5/5

Name: Dangnhi Ngo ID# 01553277

COMPUTING IV

POP QUIZ 1

You have 10 minutes to complete the quiz. The total number of points is 5. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read.

Write a recursive void function that has one parameter which is a positive integer. When called, the function writes its argument to the screen backward. That is, if the argument is 1234, it outputs the following to the screen: 4321

```
void recursion (int number){
    if (number <= 0){
        return ;
    }
    else {
      int rem = number % 10;
      printf(rem);
    }
    int result = number / 10;
    return recursion (result);
}
```

Name Dangnhi Ngo ID 01553277

COMPUTING IV

POP QUIZ 2

8

You have 15 minutes to complete the quiz. The total number of points is ~~5~~. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read.

1. **(2 points) What the output of the following programs? *Explain*.**

```
#include <iostream>
/*
Assume:
  the following size
        int - 4 bytes
        pointer 4 bytes
        char 1 byte
  no alignment in following classes
*/
class base {
   int arr[10];
};

class b1 : public base { };

class b2 : public base { };

class derived : public b1, public b2 {};

int main(void)
{
   std::cout << sizeof(derived);
   return 0;
}
```

Output: 32 x 80 -1.5

Because: sizeof(b1) = 4 + 4 + 10 = 18

sizeof(b2) = 4 + 4 + 10 = 18

Therefore, sizeof(derived) = 18 + 18 - 4 = 32

b1 & b2 are ~~called~~ both inherited from the base class (so there are 2 copies of the class base). This wastes space which is why we use virtual base class

Name: Dangnhi Ngo ID# 01553277

COMPUTING IV

POP QUIZ 3

You have 15 minutes to complete the quiz. The total number of points is 5. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read.

5

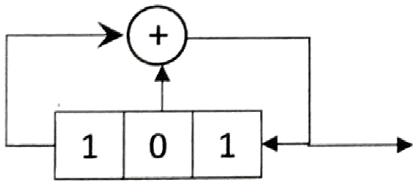
Consider a Linear Feedback Shift Register (LFSR)

1. (1 point) What is the max possible length of the sequence of random numbers that 16 bit LFSR register can generate?

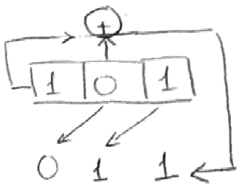
   Max possible length is:

   $2^{16} - 1$ ✓

2. (4 points) A Linear Feedback Shift Register (LFSR) of length 3 with tap location at bit 2 and initial contents of 101, as depicted below:



   What is the length of the sequence of number generated by the register? Show your work.



101 1
011 1
111 0
110 0
100 1
001 0
010 1
101

Length of the sequence is 7 ✓

Name: Dangnhi Ngo ID# 01553277

COMPUTING IV

QUIZ 1

You have 50 minutes to complete the quiz. The total number of points is 40. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read. Be sure to ***state any assumptions*** on which you have based your answers.

38/40

1. (10 points) Consider the following `Makefile` that may be used for the Sierpinski project. It's displayed with line numbers for reference in the questions that follow.

```
1: all: sierpinski
2:
3: sierpinski: main.o sierpinski.o
4:     g++ main.o sierpinski.o -o sierpinski -lsfml-graphics -lsfml-window
-lsfml-system
5:
6: main.o: main.cpp
7:     g++ -c main.cpp -Wall -Werror -ansi -pedantic
8:
9: sierpinski.o: sierpinski.cpp sierpinski.hpp
10:    g++ -c sierpinski.cpp -Wall -Werror -ansi -pedantic
11:
12: clean:
13:    rm *.o sierpinski *~
```

Assume that the files **`main.cpp`**, **`sierpinski.cpp`**, and **`sierpinski.hpp`** exist in the same directory, and are able to be compiled and linked without errors.

a. (2 points) Identify by line number the command(s) that cause code to be compiled.

Line 7 and line 10

b. (1 points) Identify by line number the command(s) that cause code to be linked.

Line 4

c. (4 points) Assume make clean is typed at the shell. Then, make is typed. What sequence of commands will be carried out at the shell as a result of the make? Write them out fully — do not use line number references.

g++ -c main.cpp -Wall -Werror -ansi -pedantic

g++ -c sierpinski.cpp -Wall -Werror -ansi -pedantic

g++ main.o sierpinski.o -o sierpinski -lsfml-graphics -lsfml-window -lsfml-system

d. (3 points) Now, assume sierpinski.cpp is modified, and make is run again. What commands will be carried out?

g++ -c sierpinski.cpp -Wall -Werror -ansi -pedantic

g++ main.o sierpinski.o -o sierpinski -lsfml-graphics -lsfml-window -lsfml-system

2. (6 points) write a template function sum that can be used to sum up the elements of an array, as long as addition is defined on those elements. For example, given your template function definition, the code

```
int a[] = {1,2,3,4};
float b[] = {1.2,3.4,6.7};
cout << sum(a,4) << endl;        // the parameters are the array
cout << sum(b,3) << endl;        // and the size of the array
```

would print: **10**
11.3

```
template <class T>  // maybe template <typename T>
T sumFunction(T member[], int size)
{
    T
    int sum = 0;
    for (int i = 0, i < size, i++)
    {
        sum += member[i];
    }

    return sum;
}
```

-2

3. (1 points) 8 bits per pixel gives 256 colors ($2^8 = 256$)

4. (2 points) What is a pure virtual function?

Pure virtual function is a virtual function which if the derived class is not abstract, pure virtual function is implemented by a derived class.

5. (4 points) Explain why and when do we need use protected instead private.

Because Protected makes class inaccessible from outside, but all derived classes of that class can get inherit. Moreover, most data members of base class are inherited in derived class, except Private. Private methods can not be inherited.

Protected can be accessed in base classes, derived classes and classes in the same package.

Protected has intermediate level of accessibility between public and private.

6. (5 points) What the output of the following programs? *Explain*.

```
/*
Assume the following size
int - 4 bytes
pointer 4 bytes
char 1 byte
*/
```

a. (1 point)

```
#include<iostream>

class ArrayNotFound{
};

int main(){
   ArrayNotFound b;
   std::cout << sizeof(b) << ":" << sizeof(ArrayNotFound)
         << std::endl;
   return 0;
}
```

1 : 1

Because the size of the empty class ArrayNotFound is equal to 1.

b. (4 points)

```
class m{
   int b;   4
   int c;   4
   int d;   4
};

class n : virtual m {};   12 + 4 = 16

class o : m {};     12

class x : n, o {};   12 + 16 = 28

int main(){
   std::cout << sizeof(x) << std::endl;
   return 0;
}
```

(28)

Because sizeof(m) = 4 + 4 + 4 = 12

sizeof(n) = sizeof(m) + virtual (hidden virtual pointer) = 12 + 4 = 16

sizeof(o) = sizeof(m) = 12

sizeof(x) = sizeof(n) + sizeof(o)

= 16 + 12 = 28

SFML event loop

7. **(12 points) The following code is the main.cpp from the SFML Hello World example. It displays a green circle in the SFML window:**

```
1: #include <SFML/Graphics.hpp>
2: int main()
3: {
4:     sf::RenderWindow window(sf::VideoMode(400, 400), "SFML works!");
5:     sf::CircleShape shape(100.f);
6:     shape.setFillColor(sf::Color::Green);
7:     while (window.isOpen())
8:     {
9:         sf::Event event;
10:        while (window.pollEvent(event))
11:        {
12:            if (event.type == sf::Event::Closed)
13:                window.close();
14:        }
15:        window.clear();
16:        window.draw(shape);
17:        window.display();
18:    }
19: }
```

How would you modify this code to hide the green-circle sprite when the "H" key is pressed, and then re-show it when the "S" key is pressed? Please refer to the SFML documentation on the **`Event`** class and the **`KeyEvent`** class in the exam appendix.
You may either write out an entire new main function, or you may write the code you would change or insert. If you are writing out changes/insertions, indicate by line number which lines are replaced by your code or where your code should be inserted.

It's fine if you can't fully figure out the KeyEvent class in order to write perfect code—just indicate approximately what you are trying to do, with comments in English if necessary.

You probably will need to change/insert code in more than one place.
Please write your code on the following page.

SFML event loop (continued)

Please write your code or code changes on this page.

Rewrite from line 15

```
15:     if (sf::Keyboard::isKeyPressed(sf::Keyboard::S)
16:     {
17:         window.draw(shape);
18:     }
19:     else if (sf:Keyboard::isKeyPressed(sf::Keyboard::H)
20:     {
21:         window.clear();
22:     }
23:     window.display();
24: }
25: }
```

Name: Dangnhi Ngo ID# 015532777

COMPUTING IV

QUIZ 2

You have 50 minutes to complete the quiz. The total number of points is 40. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read. ***Be sure to state any assumptions on which you have based your answers.***

Problem Number(s)	**Possible Points**	**Earned Points**
1	20	18
2	5	3
3	4	4
4	6	5
5	5	5
	TOTAL POINTS 40	37

SFML

1. (20 points) **Extending SFML with your own Drawable classes**

For this problem, please refer to the **`LineSegment`** code in the exam appendix. This code was provided as part of the **`Sierpinksi`** triangle assignment.
Using this code as a reference, write code to implement a class called **`Triangle`** that can be drawn in the same fashion as a **`LineSegment`**.
Your Triangle class should be based on **`sf::CircleShape`** (see API reference in exam appendix).
Your class should provide a constructor that accepts one argument: **`float radius`**.
Please write all code that is necessary to accomplish the class so that you could use it (e.g.) as follows. The lines that use your class are highlighted in bold:

```
#include <SFML/Graphics.hpp>
#include "Triangle.hpp"

int main()
{
  sf::RenderWindow window(sf::VideoMode(400, 400), "SFML works!");

  Triangle tri(100.);

  while (window.isOpen())
    {
      sf::Event event;
      while (window.pollEvent(event))
        {
          if (event.type == sf::Event::Closed)
            window.close();
        }

      window.clear();
      window.draw(tri);

      window.display();

    }

}
```

Write your code on the following two pages.

Extending SFML with your own `Drawable` classes (continued)

Code for `Triangle.hpp`:

```
#ifndef TRIANGLE_HPP_
#define TRIANGLE_HPP_
#include <SFML/Graphics.hpp>
#include <SFML/Window.hpp>

class Triangle : public sf::Drawable {
public:
    Triangle (float radius);
    void draw (sf::RenderTarget& target, sf::RenderStates states) const;

private:
    sf::CircleShape triangle;
};
#endif /* TRIANGLE_HPP_ */
```

Code for `Triangle.cpp`:

```
#include "Triangle.hpp"
using namespace sf;

Triangle::Triangle(float radius) {
    triangle.setRadius(radius);
    triangle.setPointCount(3);
}   triangle = CircleShape(radius, 3);

void Triangle::draw(RenderTarget& target, RenderStates states) const {
    target.draw(triangle);
}
```

X -2 use circleShape

18

2. (5 points) **Extending SFML with your own `Drawable` classes**

Short answer questions:

a. (2 points) How would you modify your code if you were to implement your own **`Polygon`** class, where the constructor included a number of **`sides`** in addition to a **`radius`**? Describe what would need to be done; you don't have to write the code to do it. Make sure to describe changes to both your header file (.hpp) and implementation (.cpp).

In the header file (.hpp), the constructor will include the number of sides in addition to a radius. Also, we change the object sf::CircleShape to polygon:

```
public:
    Polygon(int sides, float radius);

private:
    sf::CircleShape polygon;
```

In the implementation (.cpp), we replace the number of setPointCount into "sides" instead of 3

```
Polygon::Polygon(int sides, float radius) {
    polygon.setRadius(radius);
    polygon.setPointCount(sides);
}
```

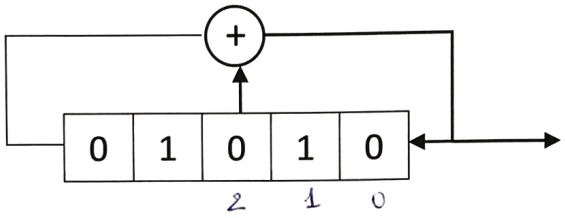
- Constructor takes side as an argument
- Code use that in instantiating CircleShape object

b. (3 points) In C++, what is the proper terminology for referring to the relationship between your `class` and `SFML`? Fill in the blanks below:

My Triangle (name of class) class is a derived (terminology) class of the SFML Drawable (name of class) class.

LFSRs

3. (4 points) A Linear Feedback Shift Register (LFSR) of length 5 with tap location at bit 2 and initial contents of 01010, as depicted below:



The method **`generate()`** takes an integer k as an argument and returns a k-bit integer obtained by simulating k steps of the LFSR.

a. (2 points) What are the ***contents*** of the register after 3 steps? What would be a **value** of output?

Step 0: 0 1 0 1 0 0
Step 1: 1 0 1 0 0 0
Step 2: 0 1 0 0 0 0
Step 3: 1 0 0 0 0
The value of output is 0 (000)

b. (2 points) What is the bit string that the register will output in the course of the first 5 steps, starting with the initial contents of 01010? What would be a value of output?

Step 0: 0 1 0 1 0 0
1: 1 0 1 0 0 0
2: 0 1 0 0 0 0
3: 1 0 0 0 0 1
4: 0 0 0 0 1 0
5: 0 0 0 1 0
The value of the output is 2 (00010)

4. (6 points) **About the N-Body simulation**
Please refer to the details of the N-body physics simulation, provided in the exam appendix, as necessary to answer the following questions.

a. (2 points) **Assume you were performing a simulation with only two bodies, B1 and B2.**
Please assume that calculating the force on B1 due to B2 implies that you've also calculated the force on B2 due to B1.

Per update cycle, how many calculations of the force between the two bodies would need to be computed? ~~1~~ 2

How many calculations of acceleration would need to be computed per update? 2

b. (2 points) **Now assume there are three bodies in the simulation:**

How many force calculations per update? 3

How many acceleration calculations per update? 6

Short answer questions.

c. (1 points) **In doing the object-oriented modeling of the N-body simulation, we stored the mass of each body in the Body class. Why did we do this?**

We stored the mass of each body in the Body class, because it makes easier for us to use the mass in calculating the force between bodies.

d. (1 point) **What are the tradeoffs involved in making the time-step larger or smaller when performing the simulation?**

If the time-step is shorter, the performance of the simulation is larger but more accurate

(5)

5. (5 points) What is the difference between **`friend`** function and **`friend`** class?

The friend function can access the private methods and protected datas of the friend class. Keyword : "friend".

The friend class is the class that all functions inside are friend functions. The friend class can access the private methods and protected datas of the class that it is derived from. Keyword : "friend"

Name: Dangnhi Ngo ID# 01553277

COMPUTING IV

QUIZ 3

You have 50 minutes to complete the quiz. The total number of points is 40. You are not allowed to use any books, notes, calculator, or electronic devices. Write your answer carefully and clearly. Incorrect answers will receive little to no points. When you are asked to write a program the program should be clear and include indentations and comments to make it easier to read. ***Be sure to state any assumptions on which you have based your answers.***

Problem Number(s)	**Possible Points**	**Earned Points**
1	8	8
2	6 + 2 extra	8
3	4	4
4	6	6
5	6	6
6	10	10
	TOTAL POINTS 40	42

DNA sequence alignment.

1. **(8 points)** Aligned two strings: X = "CTACCAGT" and Y = "TACATG"
   You should show the matrix (5 points), compute the edit distance (1 point), and show the path (2 points).

operation	*cost*
insert a gap	2
align two characters that mismatch	1
align two characters that match	0

X\Y	T	A	C	A	T	G	_	_
C	6	8	9	11	12	14	16	
T	4	6	8	9	10	12	14	
A	4	4	6	7	9	10	12	
C	5	3	4	6	7	8	10	
C	6	4	2	4	5	6	8	
A	7	5	4	2	3	4	6	
G	9	7	5	3	2	2	4	
T	10	8	6	4	2	1	2	
_	12	10	8	6	4	2	0	
_								

_ T A C A T G _
C T A C C A G T
2 0 0 0 1 1 0 2 = 6

Edit Distance = 6

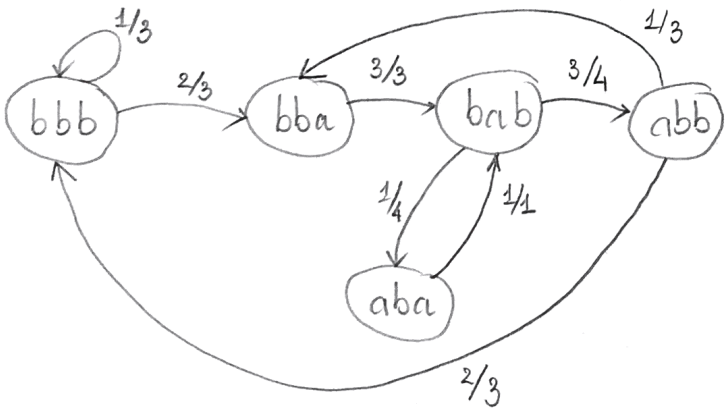
Markov Model of Natural Language

2. **(6 points)**
Suppose the text is "***bbbabbabbbbaba***" and we're using a 3-gram Markov Model. Produce a probabilistic model of the text: given a particular *k-gram* series of letters, what letters follow at what probabilities?
You should show ***k-grams*** (2 points), ***their frequencies*** (1 points), ***frequency of the next character*** (1 points), ***probability of the next character*** (1 points), and a ***possible trajectory through the Markov chain*** (a sequence of such states) (1 points).

(2 extra points) Show the Markov chain (State diagram) with transitional probabilities.

k-gram	frequencies	frequency of the next character		probability of the next character	
		a	b	a	b
bbb	3	2	1	2/3	1/3
bba	3	0	3	0	3/3
bab	4	1	3	1/4	3/4
abb	3	1	2	1/3	2/3
aba	1	0	1	0	1/1
	14	4	10		

bbb → bba → bab → abb → aba



+2

Regular expressions and DFA (*Deterministic finite state automata*)

3. **(4 points)**
For each of the following sets of strings over the alphabet **{*a, b, c*}**, write a regular expression that matches all strings in the set, and choose from among the alternatives below the DFA that accepts that same set (just circle the correct letter):

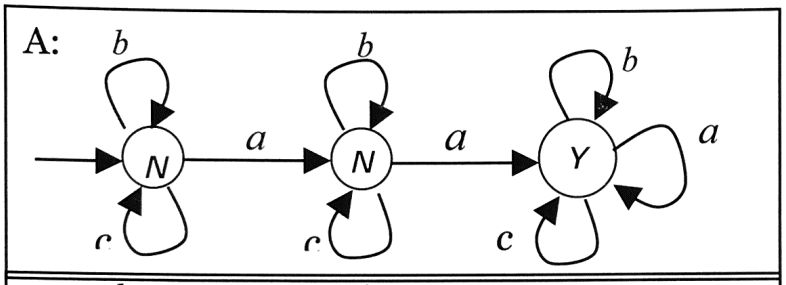
a. (2 points) strings that start and end with ***a*** (including just ***a***)
Regular expression: a | a(a|b|c)*a

DFA: A B C D (E)

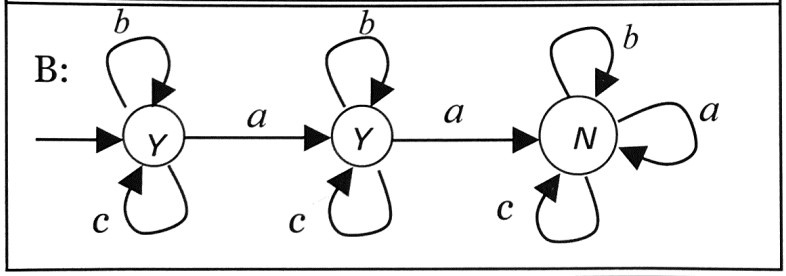
b. (2 points) strings containing at least two ***a***'s
Regular expression: (b|c)* (a(b|c)*a)(a|b|c)*

DFA: (A) B C D E

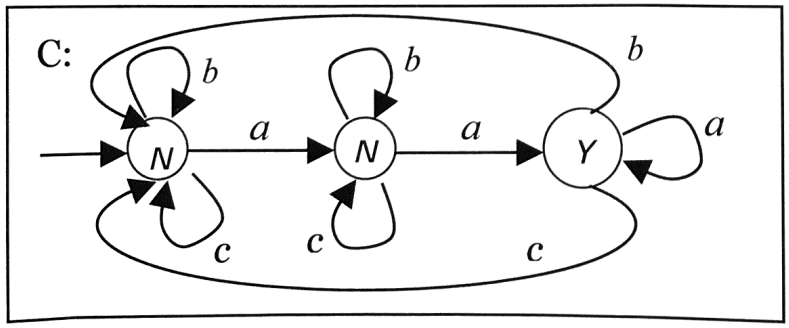
A:



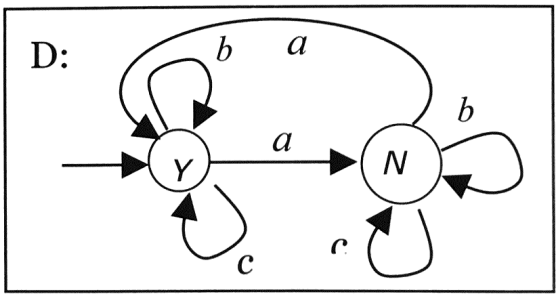
B:



C:



D:



E: none of these

Regular expressions and DFA (*Deterministic finite state automata*)

4. **(6 points)** Which of the following matches regex (*Hint*: there is more than one string):

   a. (3 points) **/[a-zA-Z]*[^,]=/**

   (A) Butt=
   B. BotHEr,=
   C. Ample
   D. FIdDlE7h=
   (E.) Brittle =
   (F.) Other.=

   b. (3 points) **[a-z]+[\.\?!]**

   (A) battle!
   B. green
   (C) swamping.
   D. jump up.
   (E.) undulate?
   F. is.?

5. (6 points) Write a regex that matches
   a. the emails of the form userid@domain.edu. Where ***userid*** is one of more alpha (alphanumeric) characters and the domain is one or more alpha characters.

   `[a-z]+@([a-z]+\.)+edu`

   b. a credit card number. Credit card number is 16 digits, and is typically divided into 4 groups of 4 digits in form of:
   xxxx xxxx xxxx xxxx
   xxxx-xxxx-xxxx-xxxx
   xxxx,xxxx,xxxx,xxxx

   `([0-9]{4}[\s\-\,]){3}[0-9]{4}`

Code optimization

6. (10 points) **Describe** at least 10 ways you can optimize your code.

1/ Move loop inside the function calls

Change: for (i = 0; i < 50; i++) DoSomething()

into : DoSomething() { for (i = 0; i < 50; i++) { } }

2/ Avoid / reduce the number of local variables. Declare local variables inside the inner most scope. Do not switch wholesale to global variables.

3/ Reduce the number of function parameters

The large number of function parameters are expensive due to the large number of function parameters are pushed on stack on each call

4/ Use constructor initialization list

Use: Color:: Color() : r(0), g(0), b(0)

instead of: Color:: Color() { r = g = b = 0; }

5/ Prefer initialization than assignment.

Color c(black); is faster than Color c; c = black;

6/ Prefer int over char or short

7/ Prefer use shift operations << or >> than division or multiplication, where possible

8/ Change if-else into switch case

9/ Prefer use operators +, -, *, / than +=, -=, *=, /=

10/ Prefer preincrement than post-increment; ++i is faster than i++

11/ GNU Compiler: Some options to control optimization: -O, -O1, -O2

It is used to reduce code size and execution time

12/ When working with two dimensional arrays, optimize the algorithms to make the loop iterate over the second index.

7/7

Your Name Dangnhi Ngo Your ID 01553277

1. Which of the following matches regexp **a.[bc]+**
   - (1) abc
   - (2) abbbbbbbb
   - (3) azc
   - (4) abcbcbcbc
   - 5) ac
   - (6) asccbbbbcbcccc

3

2. Which of the following matches regexp **abc|xyz**
   - (1) abc
   - (2) xyz
   - 3) abc|xyz
   - 4) abc xyz

2

3. Which of the following matches regexp **(very)+(fat)?(tall|ugly) man**
   - 1) very fat man
   - 2) fat tall man
   - (3) very very fat ugly man
   - (4) very very very tall man

2

Your Name Dangnhi Ngo Your ID 01553277

Problem 1

Task	Text	
Match	can	[cmf](an)
Match	man	[cmf]an
Match	fan	[^drp]an
Skip	dan	
Skip	ran	
Skip	pan	

Problem 2

Task	Text	
Match	wazzzzzup	wa[z]{2,}up
Match	wazzzup	waz[z]+{3,5}up
Skip	wazup	

Problem 3

Capture	Jan 1987	(...\\s)[\\d]{4}
Capture	May 1969	[A-z][a-z]{2}␣[0-9]{4}
Capture	Aug 2011	

Problem 4

Task	Text
Capture	1280x720
Capture	1920x1600
Capture	1024x768

Name: Dangnhi Ngo ID# 01553277

Problem 1.

Match	aaaabcc	a+[bc]+
Match	aabbbbc	
Match	aacc	
Skip	a	

Problem 2.

Match	1. abc
Match	2. abc
Match	3. abc
Skip	4.abc

Problem 3.

Match	I love cats
Match	I love dogs
Skip	I love logs
Skip	I love cogs

Name: Dangnhi Ngo ID# 01553277

Given string

"g a g g g a g a g g c g a g a a a"

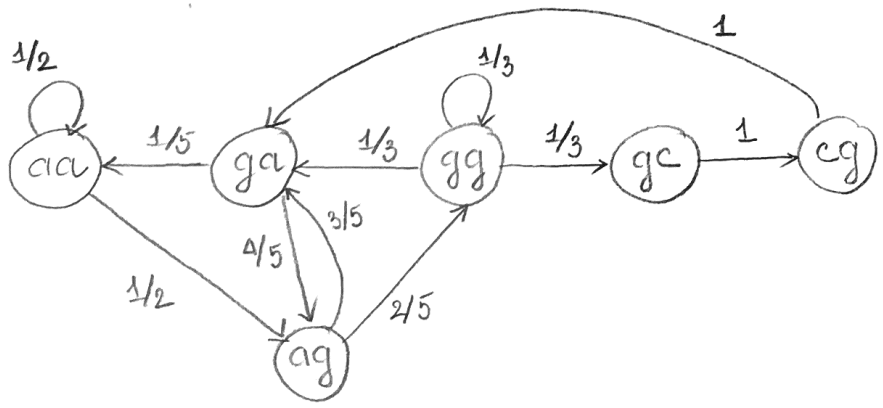
build the Markov chain for $k = 2$. Show kgrams and their frequencies.

ga	5/17
ag	5/17
gg	3/17
gc	1/17
cg	1/17
aa	2/17
g	9/17
a	7/17
c	1/17

ga → g	4/5
ga → a	1/5
gg → g	1/3
gg → c	1/3
gg → a	1/3
ag → a	3/5
ag → g	2/5
cg → a	1
aa → a	1/2
aa → g	1/2
gc → g	1

ga → ag → gg → gg → ga → ag → ga → ag → gg → gc

→ cg → ga → ag → ga → aa → aa → ag



Name Dangnhi Ngo ID 01553277

1. x= "kitten"
   y = "sitting"

		K	I	T	T	E	N	_
S		4	5	7	9	11	12	14
I		5	3	5	7	9	10	12
T		6	4	3	5	7	8	10
T		7	5	3	3	5	6	8
I		8	6	5	3	3	4	6
N		10	8	6	4	2	2	4
G		11	9	7	5	3	1	2
_		12	10	8	6	4	2	0

5/5

KITTEN_
SITTING ✓

1000102 = 4