Unit 1 – ICS4U0 – Java Applets & Strings
## Sample Test – February 14, 2024 – What a nice way to Celebrate Valentine's Day!

Name: _Gorski_____

Total	Knowledge	Communication	Thinking	Application
(100)	(24)	(21)	(21)	(34)

Knowledge

1. This is the original String:
```
String b = "BasketBALL";
```

Match the code with the output.

| 0	1	2	3	4	5	6	7	8	9
B	a	s	k	e	t	B	A	L	L

/10

h	10	a.	System.out.println ((int) b.charAt (1));
f	2	b.	System.out.println (b.charAt (0));
g	0	c.	System.out.println (b.charAt (5));
a	97	d.	System.out.println (b.compareTo ("hockey") < 0);
m	BASKETBALL	e.	System.out.println (b.equals ("hockey"));
L	basketball	f.	System.out.println (b.indexOf ('s'));
i	BasketBAMM	g.	System.out.println (b.indexOf ('B'));
e	false	h.	System.out.println (b.length ());
d	true	i.	System.out.println (b.replace ('L', 'M'));
b	B	j.	System.out.println (b.substring (0, 4));
c	t	k.	System.out.println (b.substring (0, 1) + b.substring (4, 9));
j	Bask	l.	System.out.println (b.toLowerCase ());
k	BetBAL	m.	System.out.println (b.toUpperCase ());
n	BB	n.	System.out.println ("" + b.charAt (0) + b.charAt (0));

2. Use this code to fill in the memory diagram. Then, write out what each substring prints.

/5

```
String f = "cricket and soccer";
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
c	r	i	c	k	e	t		a	n	d		s	o	c	c	e	r
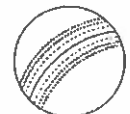
(a)	System.out.println (f.substring (0, 5));	crick
(b)	System.out.println (f.substring (13, f.length ()));	occer
(c)	System.out.println (f.substring (f.length () / 4, f.indexOf (' ')));	ket

18 (above f.length () in (b))

4 (above f.length () / 4 in (c))   7 (above f.indexOf (' ') in (c))

18/4 = 4
chop off decimal   7

3. Write the code required for each colour.   R   G   B   /6

```
blue       title.setBackground(new Color( 0 , 0 , 255 ));
cyan       title.setBackground(new Color( 0 ,255 ,255 ));
green      title.setBackground(new Color( 0 ,255 , 0 ));
yellow     title.setBackground(new Color( 255 ,255 , 0 ));
red        title.setBackground(new Color( 255 , 0 , 0 ));
magenta    title.setBackground(new Color( 255 , 0 ,255 ));
white      title.setBackground(new Color( 255 ,255 ,255 ));
black      title.setBackground(new Color( 0 , 0 , 0 ));
```

4. Fill in this colour wheel based on the computer colours. Use the proper colour names.

/3

Magenta	Red
Blue	Yellow
Cyan	Green
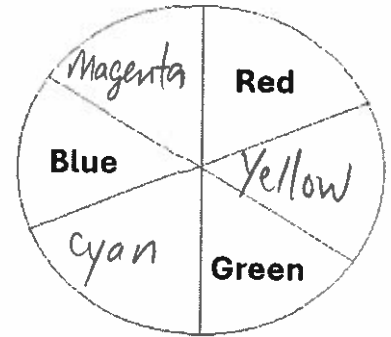
Communication

5. Fill in the words that match the descriptions given.   /10

String	(a) A type of variable that holds a group of chars.
charAt	(b) A String method that returns a char.
Brute Force	(c) An attack that can break a Caesar shift. *Sniffing is the name of the attack for the key distribution problem.*
ASCII	(d) An encoding technique for translating letters to binary.
Pig Latin	(e) Encryption named for an animal and the Ancient Roman language.
Mirror Writing	(f) Encryption that can be decrypted with a shiny reflective surface.
Cipher text	(g) Text that is not easily read.
BlackBox	(h) A type of testing where you don't look at the code.
Casting	(i) The name for using (int) in front of a char to make it into an integer.
JButton	(j) A type of widget used for mouse input.

6. Why is testing important? Provide a specific example. (3 points, at least 3 sentences)   /3

see Sheet 1.6 #8.

1. In 1996, the Ariane 5 ESA rocket exploded 40 seconds after take off, costing $7.5 billion

2. This occured because the average case black box test case for speed was not tested.

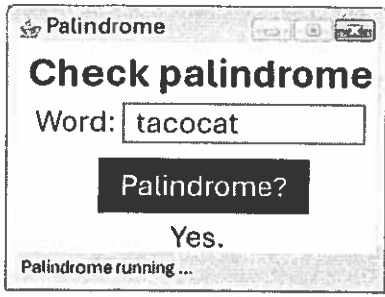3. This example shows us testing errors can be very expensive.

7. Name each PARC principle and explain how each is broken by this app.

Proximity	The similar things (group 1: buttons, group 2: sports equipment) should be together in a group, spaced from other things
Alignment	All items in the screen should be right, left or centre aligned. they are placed randomly instead.
Repetition	① 4 different fonts are used. Only 1 should appear, ② all different styles of pictures are used. ③ all different styles/sizes of buttons.
Contrast	the item which is contrasted is Enter (very large), but it isn't important. the title (choose basketball) should be large.

Chuse the basketball.

Next

PREVIOUS

Enter
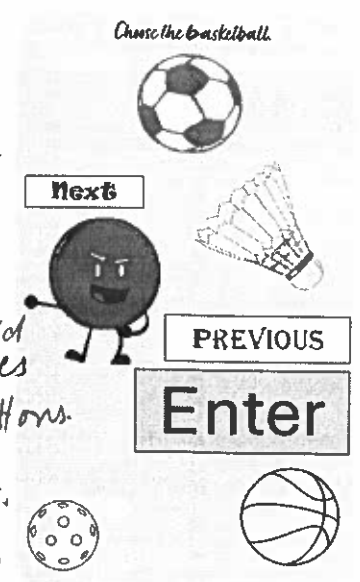
Thinking

8. White Box and Black Box test this app.

Palindrome

Check palindrome

Word: tacocat

Palindrome?

Yes.

Palindrome running ...

A palindrome is a word that reads the same backward as forward.

For example "racecar" or "nurses run".

Word	Black Box Test Case
a	small data
banana	average data
antidisestablishment-arianism.	large data
$!$	boundary case data
Word	**White Box Test Case & Line Number**
[empty string]	avoid loop (#4)
b	loop once (#4)
alligator	loop many times (#4)
tacocat	go into if [is palindrome] #6
ciphertext	go into else [not palindrome] #8

```
1    public void actionPerformed (ActionEvent e){
2         String s = input.getText();
3         String rev = "";

4         for (int i = s.length () - 1 ; i >= 0 ; i--)
5              rev += s.charAt (i);

6         if (rev.equals (s))
7              output.setText ("Yes");
8         else
9              output.setText ("No");
10   }
```

9. Decrypt the following messages. This chart may be useful: /12

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

cbtlfucbmm ofu
basketball net

enozdne llabtoof
football endzone

ockeyhay katessay
hockey skates

3 18 9 3 11 5 20
c r i c k e t

Hint: vgmcfrvgzz = basketball, Hint #2: All punctuation ', . is at it appears.

rgcf g mfgr gke y'zz rfzz opt g rgzf gzz
take a seat and I'll tell you a tale all

gvptr vgzzfr, vgmcfrvgzz gke vgmfvgzz.
about ballet, basketball and baseball.

Application ⇨

10. This is the original String: `String h = "hockey SKATES";`   /10

0	1	2	3	4	5	6	7	8	9	10	11	12
h	o	c	k	e	y		S	K	A	T	E	S

Write the code to produce the following output:

(a)	HOCKEY SKATES	System.out.println(h. toUpperCase ( ));
(b)	hockey skates	System.out.println(h. toLowerCase ( ));
(c)	A	System.out.println(h. charAt (9));
(d)	hockey PKATEP	System.out.println(h. replace ('S', 'P'));
(e)	hockey	System.out.println(h. subString (0, 6));
(g)	13 //the length	System.out.println(h. length ( ));
(h)	104 //ASCII of h	System.out.println((int) h. charAt (0));
(i)	2 //position of c	System.out.println(h. indexOf ('c'));
(j)	STATE	System.out.println(h. subString (7, 12). replace ('K', 'T'));
(k)	ok	System.out.println(h. charAt (1) + h. charAt (3));

11. Make a loop to count number of times `cat` appears in a string. The letters are all lowercase.
(if you can't do that, there are serious part marks for counting the number of times the letter `c` appears)

Also: There is **way too much space** on this page for this question. Don't let that bother you.

/8

Example 1: Word? **catdogcatcatturnip** 3 cat(s)	Example 2: Word? **c** 0 cat(s)	Example 3 Word? **cahatmatsatac** 0 cat(s)	Example 4 Word? **tacocat** 1 cat(s)

```
String s = IO.inputString("Word? ");
```

int count = 0;
for (int i=0; i < s.length()-2 , i++)
{
    if (s.substring(i, i+3).equals("cat")) *
        count ++;
}

System.out.println(count + " cat(s)");

* alternatively:
if (s.charAt(i) == 'c' && s.charAt(i+1) == 'a' &&
    s.charAt(i+2) == 't')

12. (a) Fill in init for the Encryption App (shown in on the right)

```
import java.awt.*; import javax.swing.*; import java.applet.Applet; import java.awt.event.*;
public class Encrypt extends Applet implements ActionListener
{
    JTextField word;

    //The other global variable:
    JLabel    (X)    ;

    public void init ()
    { resize (300, 100);
    //The first label: (Font is Arial, Font.BOLD and 30 pt)
    JLabel title = new JLabel ("Encryption");
    title.setFont (new Font ("Arial", Font.BOLD, 30));

    //The prompt
    JLabel pmt = new JLabel ("Word:");

    //The textfield:
    word = new JTextField (8);

    //The button: (Black background, white writing)
    JButton b = new JButton ("Encrypt It!");
    b.setBackground (Color.black);
    b.setForeground (Color.white);
    b.addActionListener (this);
    b.setActionCommand ("(Y)");

    //The last label:
    (X) = new JLabel ("Enter a word");

    //add the widgets:
    add ( title );
    add ( pmt );
    add ( word );
    add ( b );
    add ( (X) );

    } //init
}
```
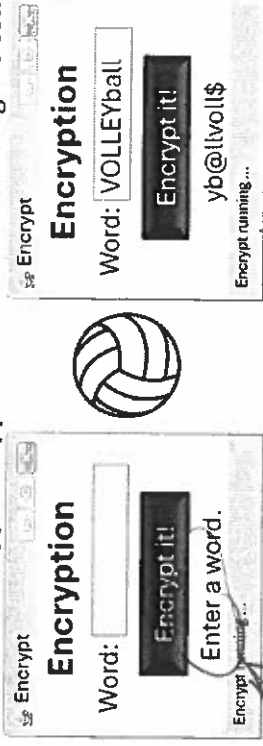
Note (X) and (Y) can be anything but they must match.

(b) Fill in the Encryption App's actionPerformed using the comments.


```
public void ActionPerformed (ActionEvent e)
{
    if(e.getActionCommand().equals("(Y)"))
    {
        String s = word .getText();

        //switch it to lowercase
        s = s.toLowerCase();

        //replace all 'a' with '@'
        s = s.replace('a', '@');

        //replace all 'e' with '$'
        s = s.replace('e', '$');

        //chop the string in half
        String first = s.substring(0, s.length()/2);
        String last = s.substring(s.length()/2, s.length());

        //switch the two halves
        s = last + first;

        //output on the screen
        (X) .setText(s);

    } //if
} //actionPerformed
} //Applet
```