

- 10
 2. Write a C++ program that calculates a user's pay. The program reads the number of hours that the user worked and the amount paid per hour from `stdin`, and then computes the total pay for the user and displays it on `stdout`.

```
#include <iostream>
using namespace std;

int initialize (int hours, int pay)
{
    int compare (int hours, int pay)
    {
        int resultpay;
        resultpay = hours * pay;
        return resultpay;
    }

    int main()
    {
        int hours, pay, totalpay;
        cout << "Enter the number of hours worked" << endl;
        cin >> hours;
        cout << "Enter the pay for each hour" << endl;
        cin >> pay;
        totalpay = compare (hours, pay);
        cout << "total pay for student is" << totalpay << endl;
    }
}
```

* Testcase *

Enter the number of hours worked

10

Enter the pay for each hour

2

total pay for student is 20

2

21807314

NAME: VIJAY MURUGENDRAM
Midterm Exam
Book and Notes

CS

// ask the

full definition:

template <class T> class list

{

private:

Node<T> * head;

public:

void clear (Node<T>*);

~list (clear (head));

}

Template <class T> class node

{

friend class list<T>;

private:

T data;

Node<T> * next;

public:

Node (const T& = T(), Node<T>* = nullptr);

}

NAME: VIJAY MURUGAPPAN SUBBIAH (21807314).

CSCI 689-1

Quiz 2
Closed Book and Notes
(10 points)

Fall 2016

1. Circle the best answer for each following C++ segments:

a) if (60 <= 12 + 5)
cout << "Hello";
cout << " There ";

outputs the following:

(i) Hello There (ii) Hello

(iii) Hello There

(iv) There

- |

b) if ('a' > 'b' || 'B' > 'A')
cout << "#*#" << endl;

outputs the following:

(i) #*#

(ii) # *

(iii) *

(iv) none of these

Ans. → (i)

#

c) if (7 <= 7)
cout << 6 - 9 * 2 / 6 << endl;

outputs the following:

(i) -1

(ii) 3

(iii) 3.0

(iv) none of these

d) if (7 < 8) {
cout << "2 4 6 8" << endl;
cout << "1 3 5 7" << endl;
}

outputs the following:

(i) 2 4 6 8 (ii) 1 3 5 7 (iii) none of these

1 3 5 7

e) if (5 < 3)
cout << "*";
else if (7 == 8)
cout << "&";
else
cout << "\$";

outputs the following:

(i) * (ii) & (iii) \$ (iv) none of these

(2) Assume the variables $x = 5$, $y = 6$, and $z = 8$. Indicate by circling the T or F whether each of the following conditions is true or false:

- | | | |
|-----------------------------------|------------------------------------|-------------------------|
| a) $x == 5 \quad \quad y > 3$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| b) $7 <= x \&& z > 4$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| c) $2 != y \&& z != 4$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| d) $x >= 0 \quad \quad x <= y$ | <input checked="" type="radio"/> T | <input type="radio"/> F |

$$x: 5, \quad y: 6, \quad z: 8$$

-2.5

1. Write a C++ function: string capitalize (const string& str) that gets the input string str and returns a string in which the initial character of str is capitalized (if it is a letter) and all of its other letters are converted to lowercase. Characters other than letters

2. Suppose that you have declared and initialized the variables s and t like this:

```
string s = "ABCDE", t = "";
```

Given these declarations, what is the effect of each of the following calls?

- a. s.length ();
- b. t.length ();
- c. s [2];
- d. s + t;
- e. t += 'a';
- f. s.replace (0, 2, "Z");
- g. s.substr (0, 3);
- h. s.substr (4);
- i. s.substr (3, 9);
- j. s.substr (3, 3);

a.

5

b.

0

c.

c.

d.

A B C D E X X

e.

x a

f.

ABCDE

Z CDE

g.

ABC

h.

E

i.

DE

j.

DE

-1

(5) Write a C++ program segment for each of the followings:

- Create an empty C++ string.
- Assign the name "Michael J. Fox" to the string.
- Erase the middle initial of the name in the string.
- Replace the first name "Michael" with "Mike" in the string.
- Create a new string from the existing one as the name in the new string will be "Fox, Mike".

a) int main()

```
{  
    string str1 = "O";  
}
```

b) int main()

```
{  
    string str1 = "Michael J. fox";  
}
```

c) int main()

```
{  
    string str1 = "Michael J. fox";  
    str1.erase(8,3); // J.-  
}
```

d) int main()

```
{  
    string str1 = "Michael J. fox";  
    str1.replace(0, 7, "Mike");  
}
```

e)

```
using namespace std;  
int main()  
{  
    string str1 = "Michael J. fox";  
    string str2, str3, str4;  
    str2 = str1.substr(5);  
    str3 = str1.substr(0, 4);  
    str4 = str2 + "+" + str3;  
}
```

NAME: VIJAY MURUGAPPAN SUBBIAH (Z1807314)

CSCI 689-1

Quiz 3
Closed Book and Notes
(10 points)

Fall 2016

-2.5

1. Write a C++ function: string capitalize (const string& str) that gets the input string str and returns a string in which the initial character of str is capitalized (if it is a letter) and all of its other letters are converted to lowercase. Characters other than letters are not affected. For example, both capitalize ("BOOLEAN") and capitalize ("boolean") should return the string "Boolean".

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string gerstring;
    cout << "Enter the desired string" << endl;
    cin >> gerstring;
    gerstring = capitalize(gerstring);
    cout << "the capitalized string is:" << gerstring;
}

string capitalize (const string &str)
{
    char c;
    for (int i=0; i<str.length(); i++)
    {
        c = str[i];
        if (isalpha(c))
            if (str[0])
                { else }
            else
                return str;
        str[i] = isupper(c);
        str[i] = str[i] + islower(c);
    }
}
```

2. There are total of 4 possible placements for the 6-queens puzzle. Using the backtracking approach, find out 1 of these 4 placements.

4	0	0	0	q	0	0
3	q	0	0	0	0	0
2	0	q	0	0	0	q
1	0	0	q	0	0	0
0	0	0	0	q	0	0
6	0	0	0	0	q	0
5	0	0	q	0	0	0

Step 1:-

Check if (row, col) is false place queen & ~~make true~~ return.

Step 2:-

if (row, col) is already true increment col if (row, col) is already true if col is same then place queen

$row+1$.

Step 3:-

if even a queen present in $row+1, col$. then check next row till size & col from size to 0 if false place queen.

Step 4:-

if queen present in $row \rightarrow size$, col size - 0. make col as row & row as col and repeat the same.

- 3.5 2. Suppose that the constant PI has been defined as

```
const double PI = 3.14159265358979323846;
```

What output manipulators would you use to produce each line of the following sample run?

3.141592653589793

3.141593

3.141593E+00

3.141592654

003.1416

i) ~~cout~~ ~~< >~~ 15d // d-double.

ii) ~~cout~~ 6d

iii) ~~cout~~ pow(10, 6d) // Scientific.

iv) 9d

v) .3d
F

8.1

NAME: JUBERINH VIBHUVISHNU (Z1861314)

CSCI 689-1

Quiz 8
Closed Book & Notes
(10 points)

Fall 2016

1. The following is a partial definition of the Vector class template:

```
template <class T> class Vector {  
public:  
    Vector ( const unsigned& =1 ); // default constructor  
    Vector ( const Vector < T >& ); // copy constructor  
    ~Vector (); // destructor  
    Vector < T >& operator= ( const Vector < T >& ); // assignment operator  
    T& operator [ ] ( const int& ) const; // subscription operator  
    unsigned size ( ) const; // returns size of an object  
    void print ( ) const; // prints elements of an object  
private:  
    unsigned sz; // size of an object  
    T* data; // points to data segment  
};
```

Modify this class template so that you add the member function, bool replace (const T& x, const T& y), which can be used to search for value x in a Vector object. If x is found, it replaces the first occurrence of x by y and returns true; otherwise, it returns false. Show your implementation of this function template.

Template <class T> class Vector {

```
public:  
    Vector ( const unsigned& =0 );  
    Vector ( const Vector < T >& );  
    ~Vector ();  
    Vector < T >& operator= ( const Vector < T >& );  
    T& operator [ ] ( const int& ) const;  
    bool replace ( const T& x, const T& y );  
    unsigned size ( ) const;  
    void print ( ) const;
```

```
private:  
    unsigned sz;  
    T* data;
```

}

template <class T> -5
bool vector < T > replace (const T& x,
 const T& y)

```
{  
    for ( int i = 0; i < sz; i++ )
```

```
    { if ( x == data [ i ] )
```

```
        { data [ i ] = y;  
        return true;
```

}

```
else  
    return false;
```

} -2

{

6.5

NAME: SUBBIAH, VIJAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 4
Closed Book and Notes
(10 points)

Fall 2016

1. What is the output of the following C++ program?

```
int main ()
{
    vector< int > v { 4, 8, 12, 16, 20 };
    for ( const int& x : v ) cout << x << ' ';
    cout << endl;

    set< int > s ( v.begin ( ) + 1, v.end ( ) - 1 );
    for ( const int& x : s ) cout << x << ' ';
    cout << endl;

    s.insert ( -4 ); s.insert ( 24 );
    for ( const int& x : s ) cout << x << ' ';
    cout << endl;

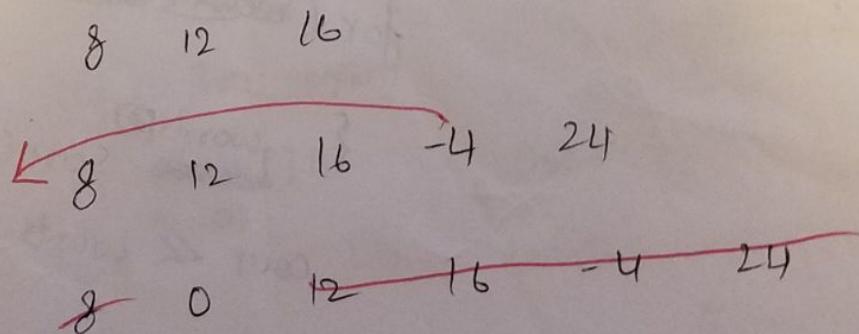
    auto p = s.insert ( ++s.begin ( ), 0 );
    cout << *p << endl;

    s.erase ( ++s.begin ( ), --s.end ( ) );
    cout << s.size ( ) << endl;

    return 0;
}
```

O/P.

4 8 12 16 20



2

-15

8. Write a recursive C++ function, vowels, that returns the number of vowels in a string object.

~~seen~~
~~int vowels (string str, const int count)~~

~~def~~
~~int vowels (const string str, const unsigned pos = 0)~~

{
 int count = 0;
 size_t size = str.size();
 if (pos < size)

{
 vowels (str, pos + 1);

~~if (str[pos] ==~~
~~'A' || tolower(str[pos]) == 'E'~~
~~'I' || tolower(str[pos]) == 'O'~~
~~'U' || tolower(str[pos]) == 'O')~~

{
 count++;
 return count + vowels (str, pos + 1);

→ 2.0

2. Write a C++ program (a) to read an English text from the `stdin`, one word at a time, and store those words in the set `< string >` container called `words`, and (b) to display the contents of two-letter words in `words` on `stdout` in alphabetical order.

```
#include <iostream>
#include <set>
using namespace std;
```

a) int main()

string s; words

Set<string> words(~~(~~ ~~vector~~, ~~vector~~, ~~end~~)~~,~~ 50);

~~Set<string>~~ cour << "Enter String";

cin >> s; // needs a loop, -s

~~words.insert("Object oriented programming");~~

words.insert(s);

b)

int s = 0 // wrong for loop - s

for (~~vector<string>~~; ~~s = 0~~; s < words.size(),

s++)

{ ~~words~~.sort(~~(~~)); // not needed - s

cour << words(s); // can use [s,

}

7. What is the output of the following C++ program?

```
void F ( const string&, const unsigned&, const unsigned& );
int main ( )
{
    string mystr = "Hello";
    cout << mystr << endl;

    F ( mystr, 0, mystr.size ( ) );
    cout << endl;

    return 0;
}

void F ( const string& str, const unsigned& pos, const unsigned& size )
{
    if ( pos < size ) {
        F ( str, pos + 1, size );
        cout << str [ pos ];
    }
}
```

Output

Hello
~~Hello~~
olleH

-O

↳ Hello⁴
↳ [pos]
n * print
5 * 4 * 3 * 2 * 1
Hello
Hello
Hello
Hello
Hello
Hello

7.0

4

NAME: SUBBIAH, VITAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 6
Closed Book and Notes
(10 points)

Fall 2016

- The following class can be used to represent long integers in a computer system:

```
class longInt {
public:
    longInt ( long = 0, unsigned long = 0 );
    longInt ( const longInt& );
    longInt& operator= ( const longInt& );
    ~longInt ( );
    void print ( ) const;
private:
    long msp;           // most significant part
    unsigned long lsp;  // least significant part
};
```

The following is a test program for this class:

```
int main ( )
{
    longInt x, y ( 1 ), z ( 1, 2 ), u ( z );

    cout << "x = "; x.print ( ); cout << endl;
    cout << "y = "; y.print ( ); cout << endl;
    cout << "z = "; z.print ( ); cout << endl;

    cout << "u = "; u.print ( ); cout << endl;
    u = y; cout << "u = "; u.print ( ); cout << endl;

    longInt v = longInt ( 0xff, 0xffffffff );
    cout << "v = "; v.print ( ); cout << endl;
    return 0;
}
```

And the output is:

```
cout << "0x" << hex << x; x = ( 0x0, 0x0 )
y = ( 0x1, 0x0 )
z = ( 0x1, 0x2 )
u = ( 0x1, 0x2 )
u = ( 0x1, 0x0 )
v = ( 0xff, 0xffffffff )
```

(i) `longInt::longInt()`

-1 { delete [] B;
} B = 0;

(ii)

print (r);
longInt::longInt (const longInt& B)

{ string P, Q, R;

R = B.to_string (msp);
Q = B.to_string (lsp);
P = P + " " + Q;
R = R + " " + Q;
print (P);

R = R + " " + Q;
print (R); } -4

Show the implementation of all member functions of the class longInt. Your print () function should print all objects of longInt in hexadecimal form.

(iii) `longInt::longInt& operator= (const longInt&)`

{ string P, Q, R;
P = B.to_string (msp);
Q = B.to_string (lsp);
R = P + " " + Q;
R = R + " " + Q;
print (*this); }

6. Write the implementations of the member functions as described in the definition of the class template Arr.

```
template < class T > class Arr {
public:
    Arr ( const unsigned& = 0, const T& = T() ); // constructor
    Arr ( const Arr& );
    Arr < T > & operator= ( const Arr& );
    ~Arr ();
    void print () const;
private:
    unsigned sz; T* p; // size of object and ptr to memory of object
};
```

Arr::Arr (const unsigned& a=0, const T& b=T()) // constructor

{

sz = a; p = new T[sz];
~~*p = b;~~; print();

copy ~~each~~ b to each
 location

}

Arr::Arr (const Arr &a) // copy constructor

{

sz = a.sz; p = new T[sz];
~~*p = a.*p;~~; (print());

-8

copy each value

}

T Arr < T > Arr::operator= (const Arr &a) // Assignment Operator

{

sz = a.sz;

~~*p = a.*p;~~

~~return *this;~~

~Arr::Arr() // destructor

{ delete [] a;

sz = 0;

*p = 0;

void Arr::print() const // print

{ cout << "Size of object is " << sz << endl; // cout << "Memory of object is " << *p << endl;

}

}

-8

6

DURBALI VIJAYMURUGAPPAN
CSCI 689-1

95

1

Fa

NAME: DURBALI VIJAYMURUGAPPAN (Z1807314)

CSCI 689-1

Fall 2016

Quiz 7
Closed Book and Notes
(10 points)

1. What is the output of the following C++ program?

```
int F ( const int& );  
  
int main ()  
{  
    const int x = 10;  
  
    cout << F ( x ) << endl;  
  
    return 0;  
}
```

```
int F ( const int& n ){ return n <= 0 ? 0 : F ( n - 1 ) + n; }
```

*(F(0)F(1)F(2)F(3)F(4)F(5)F(6)F(7)F(8)F(9)F(10)
(1+2+3+4+5+6+7+8+9+10)*

Output:

= 55

5. What is the output of the following C++ program?

```
string intToString ( const int& );  
  
class Date {  
    friend ostream& operator<< ( ostream&, const Date& );  
public:  
    Date ( const string& s ) { stringToDate ( s ); }  
    void setDate ( const string& s ) { stringToDate ( s ); }  
private:  
    int day, month, year;  
};  
  
void Date :: stringToDate ( const string& s )  
{  
    char a, b;  
    istringstream is ( s );  
    is >> day >> a >> month >> b >> year;  
}  
  
ostream& operator<< ( ostream& os, const Date& d )  
{  
    os << intToString ( d.month ) << " " << d.day << ", " << d.year;  
    return os;  
}  
  
int main ()  
{  
    Date peace ( "11-11-1918" );  
    cout << "World War I ended on " << peace << ".\n";  
    peace.setDate ( "14/8/1945" );  
    cout << "World War II ended on " << peace << ".\n";  
    return 0;  
}  
  
string intToString ( const int& n )  
{  
    switch ( n ) {  
        case 1: return "January";  
        case 2: return "February";  
        case 3: return "March";  
        case 4: return "April";  
        case 5: return "May";  
        case 6: return "June";  
        case 7: return "July";  
        case 8: return "August";  
        case 9: return "September";  
        case 10: return "October";  
        case 11: return "November";  
        case 12: return "December";  
        default: return "";  
    }  
}
```

Output:

World War I ended on November 11, 1918.
World War II ended on August 14, 1945.

- 15

2. Spherical objects, such as cannonballs, can be stacked to form a pyramid with one cannonball at the top, sitting on top of a square composed of four cannonballs, sitting on top of a square composed of nine cannonballs, and so forth. Write a recursive C++ function `cannonball()` that takes as its argument the height of the pyramid and returns the number of cannonballs it contains. Your function must operate recursively and must not use any iterative constructs, such as while or for.

Q Cannonball (int h)

{
if (h <= 0)
return 0;

else
return ((h * h) + Cannonball(h - 1));

4. Write a function template Max () that returns the maximum value of its three arguments. For example, Max (3, -2, 5) should return 5 and Max ("abc", "XYZ", "ABC") should return abc.

+ +
Template <Typename T>

T Max (T a, T b, T c) ✓

{

if (a > b && a > c)

return a;

else if (b > a && b > c)

return b;

else

return c;

}

-O

3

NAME: SUBBIAH, VIJAYMURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 9
Closed Book and Notes
(10 points)

Fall 2016

1. Consider the following statements:

```
class X {  
public:  
    void one ();  
    void two ( int, int );  
    X ();  
private:  
    int a, b;  
};
```



```
class Y : public X {  
public:  
    void one ();  
    Y ();  
private:  
    int c;  
};
```

Suppose the following statements are in a user program (client code): X x; Y y;

- a) True or false: the *private* members of X are *public* members of Y?
b) Mark the following statements as valid or invalid. If a statement is invalid, explain why.
- void X :: one () { cout << a + b << endl; }
 - x.a = 15; y.b = 30;
 - void Y :: one () { a = 10; b = 15; c = 30; cout << a + b + c << endl; }
 - cout << x.a << ' ' << x.b << ' ' << y.c << endl;

a) *False*, private members of X remain private for class Y. But it can be accessed in member functions of Y.

b) i) *Invalid* - private variables can only be accessed by assigning them to another variables in member functions.

ii) *Valid* - non-parameterized constructor.

iii) *Invalid* - values cannot be assigned directly to the private variables of base class.

iv) *Valid*.

3. Consider the definition of the following class:

```
class Test {  
public:  
    Test ( const int& = 0, const int& = 0 ); // constructor  
    Test ( const Test& ); // copy constructor  
    Test& operator= ( const Test& ); // assignment operator  
    ~Test (); // destructor  
    void print ( ) const; // prints out values of data members  
private:  
    int x, y;  
};
```

Write the implementations of the member functions as described in the definition of the class Test.

Test:: Test (const int& a=0, const int& b=0) //constructor

```
{  
    x=a;  
    y=b;  
    print();  
}
```

Test:: Test (const Test& T) //copy constructor

```
{  
    x=T.x;  
    y=T.y;  
    print();  
}
```

Test X Test:: operator= (const Test& T) //assignment operator

```
{  
    x=T.x;  
    y=T.y;  
    return *this;  
}
```

Test:: ~Test () //destructor

```
{  
    delete T;  
    x=0;  
    y=0;  
}
```

void Test:: print() const

```
{  
    cout << "value of x " << x << endl;  
    cout << "value of y " << y << endl;  
}
```

-2.5

2. The following is a partial definition of a linked-list structure:

```

template < class T > struct Node {
    T data;                                // data value
    Node* next;                             // points to next Node
    Node ( const T& = T( ), Node < T >* = nullptr ); // constructor
};

template < class T > class List {
    Node < T >* head;                      // points to 1st Node
public:
    List();                                 // default constructor
    void insert ( const T& x );             // inserts a Node with data value x
                                              // at the beginning of List
    void print () const;                   // displays elements of List on stdout,
                                              // separated by single spaces
};

```

Show implementations of:

- (a) constructor of the Node structure
- (b) default constructor of the List class
- (c) member function insert () of the List class
- (d) member function print () of the List class

~~A~~

~~2~~

~~L~~

~~Node (const T& = T(), Node < T >* = nullptr); data(T),~~

~~next (Node < T >*)~~

~~2}~~

~~List()~~

{ ~~head =~~ T.next;

~~Y~~

~~void insert (const T& x)~~

~~Q~~

~~head = T.next; head;~~

~~T.data = x;~~

~~Y~~

~~2) void print() const~~

~~{~~

~~for (int i = head; i->next != 0; i++)~~

~~{ cout << *x[i] << " "~~

~~}~~

~~2~~

2. Write a C++ program (a) to read an English text from the stdin, one word at a time, and store those words in the map < string, unsigned > words, and (b) to display the two-letter words in words and their frequencies on stdout in alphabetical order.

```
#include <iostream>
#include <map>
#include <string>
using namespace std;

int main()
{
    string Engstr;
    map<string, unsigned> words;
    while (cin >> Engstr) // (a)
    {
        words.insert(Engstr);
    }
    for (map<string, unsigned>::iterator m = m.begin(); m != m.end(); m++)
    {
        if (words[m].length() == 2) // (b)
        {
            cout << words[m] << endl;
        }
    }
    return 0;
}
```

-1 m is iterator
not int

NAME: SUBBIATH, VIJAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 10
Closed Book and Notes
(10 points)

Fall 2016

6

1. Suppose that you have the following classes, A and B:

```
class A {  
public:  
    A ( const int& = 0 );  
    void F ();  
    virtual void print () const;  
private:  
    int x;  
};  
  
A :: A ( const int& a ) : x ( a ) {}  
void A :: F () { x *= 2; }  
void A :: print () const {  
    cout << "A :: x = " << x << endl;  
}
```

```
class B : public A {  
public:  
    B ( const int& = 0, const int& = 0 );  
    void F ();  
    void print () const;  
private:  
    int y;  
};  
  
B :: B ( const int& a, const int& b ) : A ( a ), y ( b ) {}  
void B :: F () { A :: F (); y *= 2; }  
void B :: print () const {  
    A :: print (); cout << "B :: y = " << y << endl;  
}
```

- (a) What is the output of the following program?

```
int main ()  
{  
    A* p; A a ( 2 ); B b ( 3, 5 );  
    p = &a; p->F (); p->print ();  
    p = &b; p->F (); p->print ();  
    return 0;  
}
```

A :: x = 4
A :: x = 6
B :: y = 10 5

- (b) What is the output of the previous program if the definition of class A is replaced by the following definition?

```
class A {  
public:  
    A ( const int& = 0 );  
    virtual void f ();  
    virtual void print () const;  
private:  
    int x;  
};
```

A :: x = 4
A :: x = 6 10 6
B :: y = 10

1

- |

85

25

NAME: SUBBIAH, VIJAY MURUGAAPPAN (Z1807314)

CSCI 689-1

Midterm Exam 2
Closed Book and Notes
(100 points)

Fall 2016

- What is the output of the following C++ program?

```
int main ()
{
    deque< int > d { 4, 8, 12, 16, 20 };
    for ( const int& x : d ) { cout << x << ' '; } cout << endl;

    set< int > s ( d.begin () + 1, d.end () - 1 );
    for ( const int& x : s ) { cout << x << ' '; } cout << endl;

    s.insert ( -4 ); s.insert ( 24 );
    for ( const int& x : s ) { cout << x << ' '; } cout << endl;

    auto p = s.insert ( ++s.begin ( ), 0 );
    cout << *p << endl;

    s.erase ( ++s.begin ( ), --s.end ( ) );
    cout << s.size () << endl;

    return 0;
}
```

OUTPUT:-

4 8 12 16 20

8 12 16

-4 8 12 16 24

0

2

1 0

NAME: SUBBIATH, VIJAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 10
Closed Book and Notes
(10 points)

Fall 2016

6

1. Suppose that you have the following classes, A and B:

```
class A {  
public:  
    A ( const int& = 0 );  
    void F ();  
    virtual void print () const;  
private:  
    int x;  
};  
  
A :: A ( const int& a ) : x ( a ) {}  
void A :: F () { x *= 2; }  
void A :: print () const {  
    cout << "A :: x = " << x << endl;  
}
```

```
class B : public A {  
public:  
    B ( const int& = 0, const int& = 0 );  
    void F ();  
    void print () const;  
private:  
    int y;  
};  
  
B :: B ( const int& a, const int& b ) : A ( a ), y ( b ) {}  
void B :: F () { A :: F (); y *= 2; }  
void B :: print () const {  
    A :: print (); cout << "B :: y = " << y << endl;  
}
```

- (a) What is the output of the following program?

```
int main ()  
{  
    A* p; A a ( 2 ); B b ( 3, 5 );  
    p = &a; p->F (); p->print ();  
    p = &b; p->F (); p->print ();  
    return 0;  
}
```

A :: x = 4
A :: x = 6
B :: y = 10 5

- (b) What is the output of the previous program if the definition of class A is replaced by the following definition?

```
class A {  
public:  
    A ( const int& = 0 );  
    virtual void f ();  
    virtual void print () const;  
private:  
    int x;  
};
```

A :: x = 4
A :: x = 6 10 6
B :: y = 10

1

- |

(8) Suppose that the constant PI has been defined as

const double PI = 3.14159265358979323846;

What output manipulators would you use to produce each line of the following output? Answer each part as a separate question.

- a) 3.14
- b) 3.14159
- c) 3.14E+00
- d) - 3.14159
- e) 003.14159

a) ~~PI~~⁺ Serprecision(2);
cout << fixed << PI;

b) ~~PI~~⁺ Serprecision(5);
cout << fixed << PI;

c) ~~PI~~⁺ Serprecision(2); *other approach*
cout << scientific << PI;

d) PI. serfill('0')^{u u}
PI. setwidm(3); *outlevig*
PI. serprecision(5);
cout << fixed << PI;

e) PI. setwidm(2);
PI. setfill('0');
PI. serprecision(5);
cout << fixed << PI;

7.0

NAME: SUBBIAH, VIJAY MURUGAPPAN (21807314)

CSCI 689-1

Quiz 11
Closed Book and Notes
(10 points)

Fall 2016

1. The prototype of the *destructor* of a linked-list implementation of the List container is given as: template < class T > List < T > :: ~List (). Show (a) a *non-recursive* implementation and (b) a *recursive* implementation of this member function. The List container has only one *private* data element, Node < T >* head, points to the first element of a List object, where the partial definition of Node is given as:

template < class T > class List;

```
template < class T > class Node {  
friend class List < T >;  
private:  
    T data;  
    Node < T >* next;  
public:  
    Node ( const T& = T ( ), Node < T >* = nullptr );  
};
```

void List < T > :: clear (Node < T >*);

a) template < class T > List < T > :: ~List (* clear (head))

2.0 {
 head → next = nullptr; . . . you need to loop
 head → data = null;
 delete head;
}

b) void List < T > :: clear (Node < T >*)

1.0 {
 Node < T > * temp = new Node ();
 temp = head;
 head = ~~temp~~ next; clear (head → next);
 ~~delete head;~~
}

85
a
for the
85
85

(7) Write a C++ function: string lowerCase (const string& str) that gets the input string str and returns a string in which all characters of str is converted to lowercase (if it is a letter). Characters other than letters are not affected. For example, both lowerCase ("BOOLEAN") and capitalise ("BooleaN") should return the string "boolean".

lowercase

string lowerCase (const string& str)

```
{ cout << "Enter the input string:";  
cin >> str; // X-3  
for (int i = 0; i < str.length(); i++)  
{ if (!isalpha(str[i]))  
    if (isalpha(str[i]))  
    { if (isupper(str[i]))  
        str[i] = convert to lowercase toLower(str[i]); // X-3  
    } else  
    { str[i] = convert to uppercase str[i];  
    }  
}  
cout << "The lower case output is: " << str;  
}
```

NAME: VIJAY MURUGENDRAM
Midterm Exam
Book and Notes

CS

// ask the

full definition:

template <class T> class list

{

private:

Node<T> * head;

public:

void clear (Node<T>*);

~list (clear (head));

}

Template <class T> class node

{

friend class list<T>;

private:

T data;

Node<T> * next;

public:

Node (const T& = T(), Node<T>* = nullptr);

}

(6) An empty vector container is declared as `vector < int > v`. Write C++ statements for the followings:

- Add integers 13, 75, 28, and 35 in `v`.
- Double the value of each element of `v`.
- Print out the contents of `v` in reverse order on stdout on a single line separated by spaces.

a) `int main()`

```
{  
    vector<int> v;  
    vector<int>::iterator i;  
    v.push_back(13); i = v.begin();  
    v.push_back(75); v.insert(i++, 13);  
    v.push_back(28); v.insert(i++, 75);  
    v.push_back(35); v.insert(i++, 28);  
    v.insert(i++, 35); }  
    for (vector<int>::iterator iter = v.begin(), iter1 = v.end();  
         iter1 != v.end(); iter++)
```

b)

```
{  
    *iter = *iter * 2;  
}  
for (vector<int>::iterator iter = v.begin(), iter1 = v.end();  
     iter1 != v.end(); iter1++)
```

{ cout << "doubled element is: " << *iter << endl;

}

c) `for (vector<int>::iterator iter = v.end(); iter1 != v.begin();`

~~X~~ iter--)

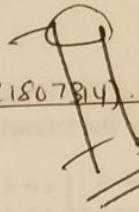
```
{ cout << "Reversed element is: " << *iter << endl;  
}
```

NAME: VIJAY MURUGANPRAN SUBBIAH (2180781)

CSCI 589-1

Midterm Exam 1
Closed Book and Notes
(100 points)

Fall 2016



- (1) Write a C++ program that calculates a car's gas mileage. The program should ask the user to enter the number of gallons of gas the car can hold and the number of miles it can be driven on a full tank. It should then display the number of miles that may be driven per gallon of gas.

in main ()

```
{ double
float gallons, mileage, capacity;
cout << "Enter the number of gallons of gas the
car can hold";
```

```
cin >> gallons; // gas
```

```
cout << "Enter the number of miles it can be driven
on a full tank";
```

```
cin >> mileage; // no. of miles.
```

```
capacity = mileage / gallons;
```

```
cout << "Number of miles that may be driven
per gallon of gas: " << capacity; // MPG.
```

3

(4) Convert the following while loop to a for loop:

```
int count = 0;  
while ( count < 10 ) {  
    cout << "count is " << count << endl;  
    count++;  
}
```

for loop:

```
for( int count = 0; count < 10; count++ )  
{  
    cout << "count is " << count << endl;  
}
```

(2) Assume the variables $x = 5$, $y = 6$, and $z = 8$. Indicate by circling the T or F whether each of the following conditions is true or false:

- | | | |
|-----------------------------------|------------------------------------|-------------------------|
| a) $x == 5 \quad \quad y > 3$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| b) $7 <= x \&& z > 4$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| c) $2 != y \&& z != 4$ | <input checked="" type="radio"/> T | <input type="radio"/> F |
| d) $x >= 0 \quad \quad x <= y$ | <input checked="" type="radio"/> T | <input type="radio"/> F |

$$x: 5, \quad y: 6, \quad z: 8$$

(3) Write a C++ program segment to add integers between a and b (inclusive), where $a \leq b$.

```

int main()
{
    int a, b, c;
    cout << "Enter First Value";
    cin >> a;
    cout << "Enter Second Value";
    cin >> b;

    if (a > b)
    {
        cout << "Invalid values";
    }
    else
    {
        for (i = a; i <= b; i++)
        {
            c = a + b;
        }
        cout << "Result sum is: " << c;
    }
}

```

~~(3)~~ Write a C++ program segment to add integers between a and b (inclusive), where a $\leq b$.

```

int main()
{
    int a, b, c;
    cout << "Enter First Value";
    cin >> a;
    cout << "Enter Second Value";
    cin >> b;

    if (a > b)
    {
        cout << "Invalid values";
    }
    else
    {
        for (i = a; i <= b; i++)
        {
            c = a + b;
        }
        cout << "Result sum is: " << c;
    }
}

```

NAME: VIJAY MURUGANPRAN SUBBIAH (2180781)

CSCI 589-1

Midterm Exam 1
Closed Book and Notes
(100 points)

Fall 2016

- (1) Write a C++ program that calculates a car's gas mileage. The program should ask the user to enter the number of gallons of gas the car can hold and the number of miles it can be driven on a full tank. It should then display the number of miles that may be driven per gallon of gas.

in main ()

{
double
gallons, mileage, capacity;
cout << "Enter the number of gallons of gas the
car can hold";

cin >> gallons; // gas

cout << "Enter the number of miles it can be driven
on a full tank";

cin >> mileage; // no. of miles.

capacity = mileage / gallons;

cout << "Number of miles that may be driven
per gallon of gas: " << capacity; // MPG.

3

(4) Convert the following while loop to a for loop:

```
int count = 0;  
while ( count < 10 ) {  
    cout << "count is " << count << endl;  
    count++;  
}
```

for loop:

```
for( int count = 0; count < 10; count++ )  
{  
    cout << "count is " << count << endl;  
}
```

7.0

NAME: SUBBIAH, VIJAY MURUGAPPAN (21807314)

CSCI 689-1

Quiz 11
Closed Book and Notes
(10 points)

Fall 2016

1. The prototype of the *destructor* of a linked-list implementation of the List container is given as: template < class T > List < T > :: ~List (). Show (a) a *non-recursive* implementation and (b) a *recursive* implementation of this member function. The List container has only one *private* data element, Node < T >* head, points to the first element of a List object, where the partial definition of Node is given as:

template < class T > class List;

```
template < class T > class Node {  
friend class List < T >;  
private:  
    T data;  
    Node < T >* next;  
public:  
    Node ( const T& = T ( ), Node < T >* = nullptr );  
};
```

void List < T > :: clear (Node < T >*);

a) template < class T > List < T > :: ~List (* clear (head))

2.0 {
 head → next = nullptr; . . . you need to loop
 head → data = null;
 delete head;
}

b) void List < T > :: clear (Node < T >*)

1.0 {
 Node < T > * temp = new Node ();
 temp = head;
 head = ~~temp~~ next; clear (head → next);
 ~~delete head;~~
}

(5) Write a C++ program segment for each of the followings:

- Create an empty C++ string.
- Assign the name "Michael J. Fox" to the string.
- Erase the middle initial of the name in the string.
- Replace the first name "Michael" with "Mike" in the string.
- Create a new string from the existing one as the name in the new string will be "Fox, Mike".

a) `int main()`

```
{  
    string str1 = "O";  
}
```

b) `int main()`

```
{  
    string str1 = "Michael J. fox";  
}
```

c) `int main()`

```
{  
    string str1 = "Michael J. fox";  
    str1.erase(8,3); // J.-  
}
```

d) `int main()`

```
{  
    string str1 = "Michael J. fox";  
    str1.replace(0, 7, "Mike");  
}
```

e)

```
using namespace std;  
int main()  
{  
    string str1 = "Michael J. fox";  
    string str2, str3, str4;  
    str2 = str1.substr(5);  
    str3 = str1.substr(0, 4);  
    str4 = str2 + "+" + str3;  
}
```

2. The prototype of the *copy constructor* of a linked-list implementation of the Stack container is given as: template < class T > Stack < T > :: Stack (const Stack < T > &). Show an implementation of this *public* member function. The Stack container has only one *private* data element, Node < T > * top, points to the top element in a Stack object.

Template < class T > Stack < T > :: Stack (const Stack < T > & S)

{

~~Stack < T > St > top = S. St > top;~~

~~Stack < T > St > ;~~

~~St > top = S. St > top;~~

}

(6) An empty vector container is declared as `vector < int > v`. Write C++ statements for the followings:

- Add integers 13, 75, 28, and 35 in `v`.
- Double the value of each element of `v`.
- Print out the contents of `v` in reverse order on stdout on a single line separated by spaces.

a) `int main()`

```
{  
    vector<int> v;  
    vector<int>::iterator i;  
    v.push_back(13); i = v.begin();  
    v.push_back(75); v.insert(i++, 13);  
    v.push_back(28); v.insert(i++, 75);  
    v.push_back(35); v.insert(i++, 28);  
    v.insert(i++, 35); }  
    for (vector<int>::iterator iter = v.begin(), iter1 = v.end();  
         iter1++)
```

b)

```
{  
    *iter = *iter * 2;  
}  
for (vector<int>::iterator iter = v.begin(), iter1 = v.end();  
     iter1++)
```

{ cout << "doubled element is: " << *iter << endl;

}

c) `for (vector<int>::iterator iter = v.end(); iter1 = v.begin();`

~~X~~ iter--)

```
{ cout << "Reversed element is: " << *iter << endl;  
}
```

NAME: SUBBIATH, VIJAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 10
Closed Book and Notes
(10 points)

Fall 2016

6

1. Suppose that you have the following classes, A and B:

```
class A {  
public:  
    A ( const int& = 0 );  
    void F ();  
    virtual void print () const;  
private:  
    int x;  
};  
  
A :: A ( const int& a ) : x ( a ) {}  
void A :: F () { x *= 2; }  
void A :: print () const {  
    cout << "A :: x = " << x << endl;  
}
```

```
class B : public A {  
public:  
    B ( const int& = 0, const int& = 0 );  
    void F ();  
    void print () const;  
private:  
    int y;  
};  
  
B :: B ( const int& a, const int& b ) : A ( a ), y ( b ) {}  
void B :: F () { A :: F (); y *= 2; }  
void B :: print () const {  
    A :: print (); cout << "B :: y = " << y << endl;  
}
```

- (a) What is the output of the following program?

```
int main ()  
{  
    A* p; A a ( 2 ); B b ( 3, 5 );  
    p = &a; p->F (); p->print ();  
    p = &b; p->F (); p->print ();  
    return 0;  
}
```

A :: x = 4
A :: x = 6
B :: y = 10 5

- (b) What is the output of the previous program if the definition of class A is replaced by the following definition?

```
class A {  
public:  
    A ( const int& = 0 );  
    virtual void f ();  
    virtual void print () const;  
private:  
    int x;  
};
```

A :: x = 4
A :: x = 6 10 6
B :: y = 10

1

- |

85
a
for the
85
85

(7) Write a C++ function: string lowerCase (const string& str) that gets the input string str and returns a string in which all characters of str is converted to lowercase (if it is a letter). Characters other than letters are not affected. For example, both lowerCase ("BOOLEAN") and capitalise ("BooleaN") should return the string "boolean".

lowercase

string lowerCase (const string& str)

```
{ cout << "Enter the input string:";  
cin >> str; // X-3  
for (int i = 0; i < str.length(); i++)  
{ if (!isalpha(str[i]))  
    if (isalpha(str[i]))  
    { if (isupper(str[i]))  
        str[i] = convert to lowercase toLower(str[i]); // X-3  
    } else  
    { str[i] = convert to uppercase str[i];  
    }  
}  
else  
{ str[i] = convert to uppercase str[i];  
}  
cout << "The lower case output is: " << str;  
}
```

3

NAME: SUBBIAH, VIJAYMURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 9
Closed Book and Notes
(10 points)

Fall 2016

1. Consider the following statements:

```
class X {  
public:  
    void one ();  
    void two ( int, int );  
    X ();  
private:  
    int a, b;  
};
```



```
class Y : public X {  
public:  
    void one ();  
    Y ();  
private:  
    int c;  
};
```

Suppose the following statements are in a user program (client code): X x; Y y;

- a) True or false: the *private* members of X are *public* members of Y?
b) Mark the following statements as valid or invalid. If a statement is invalid, explain why.
- void X :: one () { cout << a + b << endl; }
 - x.a = 15; y.b = 30;
 - void Y :: one () { a = 10; b = 15; c = 30; cout << a + b + c << endl; }
 - cout << x.a << ' ' << x.b << ' ' << y.c << endl;

a) *False*, private members of X remain private for class Y. But it can be accessed in member functions of Y.

b) i) *invalid* - private variables can only be accessed by assigning them to another variables in member functions.

ii) *valid* - non-parameterized constructor.

iii) *invalid* - values cannot be assigned directly to the private variables of base class.

iv) *valid*.

(8) Suppose that the constant PI has been defined as

const double PI = 3.14159265358979323846;

What output manipulators would you use to produce each line of the following output? Answer each part as a separate question.

- a) 3.14
- b) 3.14159
- c) 3.14E+00
- d) - 3.14159
- e) 003.14159

a) ~~PI~~⁺ Serprecision(2);
cout << fixed << PI;

b) ~~PI~~⁺ Serprecision(5);
cout << fixed << PI;

c) ~~PI~~⁺ Serprecision(2); *other approach*
cout << scientific << PI;

d) PI. serfill('0')^{u u}
PI. setwidm(3); *outlevig*
PI. serprecision(5);
cout << fixed << PI;

e) PI. setwidm(2);
PI. setfill('0');
PI. serprecision(5);
cout << fixed << PI;

DURBALI VIJAYMURUGAPPAN
CSCI 689-1

95

1

Fa

NAME: DURBALI VIJAYMURUGAPPAN (Z1807314)

CSCI 689-1

Fall 2016

Quiz 7
Closed Book and Notes
(10 points)

1. What is the output of the following C++ program?

```
int F ( const int& );  
  
int main ()  
{  
    const int x = 10;  
  
    cout << F ( x ) << endl;  
  
    return 0;  
}
```

```
int F ( const int& n ){ return n <= 0 ? 0 : F ( n - 1 ) + n; }
```

*(F(1)F(2) F(3) F(4) F(5) F(6) F(7) F(8) F(9) F(10)
(1+2+3+4+5+6+7+8+9+10)*

Output:

= 55

85

25

NAME: SUBBIAH, VIJAY MURUGAAPPAN (Z1807314)

CSCI 689-1

Midterm Exam 2
Closed Book and Notes
(100 points)

Fall 2016

1. What is the output of the following C++ program?

```
int main ()
{
    deque< int > d { 4, 8, 12, 16, 20 };
    for ( const int& x : d ) { cout << x << ' '; } cout << endl;

    set< int > s ( d.begin () + 1, d.end () - 1 );
    for ( const int& x : s ) { cout << x << ' '; } cout << endl;

    s.insert ( -4 ); s.insert ( 24 );
    for ( const int& x : s ) { cout << x << ' '; } cout << endl;

    auto p = s.insert ( ++s.begin ( ), 0 );
    cout << *p << endl;

    s.erase ( ++s.begin ( ), --s.end ( ) );
    cout << s.size () << endl;

    return 0;
}
```

OUTPUT:-

4 8 12 16 20

8 12 16

-4 8 12 16 24

0

2

1 0

2. What is the output of the following C++ program?

```

template < class T > class Arr {
public:
    Arr ( const unsigned& s = 0, const T& x = T ( ) );
    sz ( s ), p ( new T [ sz ] ) {
        for ( unsigned i = 0; i < sz; i++ ) p [ i ] = x;
    }
    Arr ( const Arr& a ) : sz ( a.sz ) { copy ( a ); }
    Arr < T > & operator= ( const Arr& a ) {
        if ( this != &a ) { delete [ ] p; copy ( a ); }
        return *this;
    }
    ~Arr ( ) { delete [ ] p; p = 0; }
    void print ( ) const {
        for ( unsigned i = 0; i < sz; i++ ) cout << p [ i ] << ' ';
        cout << endl;
    }
private:
    unsigned sz; T* p;
    void copy ( const Arr& a ) { sz = a.sz; p = new T [ sz ];
        for ( unsigned i = 0; i < sz; i++ ) p [ i ] = a.p [ i ];
    }
};

int main ( )
{
    Arr < int > A, B ( 10, 1 ), C ( B ), D ( 5 );
    Arr < char > E ( 8, 'X' );

    cout << "A: "; A.print ( );
    cout << "B: "; B.print ( );
    cout << "C: "; C.print ( );

    cout << "D: "; D.print ( );
    D = C;
    cout << "D: "; D.print ( );

    cout << "E: "; E.print ( );
    return 0;
}

```

Output:

A: 0
B:
~~(10 times)~~

C:
~~(10 times)~~

D:
~~(5 times)~~

no new lines

E: X
X
X
X
X
X
X

2 -2

2. Write a C++ program (a) to read an English text from the stdin, one word at a time, and store those words in the map < string, unsigned > words, and (b) to display the two-letter words in words and their frequencies on stdout in alphabetical order.

```
#include <iostream>
#include <map>
#include <string>
using namespace std;

int main()
{
    string Engstr;
    map<string, unsigned> words;
    while (cin >> Engstr) // (a)
    {
        words.insert(Engstr);
    }
    for (map<string, unsigned>::iterator m = m.begin(); m != m.end(); m++)
    {
        if (words[m].length() == 2) // (b)
        {
            cout << words[m] << endl;
        }
    }
    return 0;
}
```

-1 m is iterator
not int

7.0

4

NAME: SUBBIAH, VITAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 6
Closed Book and Notes
(10 points)

Fall 2016

- The following class can be used to represent long integers in a computer system:

```
class longInt {
public:
    longInt ( long = 0, unsigned long = 0 );
    longInt ( const longInt& );
    longInt& operator= ( const longInt& );
    ~longInt ( );
    void print ( ) const;
private:
    long msp;           // most significant part
    unsigned long lsp;  // least significant part
};
```

The following is a test program for this class:

```
int main ( )
{
    longInt x, y ( 1 ), z ( 1, 2 ), u ( z );

    cout << "x = "; x.print ( ); cout << endl;
    cout << "y = "; y.print ( ); cout << endl;
    cout << "z = "; z.print ( ); cout << endl;

    cout << "u = "; u.print ( ); cout << endl;
    u = y; cout << "u = "; u.print ( ); cout << endl;

    longInt v = longInt ( 0xff, 0xffffffff );
    cout << "v = "; v.print ( ); cout << endl;
    return 0;
}
```

And the output is:

```
cout << "0x" << hex << x; x = ( 0x0, 0x0 )
y = ( 0x1, 0x0 )
z = ( 0x1, 0x2 )
u = ( 0x1, 0x2 )
u = ( 0x1, 0x0 )
v = ( 0xff, 0xffffffff )
```

(i) `longInt::longInt()`

-1 { delete [] B;
} B = 0;

(ii)

print (r);
longInt::longInt (const longInt& B)

{ string P, Q, R;

R = B.to_string (msp);
Q = B.to_string (lsp);
P = P + " " + Q;
R = R + " " + Q;
print (P);

R = R + " " + Q;
print (R); } -4

Show the implementation of all member functions of the class longInt. Your print () function should print all objects of longInt in hexadecimal form.

(iii) `longInt::longInt& operator= (const longInt&)`

{ string P, Q, R;
P = B.to_string (msp);
Q = B.to_string (lsp);
R = P + " " + Q;
R = R + " " + Q;
print (*this); }

3. Consider the definition of the following class:

```
class Test {  
public:  
    Test ( const int& = 0, const int& = 0 ); // constructor  
    Test ( const Test& ); // copy constructor  
    Test& operator= ( const Test& ); // assignment operator  
    ~Test (); // destructor  
    void print ( ) const; // prints out values of data members  
private:  
    int x, y;  
};
```

Write the implementations of the member functions as described in the definition of the class Test.

Test:: Test (const int& a=0, const int& b=0) //constructor

```
{  
    x=a;  
    y=b;  
    print();  
}
```

Test:: Test (const Test& T) //copy constructor

```
{  
    x=T.x;  
    y=T.y;  
    print();  
}
```

Test X Test:: operator= (const Test& T) //assignment operator

```
{  
    x=T.x;  
    y=T.y;  
    return *this;  
}
```

Test:: ~Test () //destructor

```
{  
    delete T;  
    x=0;  
    y=0;  
}
```

void Test:: print() const

```
{  
    cout << "value of x " << x << endl;  
    cout << "value of y " << y << endl;  
}
```

-2.5

2. Write a C++ program (a) to read an English text from the `stdin`, one word at a time, and store those words in the set `< string >` container called `words`, and (b) to display the contents of two-letter words in `words` on `stdout` in alphabetical order.

```
#include <iostream>
#include <set>
using namespace std;
```

a) int main()

string s; words

Set<string> words(~~(~~ ~~vector~~, ~~vector~~, ~~end~~)~~,~~ 50);

~~Set<string>~~ cour << "Enter String";

cin >> s; // needs a loop, -s

~~words.insert("Object oriented programming");~~

words.insert(s);

b)

int s = 0 // wrong for loop - s

for (~~vector<string>~~; ~~s = 0~~; s < words.size();

s++)

{ ~~words~~.sort(~~(~~)); // not needed - s

cour << words(s); // can use [s,

}

2 -2

4. Write a function template Max () that returns the maximum value of its three arguments. For example, Max (3, -2, 5) should return 5 and Max ("abc", "XYZ", "ABC") should return abc.

+ +
Template <Typename T>

T Max (T a, T b, T c) ✓

{

if (a > b && a > c)

return a;

else if (b > a && b > c)

return b;

else

return c;

}

-O

6.5

NAME: SUBBIAH, VIJAY MURUGAPPAN (Z1807314)

CSCI 689-1

Quiz 4
Closed Book and Notes
(10 points)

Fall 2016

1. What is the output of the following C++ program?

```
int main ()
{
    vector< int > v { 4, 8, 12, 16, 20 };
    for ( const int& x : v ) cout << x << ' ';
    cout << endl;

    set< int > s ( v.begin ( ) + 1, v.end ( ) - 1 );
    for ( const int& x : s ) cout << x << ' ';
    cout << endl;

    s.insert ( -4 ); s.insert ( 24 );
    for ( const int& x : s ) cout << x << ' ';
    cout << endl;

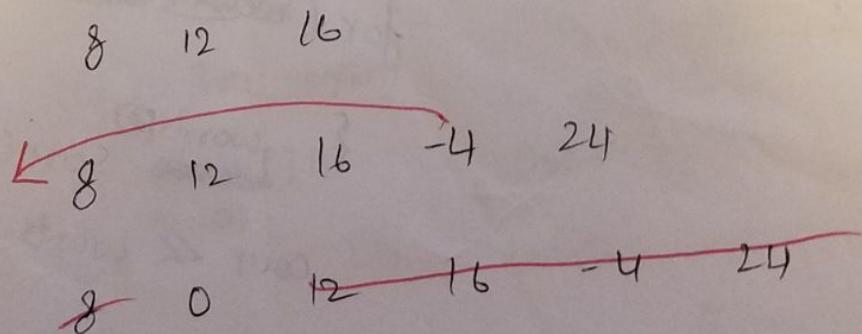
    auto p = s.insert ( ++s.begin ( ), 0 );
    cout << *p << endl;

    s.erase ( ++s.begin ( ), --s.end ( ) );
    cout << s.size ( ) << endl;

    return 0;
}
```

O/P.

4 8 12 16 20



2

-15

5. What is the output of the following C++ program?

```
string intToString ( const int& );  
  
class Date {  
    friend ostream& operator<< ( ostream&, const Date& );  
public:  
    Date ( const string& s ) { stringToDate ( s ); }  
    void setDate ( const string& s ) { stringToDate ( s ); }  
private:  
    int day, month, year;  
};  
  
void Date :: stringToDate ( const string& s )  
{  
    char a, b;  
    istringstream is ( s );  
    is >> day >> a >> month >> b >> year;  
}  
  
ostream& operator<< ( ostream& os, const Date& d )  
{  
    os << intToString ( d.month ) << " " << d.day << ", " << d.year;  
    return os;  
}  
  
int main ()  
{  
    Date peace ( "11-11-1918" );  
    cout << "World War I ended on " << peace << ".\n";  
    peace.setDate ( "14/8/1945" );  
    cout << "World War II ended on " << peace << ".\n";  
    return 0;  
}  
  
string intToString ( const int& n )  
{  
    switch ( n ) {  
        case 1: return "January";  
        case 2: return "February";  
        case 3: return "March";  
        case 4: return "April";  
        case 5: return "May";  
        case 6: return "June";  
        case 7: return "July";  
        case 8: return "August";  
        case 9: return "September";  
        case 10: return "October";  
        case 11: return "November";  
        case 12: return "December";  
        default: return "";  
    }  
}
```

Output:

World War I ended on November 11, 1918.
World War II ended on August 14, 1945.

- 15

- 3.5 2. Suppose that the constant PI has been defined as

```
const double PI = 3.14159265358979323846;
```

What output manipulators would you use to produce each line of the following sample run?

3.141592653589793

3.141593

3.141593E+00

3.141592654

003.1416

i) ~~cout~~ ~~< >~~ 15d // d-double.

ii) ~~cout~~ 6d

iii) ~~cout~~ pow(10, 6d) // Scientific.

iv) 9d

v) .3d
F

6. Write the implementations of the member functions as described in the definition of the class template Arr.

```
template < class T > class Arr {
public:
    Arr ( const unsigned& = 0, const T& = T() ); // constructor
    Arr ( const Arr& );
    Arr < T > & operator= ( const Arr& );
    ~Arr ();
    void print () const;
private:
    unsigned sz; T* p; // size of object and ptr to memory of object
};
```

Arr::Arr (const unsigned& a=0, const T& b=T()) // constructor

{

sz = a; p = new T[sz];
~~*p = b;~~; print();

copy ~~each~~ b to each
 location

}

Arr::Arr (const Arr &a) // copy constructor

{

sz = a.sz; p = new T[sz];
~~*p = a.*p;~~; (print());

-8

copy each value

}

T Arr < T > Arr::operator= (const Arr &a) // Assignment Operator

{

sz = a.sz;

~~*p = a.*p;~~

~~return *this;~~

~Arr::Arr() // destructor

{ delete [] a;

sz = 0;

*p = 0;

void Arr::print() const // print

{ cout << "Size of object is " << sz << endl; // cout << "Memory of object is " << *p << endl;

}

}

-8

6

NAME: VIJAY MURUGAPPAN SUBBIAH (Z1807314)

CSCI 689-1

Quiz 3
Closed Book and Notes
(10 points)

Fall 2016

-2.5

1. Write a C++ function: string capitalize (const string& str) that gets the input string str and returns a string in which the initial character of str is capitalized (if it is a letter) and all of its other letters are converted to lowercase. Characters other than letters are not affected. For example, both capitalize ("BOOLEAN") and capitalize ("boolean") should return the string "Boolean".

```
#include <iostream>
#include <string>
using namespace std;

int main()
{
    string gerstring;
    cout << "Enter the desired string" << endl;
    cin >> gerstring;
    gerstring = capitalize(gerstring);
    cout << "the capitalized string is:" << gerstring;
}

string capitalize (const string &str)
{
    char c;
    for (int i=0; i<str.length(); i++)
    {
        c = str[i];
        if (isalpha(c))
            if (str[0])
                { else }
            else
                return str;
        str[i] = isupper(c);
        str[i] = str[i] + islower(c);
    }
}
```

7. What is the output of the following C++ program?

```
void F ( const string&, const unsigned&, const unsigned& );
int main ( )
{
    string mystr = "Hello";
    cout << mystr << endl;

    F ( mystr, 0, mystr.size ( ) );
    cout << endl;

    return 0;
}

void F ( const string& str, const unsigned& pos, const unsigned& size )
{
    if ( pos < size ) {
        F ( str, pos + 1, size );
        cout << str [ pos ];
    }
}
```

Output

Hello
~~Hello~~
olleH

>Hello⁴
Hello
Hello
Hello
Hello
Hello

{pos:
n*pos+1
5*4*3*2*1
0

-2.5

1. Write a C++ function: string capitalize (const string& str) that gets the input string str and returns a string in which the initial character of str is capitalized (if it is a letter) and all of its other letters are converted to lowercase. Characters other than letters

2. Suppose that you have declared and initialized the variables s and t like this:

```
string s = "ABCDE", t = "";
```

Given these declarations, what is the effect of each of the following calls?

- a. s.length ();
- b. t.length ();
- c. s [2];
- d. s + t;
- e. t += 'a';
- f. s.replace (0, 2, "Z");
- g. s.substr (0, 3);
- h. s.substr (4);
- i. s.substr (3, 9);
- j. s.substr (3, 3);

a.

5

b.

0

c.

c.

d.

A B C D E X X

e.

x a

f.

ABCDE

Z CDE

g.

ABC

h.

E

i.

DE

j.

DE

-1

8. Write a recursive C++ function, vowels, that returns the number of vowels in a string object.

~~seen~~
~~int vowels (string str, const int count)~~

~~def~~
~~int vowels (const string str, const unsigned pos = 0)~~

{
 int count = 0;
 size_t size = str.size();
 if (pos < size)

{
 vowels (str, pos + 1);

~~if (str[pos] ==~~
~~'A' || tolower(str[pos]) == 'E'~~
~~'I' || tolower(str[pos]) == 'O'~~
~~'U')~~
 if (toupper(str[pos]) == 'A' || toupper(str[pos]) == 'E'
 || toupper(str[pos]) == 'I' || toupper(str[pos]) == 'O'
 || toupper(str[pos]) == 'U')

{
 count++;
 return count + vowels (str, pos + 1);

→ 2.0

NAME: VIJAY MURUGAPPAN SUBBIAH (21807314).

CSCI 689-1

Quiz 2
Closed Book and Notes
(10 points)

Fall 2016

1. Circle the best answer for each following C++ segments:

a) if (60 <= 12 + 5)
cout << "Hello";
cout << " There ";

outputs the following:

(i) Hello There (ii) Hello

(iii) Hello There

(iv) There

- |

b) if ('a' > 'b' || 'B' > 'A')
cout << "#*#" << endl;

outputs the following:

(i) #*#

(ii) # *

(iii) * (iv) none of these

Ans. → (i)

#

c) if (7 <= 7)
cout << 6 - 9 * 2 / 6 << endl;

outputs the following:

(i) -1 (ii) 3 (iii) 3.0

(iv) none of these

d) if (7 < 8) {
cout << "2 4 6 8" << endl;
cout << "1 3 5 7" << endl;
}

outputs the following:

(i) 2 4 6 8 (ii) 1 3 5 7 (iii) none of these

1 3 5 7

e) if (5 < 3)
cout << "*";
else if (7 == 8)
cout << "&";
else
cout << "\$";

outputs the following:

(i) * (ii) & (iii) \$ (iv) none of these

8.1

NAME: JUBERINH VISHNUVATHUPPAN (Z1861314)

CSCI 689-1

Quiz 8
Closed Book & Notes
(10 points)

Fall 2016

1. The following is a partial definition of the Vector class template:

```
template <class T> class Vector {  
public:  
    Vector ( const unsigned& =1 ); // default constructor  
    Vector ( const Vector < T >& ); // copy constructor  
    ~Vector (); // destructor  
    Vector < T >& operator= ( const Vector < T >& ); // assignment operator  
    T& operator [ ] ( const int& ) const; // subscription operator  
    unsigned size ( ) const; // returns size of an object  
    void print ( ) const; // prints elements of an object  
private:  
    unsigned sz; // size of an object  
    T* data; // points to data segment  
};
```

Modify this class template so that you add the member function, bool replace (const T& x, const T& y), which can be used to search for value x in a Vector object. If x is found, it replaces the first occurrence of x by y and returns true; otherwise, it returns false. Show your implementation of this function template.

Template <class T> class Vector {

```
public:  
    Vector ( const unsigned& =0 );  
    Vector ( const Vector < T >& );  
    ~Vector ();  
    Vector < T >& operator= ( const Vector < T >& );  
    T& operator [ ] ( const int& ) const;  
    bool replace ( const T& x, const T& y );  
    unsigned size ( ) const;  
    void print ( ) const;
```

```
private:  
    unsigned sz;  
    T* data;
```

}

template <class T> -5
bool vector < T > -2
replace (const T& x,
 const T& y)

```
{  
    for ( int i = 0; i < sz; i++ )  
    {  
        if ( x == data [ i ] )  
        {  
            data [ i ] = y;  
            return true;  
        }  
    }  
    else  
        return false;  
}
```

- 10
2. Write a C++ program that calculates a user's pay. The program reads the number of hours that the user worked and the amount paid per hour from `stdin`, and then computes the total pay for the user and displays it on `stdout`.

```
#include <iostream>
using namespace std;

int initialize (int hours, int pay)
{
    int compute (int hours, int pay)
    {
        int resultpay;
        resultpay = hours * pay;
        return resultpay;
    }

    int main()
    {
        int hours, pay, totalpay;
        cout << "Enter the number of hours worked" << endl;
        cin >> hours;
        cout << "Enter the pay for each hour" << endl;
        cin >> pay;
        totalpay = compute (hours, pay);
        cout << "total pay for student is" << totalpay << endl;
    }
}
```

* Testcase *

Enter the number of hours worked

10

Enter the pay for each hour

2

total pay for student is 20

2

21807314

2. There are total of 4 possible placements for the 6-queens puzzle. Using the backtracking approach, find out 1 of these 4 placements.

4	0	0	0	q	0	0
3	q	0	0	0	0	0
2	0	q	0	0	0	q
1	0	0	q	0	0	0
0	0	0	0	q	0	0
6	0	0	0	0	q	0
5	0	0	0	0	0	q

Step 1:-

Check if (row, col) is false place queen & ~~make true~~ return.

Step 2:-

if (row, col) is already true increment col if (row, col) is already true if col is same then place queen

$row+1$.

Step 3:-

if even a queen present in $row+1, col$. then check next row till size & col from size to 0 if false place queen.

Step 4:-

if queen present in $row \rightarrow size$, col size - 0. make col as row & row as col and repeat the same.

10

NAME: VIJAY MURUGAPPAN SUBBIAH (Z1807314)

CSCI 689-1

Fall 2016

Quiz 1
Closed Book and Notes
(10 points)

1. Answer each part of this question.

- a) What characters are used to mark comments in a C++ program?
b) How would you define a constant called CENTIMETERS_PER_INCH with the value 2.54?
c) What is the name of the function that must be defined in every C++ program?
What statement typically appears at the end of that function?
d) What is the purpose endl when you are writing to the cout output stream?
e) List all possible values of type bool.

a) 1. //

2. /*---*/

b)

#define CENTIMETERS_PER_INCH 2.54

c)

main() function, return a end if the type is not void.
for this function main() it cannot be void

d)

It take the cursor to the next line for the next
execution statement.

e)

* true - 0
* false - 1

10

NAME: VIJAY MURUGAPPAAN SUBBIAH (Z1807314)

CSCI 689-1

Fall 2016

Quiz 1
Closed Book and Notes
(10 points)

1. Answer each part of this question.

- a) What characters are used to mark comments in a C++ program?
b) How would you define a constant called CENTIMETERS_PER_INCH with the value 2.54?
c) What is the name of the function that must be defined in every C++ program?
What statement typically appears at the end of that function?
d) What is the purpose endl when you are writing to the cout output stream?
e) List all possible values of type bool.

a) 1. //

2. /*---*/

b)

#define CENTIMETERS_PER_INCH 2.54

c)

main() function, return a end if the type is not void.
for this function main() it cannot be void

d)

It take the cursor to the next line for the next
execution statement.

e)

* true - 0
* false - 1

2. Suppose the input is 5. What is the value of alpha after the following C++ code executes?

```
cin >> alpha;
switch ( alpha ) {
case 1:
case 2:
    alpha += 2;
    break;
case 4:
    alpha++;
case 5:
    alpha *= 2;
case 6:
    alpha += 5;
default:
    alpha--;
}
```

14

- 2) Write a C++ function with the prototype: `unsigned count_vowels (const string& s)`; that can be used to return the number of vowels in the input string `s`.

```
#include <iostream>
// using namespace std;
unsigned count_vowels (const string& s) {
    unsigned count_vowels (const string& s) {
        unsigned counter = 0;
        for (int i=0; i<s.length(); i++) {
            if ((s[i] == 'a' || s[i] == 'e' || s[i] == 'i' ||
                 s[i] == 'o' || s[i] == 'u' || s[i] == 'A' ||
                 s[i] == 'E' || s[i] == 'I' || s[i] == 'O' ||
                 s[i] == 'U') )
                counter += 1;
        }
        return counter;
    }
    int main () {
        cout << "Enter a string";
        string input;
        cin >> input;
        cout << "No. of vowels = " << count_vowels (input);
    }
}
```

no. of hours
amt per hour

2. Write a C++ program that calculates a user's pay. The program reads the number of hours that the user worked and the amount paid per hour from `stdin`, and then computes the total pay for the user and displays it on `stdout`.

```
#include <iostream>
using namespace std;
double calculate-pay (const double & no-of-hours, const double & amt-per-hour);
int main () {
    double no-of-hours;
    double amt-per-hour;
    cout << "Enter No. of hours worked" << endl;
    cin >> no-of-hours; // taking input
    cout << "Enter Wage per hour" << endl;
    cin >> amt-per-hour; // taking input
    cout << "Total amount earned by working" << no-of-hours <<
        "hours for" << amt-per-hour << "hourly wage" <<
        calculate-pay (no-of-hours, amt-per-hour) << endl; // calling
                                                               & displaying
                                                               result
    return 0;
}
double calculate-pay (const double & no-of-hours, const double & amt-per-hour)
{
    return no-of-hours * amt-per-hour; // returns total amount earned
}
```

Z 1819332

NAME: Mohammed Mustafa Shahzad Syed

CSCI 689-3

Quiz 2
Closed Book & Notes

Fall 2017

1. Circle the best answer for each following C++ segments:

a) if (60 <= 12 + 5)
cout << "Hello";
cout << " There ";

outputs the following:

- (i) Hello There (ii) Hello (iii) Hello (iv) There
There

b) if ('a' > 'b' || 'B' > 'A')
cout << "#*#" << endl;

outputs the following:

- (i) #*# (ii) # * (iii) * (iv) none of these
*
#

c) if (7 <= 7)
cout << 6 - 9 * 2 / 6 << endl;

outputs the following:

- (i) -1 (ii) 3 (iii) 3.0 (iv) none of these

d) if (7 < 8) {
cout << "2 4 6 8" << endl;
cout << "1 3 5 7" << endl;
}

outputs the following:

- (i) 2 4 6 8 (ii) 1 3 5 7 (iii) none of these
1 3 5 7

e) if (5 < 3)
cout << "*";
else if (7 == 8)
cout << "&";
else
cout << "\$";

outputs the following:

- (i) * (ii) & (iii) \$ (iv) none of these

Z1819332

NAME: Mohammed Mustafa Shahzad Syed

CSCI 689-3

Quiz 3
Closed Book & Notes

Fall 2017

- 1) Suppose that you have declared and initialized the variables s and t like this:

```
string s = "ABCD", t = "";
```

Given these declarations, what is output of each of the following calls?

- a. cout << s.length (); 4
- b. cout << t.length (); 0
- c. cout << s [2]; C
- d. cout << s.at (2); C
- e. cout << s + t; ABCD
- f. cout << (t += 'a'); a
- g. cout << s.substr (0, 3); ABC
- h. cout << s.substr (4); " "
- i. cout << s.substr (3, 9); D
- j. cout << s.substr (3, 3); DZ

Z1819332

NAME: Mohammed Murtuja Shahzad Syed

CSCI 689-3

Quiz 1
Closed Book & Notes

Fall 2017

1. Assuming x is 5, y is 6, and z is 8, indicate by circling the T or F whether each of the following relational expressions is true or false:

- | | | |
|-------------------|------------------------------------|------------------------------------|
| a) $x == 5$ | <input type="radio"/> T | F |
| b) $7 <= (x + 2)$ | <input checked="" type="radio"/> T | F |
| c) $z < 4$ | <input type="radio"/> T | <input checked="" type="radio"/> F |
| d) $(2 + x) != y$ | <input checked="" type="radio"/> T | F |
| e) $z != 4$ | <input type="radio"/> T | F |
| f) $x >= 9$ | <input type="radio"/> T | <input checked="" type="radio"/> F |
| g) $x <= (y * 2)$ | <input checked="" type="radio"/> T | F |

2. Spherical objects, such as cannonballs, can be stacked to form a pyramid with one cannonball at the top, sitting on top of a square composed of four cannonballs, sitting on top of a square composed of nine cannonballs, and so forth. Write a recursive C++ function `cannonball()` that takes as its argument the height of the pyramid and returns the number of cannonballs it contains. Your function must operate recursively and must not use any iterative constructs, such as while or for.

Q Cannonball (int h)

{
if (h <= 0)
return 0;

else
return ((h * h) + Cannonball(h - 1));

2. The following is a partial definition of a linked-list structure:

```

template < class T > struct Node {
    T data;                                // data value
    Node* next;                             // points to next Node
    Node ( const T& = T( ), Node < T >* = nullptr ); // constructor
};

template < class T > class List {
    Node < T >* head;                      // points to 1st Node
public:
    List();                                 // default constructor
    void insert ( const T& x );             // inserts a Node with data value x
                                              // at the beginning of List
    void print () const;                   // displays elements of List on stdout,
                                              // separated by single spaces
};

```

Show implementations of:

- (a) constructor of the Node structure
- (b) default constructor of the List class
- (c) member function insert () of the List class
- (d) member function print () of the List class

~~A~~

~~2~~

~~L~~

~~Node (const T& = T(), Node < T >* = nullptr); data(T),~~

~~next (Node < T >*)~~

~~2}~~

~~L~~

~~{ head = ~~nullptr~~ T.next;~~

~~void insert (const T& x)~~

~~T.next = head;~~

~~T.data = x;~~

~~d) void print() const~~

~~{~~

~~for (int i = head; i->next != 0; i++)~~

~~cout << *x[i] << " "~~

~~}~~

~~2~~