " = " << die1 + die2 << endl;

if (die1 + die2 == 4 || die1 + die2 == 5 || die1 + die2 == 6 || die1 + die2 == 8 || die1 + die2 == 9 || die1 + die2 == 10)

{

cout << "Point is " << die1 + die2 << endl;

int point = die1 + die2;

while (true)

{

die1 = rand() % 6 + 1;

die2 = rand() % 6 + 1;

cout << "You rolled " << die1 << " + " << die2 << " = " << die1 + die2 << endl;

if (die1 + die2 == point)

{

cout << "YOU WON!!" << endl;

break;

}

if (die1 + die2 == 7)

{

cout << "YOU LOSE!!" << endl;

break;

}

}

}

else if (die1 + die2 == 7 || die1 + die2 == 11)

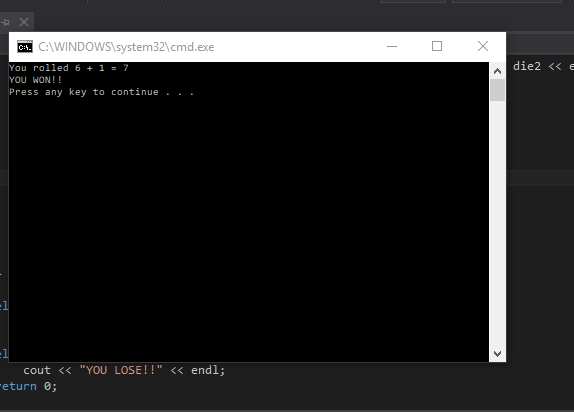
cout << "YOU WON!!" << endl;

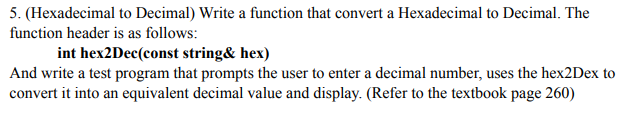
else

cout << "YOU LOSE!!" << endl;

return 0;

}





#include <iostream>

#include <string>

#include <stdlib.h>

using namespace std;

int hex2dec(const string &hex)

{

int sum = 0;

int power = hex.length() - 1;

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

{

if (hex[i] == 'A')

sum = sum + (10 \* pow(16, power));

else if (hex[i] == 'B')

sum = sum + (11 \* pow(16, power));

else if (hex[i] == 'C')

sum = sum + (12 \* pow(16, power));

else if (hex[i] == 'D')

sum = sum + (13 \* pow(16, power));

else if (hex[i] == 'E')

sum = sum + (14 \* pow(16, power));

else if (hex[i] == 'F')

sum = sum + (15 \* pow(16, power));

else

{

int num = hex[i] - 48;

sum = sum + (num \* pow(16, power));

}

power--;

}

return sum;

}

int main()

{

string hex;

cout << "Enter a hexadecimal number:";

cin >> hex;

cout << hex2dec(hex) << endl;

return 0;

}

