Practical 4 Answer Guidelines

Part A (Understanding Concepts)

1. (a) main – none
instruct – none
find_velocity – 4 parameters of type double

(b) main – int
instruct – none (void)
find_velocity – double

(c) Yes, valid.

(d) No, invalid because the function instruct() does not return any value to display using cout.

(e)
```
/*
 * Computes the average velocity of a particle travelling on a line
 * between points p1, and p2 in time t1 to t2.
 * Pre:      p1, p2, t1, and t2 are defined.
 *           p1 < p2 and t1 < t2.
 */
```

2.

(a)
```
void func1(int x);
```

(b)
```
int func2(int x, int y)
```

3.

(a) Only 2 actual parameters allowed since function definition has only 2 formal parameters.

(b) Should be func(x, y)

4.

(a) In main: a, b In funcA: a In funcB: a, r

(b) The concept of scope is applied.
The variable a in function main is only visible from the its declaration to the end of the function.
Similarly, the variables a in function funcA and funcB are only visible from its declaration to the end of the function.

(c)

No.	Statement executed	Data Area for function		
		main	**funcA**	**funcB**
1.	In main: `funcA()`	a ? b ?	a ?	
2.	In funcA: `cin >> a;`	a ? b ?	a 23	
3.	In funcA: `return a;` In main: `a = funcA();`	a 23 b ?		
4.	In main: `funcB(a)`	a 23 b ?		x 23 a ? r ?
5.	In funcB: `a = x / 10;`	a 23 b ?		x 23 a 2 r ?
6.	In funcB: `r = x – a * 10;`	a 23 b ?		x 23 a 2 r 3
7.	In funcB: `return r;` In main: `b = funcB(a);`	a 23 b 3		

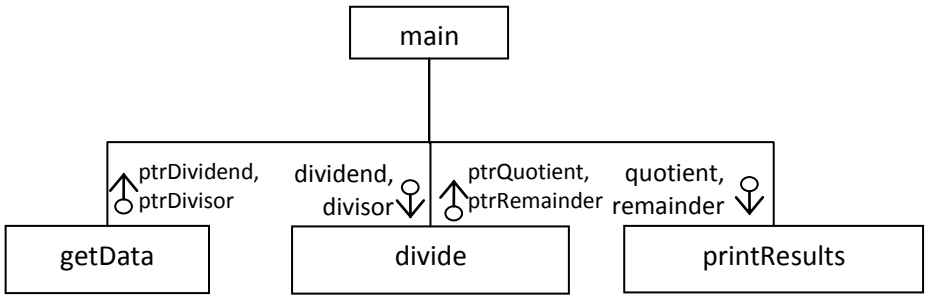
5.

No.	Statement executed	Variables						Output
		a	b	c	d	r	s	

1.	a=1;
	b=2;
	c=3;
	d=4;
		1	**2**	**3**	**4**	?	?	r = 3
								s = 10
								result = 13

2.	r = strange (a, b);	1	2	3	4	**3**	?	
3.	cout << "r = " << r << endl;	1	2	3	4	3	?	
4.	s = strange (r, strange(c, d));	1	2	3	4	3	**10**	
5.	cout << "s = " << s << endl;	1	2	3	4	3	10	
6.	cout << "result = " << strange (r, s) << endl;	1	2	3	4	3	10	

6.



Part B (Programming Exercises)

1.

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

double find_velocity(double p1, double p2, double t1, double t2);

int main(void)
{
       double p1, p2, t1, t2;

       cout << "Enter point1 and point2: ";
       cin >> p1 >> p2;
       cout << "Enter time1 and time2: ";
       cin >> t1 >> t2;

       cout << "Average velocity is " << fixed << setprecision(2)
            << find_velocity(p1, p2, t1, t2) << endl;


   return 0;
}

double find_velocity(double p1, double p2, double t1, double t2)
{
       return (p2 - p1) / (t2 - t1);
}
```

2. (a) output: a = 5, b = 3
(b) output: a = 3, b = 5

3.

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

void getData(int* ptrDividend, int* ptrDivisor);
void divide(int dividend, int divisor, int* ptrQuotient, int* ptrRemainder);
void printResults(int quotient, int remainder);

int main(void)
{
	int dividend, divisor, quotient, remainder;

	getData(&dividend, &divisor);
	divide(dividend, divisor, &quotient, &remainder);
	printResults(quotient, remainder);

	return 0;
}

void getData(int* ptrDividend, int* ptrDivisor)
{
	int dividend, divisor;

	cout << "Enter the dividend and divisor: ";
	cin >> dividend >> divisor;

	*ptrDividend = dividend;
	*ptrDivisor = divisor;

	return;
}

void divide(int dividend, int divisor, int* ptrQuotient, int* ptrRemainder)
{
	*ptrQuotient = dividend /divisor;
	*ptrRemainder = dividend % divisor;

return;
}

void printResults(int quotient, int remainder)
{
	cout << "The quotient is " << quotient << endl;
	cout << "The remainder is " << remainder << endl;

	return;
}
```