CAMPEAU TEST
MULTIPLE CHOICE QUESTIONS

PART B – FOCUS ON OOP (PLUS STRINGS & ARRAYLISTS)

This is a simple Multiple Choice test to prepare you for the Multiple Choice Questions on the Computer Science AP exam.

Please note that many of these questions are easier than the ones on the AP exam in order to help you remember different concepts.

Version 0.4 (about 40% done)

Q-What will the following code print out to screen?

```
public class Point
{
	public int x;
	public int y;
}
```

```
public class CampeauTest
{
	public static void main(String[] args)
	{
		Point p = new Point();
		p.x = 4;
		p.y = 8;

		Point q = p;
		q.x = 7;
		q.y = 3;

		int xy = p.x + p.y;
		System.out.println(xy);
	}
}
```

A) 10
B) 11
C) 12
D) 48
E) 73

SOLUTION

Answer A) 10

EXPLANATION

After Point q is set to p, both p and q are references to the same object in memory. So q.x is in fact accessing the same memory location as p.x. So setting q.x to 7 and q.y to 3 is in fact doing the same thing as if we were setting p.x and p.y. So, xy gets the total of 7 + 3.
Topics: OOP, References

Q-The Sock class below needs a constructor so that the code works. Which constructor should be added?

```
public class Sock
{
     public int size;
     public String colDesc;    //colour description
}
```

```
public class CampeauTest
{
     public static void main(String[] args)
     {
          Sock s = new Sock(9, "blue & green");
     }
}
```

A)
public Sock(int s, String c)
{
     s=size;
     c=colDesc;
}
	B)
	public static Sock()
	{
	     size=9;
	     colDesc="blue & green";
	}

C)
public Sock()
{
     size=9;
     colDesc="blue & green";
}
	D)
	public Sock(int s, String c)
	{
	     size=s;
	     colDesc=c;
	}

SOLUTION

Answer D)

EXPLANATION

Let’s start by eliminating some of the possible answers. The answer B has the word static. Constructors are not static. B is incorrect.

The answer C has no parameters in the header and therefore doesn’t match the constructor call in the main. So C cannot be the correct answer.

Both answers A and D have the correct parameter list. However, A incorrectly sets the value of the instance variables to the local variables in the constructor. So A is incorrect. This leaves us with D which is correct.

Q-The Sock class below needs a constructor so that the code works. Which constructor should be added?

```
public class Sock
{
	public int size;
	public String colDesc;   //colour description
}
```

```
public class CampeauTest
{
	public static void main(String[] args)
	{
		Sock s = new Sock(9, "blue & green");
	}
}
```

```
A)
public Sock(int size, String colDesc)
{
	size=size;
	colDesc=colDesc;
}
```

```
B)
public Sock(int size, String colDesc)
{
	size=this.size;
	colDesc=this.colDesc;
}
```

```
C)
public Sock(int size, String colDesc)
{
	this.size=size;
	this.colDesc=colDesc;
}
```

```
D)
public Sock(int size, String colDesc)
{
	this.size=this.size;
	this.colDesc=this.colDesc;
}
```

SOLUTION

Answer C)

EXPLANATION

Let's start by eliminating some of the possible answers. The answer A sets both local variables to themselves. Because the instance variables and the local variables are the same, we need to use the *this* keyword.

The answer C has no parameters in the header and therefore doesn't match the constructor call in the main. So C cannot be the correct answer.

Both answers A and D have the correct parameter list. However, A incorrectly sets the value of the instance variables to the local variables in the constructor. So A is incorrect. This leaves us with D which is correct.
Topics: this, OOP, constructors

Q-What will the following code print out to screen?

```java
public class Ticker
{
	private int time;

	public Ticker()
	{
		time = 0;
	}

	public void inc()
	{
		time++;
	}

	public void dec()
	{
		time--;
	}

	public int getTime()
	{
		return time;
	}
}
```

```java
public class CampeauTest
{
	public static void main(String[] args)
	{
		Ticker t = new Ticker();
		t.inc();
		t.inc();
		t.inc();
		t.dec();
		t.inc();
		System.out.println(t.getTime());
	}
}
```

A) Ticker
B) 0
C) 3
D) 4
E) 5

SOLUTION

Answer C) 3

EXPLANATION

First, one should look at the Ticker class. It has one instance variable named time that gets set to zero in the constructor. The method inc() increases time by one and the method dec() decreases time by one.

Now, in main, we create Ticker object named t. Its time is initially zero. We increment it 4 times and decement it 1 time. So, t's time is three.
Topics: OOP, Method calls

Q-A tailoring company offers many clothing altering services. One of its services is to replace waist elastics in pants. Every day, the company gets a shipment of pants that required the waist elastic replaced.

Below, you will find a Pants class that represents Pants sent to the company. The function totalElastic uses an array of Pants objects (which represents a daily shipment of pants) and calculates the minimum amount of elastic material length required.

Which implementation of the function is correct?

```
public class Pants
{
      private int length;
      private int waist;
      private String description;
      //constructor not shown
      //get methods not shown
}
```

```
public static int totalElastic(Pants[] p)
{
      //choose implementation below
}
```

A)

```
int tot = 0;
for (Pants pant: p)
{
      tot += pant.getWaist();
}
return tot;
```

B)

```
int tot = 0;
for (Pants pant: p)
{
      tot += p.getWaist();
}
return tot;
```

C)

```
int tot = 0;
for (Pants pant: p)
{
      tot += getWaist();
}
return tot;
```

D)

```
int tot = 0;
for (Pants pant: p)
{
      tot += Pants.getWaist();
}
return tot;
```

SOLUTION

Answer A)

EXPLANATION

After understanding that we need to total the waist value for each pair of pants, we need to understand the enhanced for loop and the assignment statement inside of it.

In B, we call getWaist() on the array name. That is not correct. In C, we call getWaist without an object name. That is not correct. In D, we call getWaist() with the class name, which is also no correct because this is an instance method. In A, we call getWaist() on the object name which his correct. So A.
Topics: OOP, Enhanced for loop, method calls, object arrays

Q-Which of the statements in the main function will NOT give an error?

```
public class SomeObject
{
        public int a;
        public int b;
        public int c;

        public SomeObject(int ta, int tb)
        {
                a = ta;
                b = tb;
                c = a + b;
        }
}
```

```
public class CampeauTest
{
        public static void main(String[] args)
        {
                //A
                SomeObject obj = new SomeObject();
                //B
                SomeObject obj = new SomeObject(1);
                //C
                SomeObject obj = new SomeObject(1, 2);
                //D
                SomeObject obj = new SomeObject(1, 2, 3);
        }
}
```

A) A
B) B
C) C
D) D
E) They will all give errors.

SOLUTION

Answer C) C

EXPLANATION

The key is to remember that the number of parameters must match the number of parameters in the constructor. Since the construct has two parameters, so must our statement to create the object.

Note that some people may have chosen D thinking it matched up with the number of instance variables but that is not the correct approach though it often is true for simpler classes.

Topic: OOP, Constructors

To add in Part B:
ArrayLists
Strings
Inheritence
extends

Comparable