**Practical 3 Answer Guidelines**

**Part A (Understanding Concepts)**

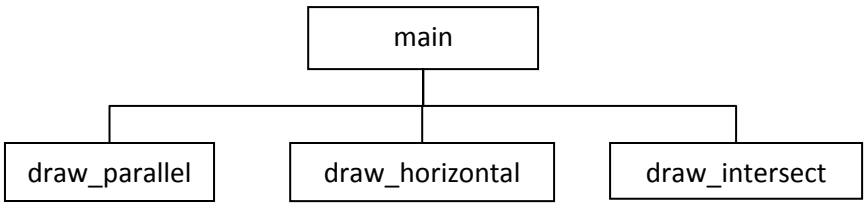
1. (a) [0-5]
   (b) [0-5] + 1 = [1-6]
   (c) [0-5] + 11 = [11-16]

2.
(a) line 4
(b) line 8
(c) lines 12-15

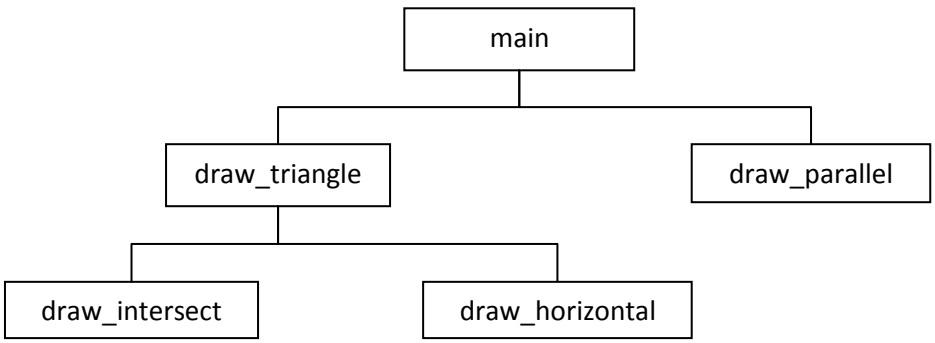
3.
(a) main, func1, func2
(b) The reserved word 'void' indicates that the functions main and func1 do not have any parameters.
(c) Formal parameter - num in line 5 and 24.
Actual parameter - 123 in line 13.
- number in line 15.
(d) output: ABC123
xyz-6789

4.
(a) function call
(b) function prototype
(c) function prototype
(d) function call

5.
(a)

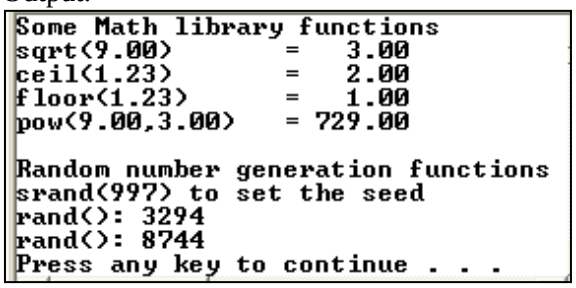



(b)



**Part B (Programming Exercises)**

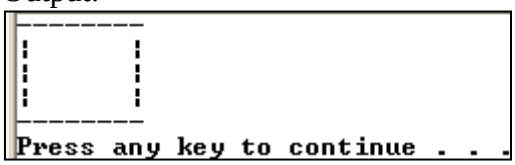
1. Output:




2. The same set of random numbers generated every time the program is run.

3. Different set of random numbers generated every time the program is run.

4. Output:




5.

(a)

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

// Function prototypes
void draw_intersect(void);
void draw_parallel(void);
void draw_horizontal(void);

int main(void)
{
	// Draw parallel lines
	draw_parallel();

	// Draw a horizontal line
	draw_horizontal();

	// Draw intersecting lines
	draw_intersect();

	return 0;
}

/*
 *  Draws intersecting lines
 */
void draw_intersect(void)
{
	cout << "   /\\   \n";
	cout << "  /  \\  \n";
	cout << " /    \\ \n";
}
```

```
/*
 *  Draws a triangle
 */
void draw_triangle(void)
{
	draw_intersect();
	draw_horizontal();
}

/*
 *  Draws a horizontal line
 */
void draw_horizontal(void)
{
	cout << "--------\n");
}

/*
 *  Draws parallel lines
 */
void draw_parallel(void)
{
	cout << "|      |\n";
	cout << "|      |\n";
	cout << "|      |\n";
}
```

(b)

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

// Function prototypes
void draw_intersect(void);
void draw_parallel(void);
void draw_horizontal(void);
void draw_triangle(void);

int main(void)
{
	// Draw a triangle
	draw_triangle();

	// Draw parallel lines
	draw_parallel();

	return 0;
}


/*
 *  Draws intersecting lines
 */
void draw_intersect(void)
{
	cout << "   /\\   \n";
	cout << "  /  \\  \n";
	cout << " /    \\ \n";
}
```

```
/*
 *  Draws a triangle
 */
void draw_triangle(void)
{
	// Draw intersecting lines
	draw_intersect();

	// Draw a horizontal line
	draw_horizontal();
}

/*
 *  Draws a horizontal line
 */
void draw_horizontal(void)
{
	cout << "--------\n";
}

/*
 *  Draws parallel lines
 */
void draw_parallel(void)
{
	cout << "|      |\n";
	cout << "|      |\n";
	cout << "|      |\n";
}
```

6.

```
1     #include <iostream>
2     #include <iomanip>
3     // defined constant for price per apple
4->   #define PRICE_PER_APPLE 0.85
5     // defined constant for price per orange
6->   #define PRICE_PER_ORANGE 1.00
7     using namespace std;
8
9     // function prototypes
10    void display_heading(void);
11->  void display_apple_item (int qty, double total);
12->  void display_orange_item(int qty, double total);
13->  void display_grand_total(double grand_total);
14
15    int main(void)
16    {
17          int apple_qty, orange_qty;
18          double apple_total, orange_total, grand_total;
19
20          // get number of apples and oranges
21          cout << "Enter number of apples: ";
22          cin >> apple_qty;
23          cout << "Enter number of oranges: ";
24          cin >> orange_qty;
25
26          // compute total for apples and oranges
27->         apple_total = apple_qty * PRICE_PER_APPLE;
28->         orange_total = orange_qty * PRICE_PER_ORANGE;
```

```
29          // compute grand total
30          grand_total = apple_total + orange_total;
31
32          // call the functions to display the result
33          display_heading();
34->        display_apple_item (apple_qty, apple_total);
35->        display_orange_item(orange_qty, orange_total);
36          display_grand_total(grand_total);
37          return 0;
38      }
39
40      void display_heading(void)
41      {
42          cout << "=========My Fruit Store========\n";
43          cout << "Fruit   Qty  Price        Total\n";
44          cout << "------------------------------\n";
45      }
46
47      void display_apple_item (int qty, double total)
48      {
49->        cout << "Apple\t" << setw(3) << qty;
            cout << "  RM" << setw(4) << fixed << setprecision(2) << PRICE_PER_APPLE;
            cout << "\tRM" << setw(5) << total << endl;
50      }
51
52      void display_orange_item(int qty, double total)
53      {
54->        cout << "Orange\t" << setw(3) << qty;
            cout << "  RM" << setw(4) << fixed << setprecision(2) << PRICE_PER_ORANGE;
            cout << "\tRM" << setw(5) << total << endl;
55      }
56
57      void display_grand_total(double total)
58      {
59          cout << "------------------------------\n";
60->         cout << "Total\t\t\tRM" << setw(5) << fixed << setprecision(2)
                 << total << endl;
61      }
```