

Swinburne University of Technology

Faculty of Science, Engineering and Technology

MIDTERM COVER SHEET

Subject Code: COS30008
Subject Title: Data Structures and Patterns
Assignment number and title: Midterm, Solution Design, Design Pattern, and Iterators
Due date: Nov 10, 2023, 23:59
Lecturer: Dr. Van Dai PHAM

Your name: Le Minh Kha**Your student ID:** 104179506

Check Tutorial	Mon 10:30	Mon 14:30	Tues 08:30	Tues 10:30	Tues 12:30	Tues 14:30	Thursday 10:00 (Innovation Lab)
							X

Marker's comments:

Problem	Marks	Obtained
1	68	66
2	120	128
3	56	48
4	70	64
Total	314	306

If you fancy testing out the solution, you could do so by downloading the ZIP file of Vigenere Cipher via my drive link here:

<https://drive.google.com/file/d/12b5874RChcs-1pzgls6ONdMhLIO5S7IG/view?usp=sharing>

KeyProvider.cpp

```
#include "KeyProvider.h"
#include <iostream>          // -2: #include <cctype>
#include <algorithm>

KeyProvider::KeyProvider(const std::string& aKeyword): fKeyword(nullptr), fsize(0),
fIndex(0)
{

    fsize = aKeyword.size();
    //memcpy(fKeyword, aKeyword.c_str(), fsize);

    fKeyword = new char[fsize]; // allocate heap memory the size of fsize

    for (int i = 0; i < fsize; i++)          // Just reuse initialize function
    {
        fKeyword[i] = std::toupper(aKeyword[i]);
    }
    fIndex = 0;
}

KeyProvider::~KeyProvider()
{
    delete[] fKeyword;
}

void KeyProvider::initialize(const std::string& aKeyword)
{
    delete[] fKeyword;

    fsize = aKeyword.size();

    fKeyword = new char[fsize];

    for (int i = 0; i < fsize; i++)
    {
        fKeyword[i] = std::toupper(aKeyword[i]);
    }
    fIndex = 0;
}

char KeyProvider::operator*() const
{
    return fKeyword[fIndex];
};

KeyProvider& KeyProvider::operator<<(char aKeyCharacter)
{
    if (std::isalpha(aKeyCharacter))
    {
```

```

        // Convert the input character to uppercase and store it in the keyword
array    fKeyword[fIndex] = std::toupper(aKeyCharacter);

        // Modulus to wrap it around if it reaches fsize
        fIndex = (fIndex + 1) % fsize;
    }
    return *this;
}

```

Vigenere.cpp

```

#include "Vigenere.h"
#include <string>

// -2 #include <cctype>

void Vigenere::initializeTable()
{
    for (char row = 0; row < CHARACTERS; row++)
    {
        char lChar = 'B' + row;
        for (char column = 0; column < CHARACTERS; column++)
        {
            if (lChar > 'Z')
                lChar = 'A';
            fMappingTable[row][column] = lChar++;
        }
    }
}

Vigenere::Vigenere(const std::string& aKeyword) : fKeyword(aKeyword),
fKeywordProvider(aKeyword)
{
    initializeTable();
}

std::string Vigenere::getCurrentKeyword()
{
    std::string result;

    for (size_t i = 0; i < fKeyword.length(); i++)
    {
        result += *fKeywordProvider;
        fKeywordProvider << *fKeywordProvider;
    }

    return result;
}

void Vigenere::reset()
{
    fKeywordProvider.initialize(fKeyword);
}

```

// my approach is uppercasing the input, encode and return the lowercase version

```

// well the original state of a character should also be checked to return correct
case
char Vigenere::encode(char aCharacter)
{
    char upperChar = static_cast<char>(toupper(aCharacter));

    // only perform the encryption if its a valid character.
    if (isalpha(upperChar))
    {
        // Get the encoded character from the mapping table using the current
keyword character and the input character
        char encodedChar = fMappingTable[*fKeywordProvider - 'A'][upperChar -
'A'];

        // Append the input character to the keyword provider
fKeywordProvider << upperChar;

        // Check if the OG character was lowercase
        if (islower(aCharacter))
        {
            // If the OG character was lowercase, convert the encoded
character to lowercase
            return char(tolower(encodedChar));
        }
        else
        {
            // leave the encoded character as uppercase
            return encodedChar;
        }
    }
    else
    {
        return aCharacter;
    }
}

// similar to encode
char Vigenere::decode(char aCharacter)
{
    if (isalpha(aCharacter))
    {
        char upperChar = static_cast<char>(toupper(aCharacter));
        char decoded = 0;

        for (char col = 0; col < CHARACTERS; col++)
        {
            if (fMappingTable[*fKeywordProvider - 'A'][col] == upperChar)
            {
                decoded = char(col + 'A');
                break;
            }
        }
        fKeywordProvider << decoded;

        if (islower(aCharacter))
        {

```

```

        return char(tolower(decoded));
    }
    else
    {
        return decoded;
    }
}
else
{
    return aCharacter;
}
}

```

iVigenereStream.cpp

```
#include "iVigenereStream.h"
```

```

iVigenereStream::iVigenereStream(Cipher aCipher, const std::string& aKeyword, const
char* aFileName)
    :fIStream(std::ifstream()), fCipherProvider(Vigenere(aKeyword)),
fCipher(move(aCipher))
{
    if (aFileName != nullptr)
        open(aFileName);
}

iVigenereStream::~iVigenereStream()
{
    close();
}

void iVigenereStream::open(const char* aFileName)
{
    fIStream.open(aFileName, std::ios::binary);
}

void iVigenereStream::close()
{
    fIStream.close();
}

void iVigenereStream::reset()
{
    fCipherProvider.reset();
    seekstart();
}

bool iVigenereStream::good() const
{
    return fIStream.good();
}

bool iVigenereStream::is_open() const
{
    return fIStream.is_open();
}

```

```

bool iVigenerStream::eof() const
{
    return fIStream.eof();
}

// -8: before doing this, need to check
// fIStream.get(lCharacter);
iVigenerStream& iVigenerStream::operator>>(char& aCharacter)
{
    aCharacter = fCipher(fCipherProvider, static_cast<char>(fIStream.get()));
    return *this;
}

```

VigenerForwardIterator.cpp

```

#include "VigenerForwardIterator.h"

VigenerForwardIterator::VigenerForwardIterator(iVigenerStream& aIStream)
    :fIStream(aIStream), fCurrentChar(0), fEOF(aIStream.eof())
{
    if (!fEOF) // -4: ++(*this);
        fIStream >> fCurrentChar;
}

// overload the dereference operator to get value iterator is pointing at
char VigenerForwardIterator::operator*() const
{
    return fCurrentChar;
}

// overload the pre-fix increment operator (++i)
// returns the original value before incrementing
VigenerForwardIterator& VigenerForwardIterator::operator++()
{
    fIStream >> fCurrentChar;
    fEOF = fIStream.eof();
    return *this;
}

// also increment overload, but for the post-fix increment (i++)
// returns the updated value
VigenerForwardIterator VigenerForwardIterator::operator++(int)
{
    VigenerForwardIterator tmp = *this;
    ++(*this);
    return tmp;
}

// compare two iterators to see if they're pointing to the same position or not.
// {
bool VigenerForwardIterator::operator==(const VigenerForwardIterator& aOther)
const
{
    return (&fIStream == &aOther.fIStream) && (fEOF == aOther.fEOF);
}

bool VigenerForwardIterator::operator!=(const VigenerForwardIterator& aOther)
const
{

```

```

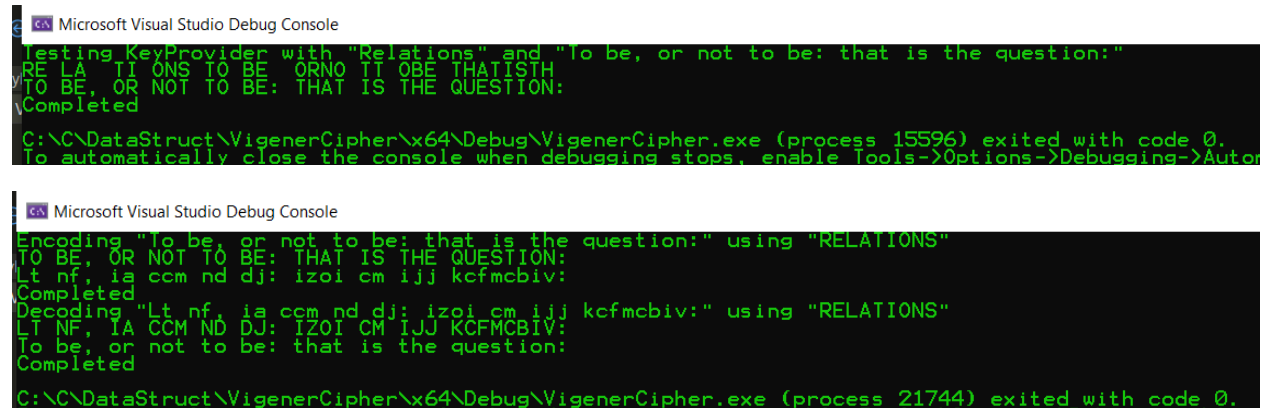
        return !(*this == aOther);
    }
    // }

//returns an iterator pointing to the beginning of the stream
VigenerForwardIterator VigenerForwardIterator::begin() const
{
    //copy the current iterator using the copy constructor
    VigenerForwardIterator lResult = *this;
    lResult.fIStream.reset();
    lResult.fEOF = lResult.fIStream.eof(); // end of the stream has been reached
    if (!lResult.fEOF)
        lResult.fIStream >> lResult.fCurrentChar; // else reads the first
character
    return lResult;
}

//returns an iterator pointing to the end of the stream
VigenerForwardIterator VigenerForwardIterator::end() const
{
    VigenerForwardIterator lResult = *this;
    lResult.fEOF = true;
    return lResult;
}

```

Output console, might as well:



```

Microsoft Visual Studio Debug Console
Testing KeyProvider with "Relations" and "To be, or not to be: that is the question:"
RE LA TI ONS TO BE ORNO TI OBE THATISTH
TO BE, OR NOT TO BE: THAT IS THE QUESTION:
Completed
C:\C\DataStruct\VigenerCipher\x64\Debug\VigenerCipher.exe (process 15596) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Auto

Microsoft Visual Studio Debug Console
Encoding "To be, or not to be: that is the question:" using "RELATIONS"
TO BE, OR NOT TO BE: THAT IS THE QUESTION:
Lt nf, ia ccm nd dj: izoi cm ijj kcfmcbiv:
Completed
Decoding "Lt nf, ia ccm nd dj: izoi cm ijj kcfmcbiv:" using "RELATIONS"
LT NF, IA CCM ND DJ: IZOI CM IJJ KCFMCBIV:
To be, or not to be: that is the question:
Completed
C:\C\DataStruct\VigenerCipher\x64\Debug\VigenerCipher.exe (process 21744) exited with code 0.

```

Decoding "sample_3.txt" using "Relations".

ACT I

SCENE I

London. A Street.

Enter Gloucester.

Gloucester. Now is the winter of our discontent
 Made glorious summer by this sun of York;
 And all the clouds that lour'd upon our house
 In the deep bosom of the ocean buried.
 Now are our brows bound with victorious wreaths;
 Our bruised arms hung up for monuments;
 Our stern alarums changed to merry meetings;
 Our dreadful marches to delightful measures.
 Grim-visag'd war hath smooth'd his wrinkled front;
 And now, - instead of mounting barbed steeds,
 To fright the souls of fearful adversaries, -
 He capers nimbly in a lady's chamber
 To the lascivious pleasing of a lute.
 But I, that am not shap'd for sportive tricks,
 Nor made to court an amorous looking-glass;
 I, that am rudely stamp'd, and want love's majesty
 To strut before a wanton ambling nymph;
 I, that am curtail'd of this fair proportion,
 Cheated of feature by dissembling nature,
 Deform'd, unfinish'd, sent before my time
 Into this breathing world, scarce half made up,
 And that so lamely and unfashionable
 That dogs bark at me, as I halt by them;
 Why, I, in this weak piping time of peace,
 Have no delight to pass away the time,
 Unless to see my shadow in the sun
 And descant on mine own deformity:
 And therefore, since I cannot prove a lover,
 To entertain these fair well-spoken days,
 I am determined to prove a villain,
 And hate the idle pleasures of these days.
 Plots have I laid, inductions dangerous,
 By drunken prophecies, libels, and dreams,
 To set my brother Clarence and the king
 In deadly hate the one against the other:
 And if King Edward be as true and just
 As I am subtle, false, and treacherous,
 This day should Clarence closely be mew'd up,
 About a prophecy, which says, that G
 Of Edward's heirs the murderer shall be.
 Dive, thoughts, down to my soul: here Clarence comes.

Brother, good day: what means this armed guard
 That waits upon your Grace?

Completed.

C:\C\DataStruct\VigenerCipher\x64\Debug\VigenerCipher.exe (process 21800) exited with code 0

Forward Iterator Decoding "sample_4.txt" using "Relations".

SHF J

MLTBX J

Oiwwrs. 0 Xcdtsx.

Tbuxl Yqtohsmywy.

Sajxhxmywy. Zdr lx mby owcqnk in tra rcxudtizfx
 Vigt uftfcbvw xbybwa qt madf fz f qe Swad;
 Tir prk izp dzsvpe nptw xdpv'w oxph ajm zspit
 Wc ozm szxu kcmwr sk yxg dvtnc hozntg.
 Spk cmw xzv pglxk gdpff oxqa xxxhsorico fuyplqh;
 Jno twvcaxs vjok cdgl yq xbk ujbhbutik;
 Bif ngjfh tavingx uvbzhwy gh pmsff riyivsyk;
 Nhw ilnokyyg eejhiiy oa qfdlommzjp rjzclkn.
 Senn-odkfz'k oje djmi zqlplp'e bql jgxhspmm yogwh;
 Lzi ruo, - xbmuser dc vcnhyjrv hngwsx bhlgk,
 Vt jkclmx mbt ygsa iz njtgage pjbjskgmufw, -
 Dj utgwax gqremo nf t zjqae bqonnfv
 Sh wpf ycxuckcwe qehjorcb hv m qvmn.
 Pbi 0, utvp fo iic mpbj'e scg miwsjmbt lkyrcm,
 Wkw gsmh eh qdmeu es ubrgjmm mcpxxfv-bemhh;
 T, cvha mn knmytz munek'h, fzc puog bsws'w jbxyeiu
 Yh fubzm vdzdky s rupyuc srcijba cmncj;
 U, cvhh zz scanijf'e bi ozct ommg vlywxurgd,
 Uwuplym dt imfuowi qe jntmzegnhp gtyhtq,
 Msmcsg'y, msjntxku'h, nstc pnywvx rm nkrk
 Xfyb sbrf gasuibqwx ygwmx, aloyzt zmpy pbvh zx,
 Bzj gley ne moqytz ugs gosfegjcrvprf
 Mpji rpie guzl ux bl, tu J zlmn od uanu;
 Xts, K, hh bmv edjt dcxrgd yjxu xv yshwn,
 Ufkk dt ehqghdy hd tfeb heus ixf mbnb,
 Vmfmxm cb xzs yd laushb na sam tyc
 Xwr xmxvvbu cr qngh pkh sssxfrxqm:
 Esj izrayepfi, mqsuj 0 rssgxx swxyf o zdpuj,
 Ik joftnyscc yvyxw zbrf qmqe-xvptwk imkl,
 Y px isxfqfrorh yi ujbes f zcabsxj,
 Foz qmff cvf whtf jqvixdvqx er yixnw itny.
 Jttmx lbux Y xpcw, qozzlfjxrb reijyadil,
 Fz ryzfzzg rqshcsnnsi, dxrmqv, jsw pagfyl,
 Uc wil rz okiiiajl Pkcjthkj sqp uzj yls
 Wr xmfoum ojhi yii amm bafcvxi hmf vuqsk:
 Uhl nu Eqsy Frajxo ks hx xovw epi knml
 Vx J oq cpunmx, obylz, chp yxfmvffsok
 Yila isn naicuw Gmzkmcxq goatwqm ej pql'w zb,
 Zdthy x tmeqjtxs, xxarx afbr, qpjw 0
 Hg Dwqisx'z wknvp uzi fcwmwkyz xuvdp gw.
 laom, utawllco, iiec ov us lsji: vygr Bepmqvhw hryfk.

Gfrykte, lhqv sug: bzhi bieor qpjm nwnsw acjke

Lufx dvjlw oxph vpdL Zmqrs?

Completed.

C:\C\DataStruct\VigenerCipher\x64\Debug\VigenerCipher.exe (process 1608) exited with code 0.
 To automatically close the console when debugging stops, enable Tools->Options->Debugging->A
 Press any key to close this window . . .