1. (30 pts)
(a) Programs that translate high-level programs into machine language are called ______________.
(b) Determine the values of each variable after the calculation is performed. Assume that when each
statement begins executing all variables have the value 5.

```c
int x, product, quotient;
(a) product *= ++x;
   product = __________, x = __________
(b) quotient /= x++;
   quotient = __________, x = __________
```
(c) Here is a flow diagram for the if-else statement

```c
if ( x == 0 )
   y = 3;
else
   y = 5;
```

Now give a flow diagram for the following for statement.

```c
for ( int i = 1; i <= 10; i++ )
   sum += i;
```

(d) A variable that is known only in the body of the function in which it is defined is called a
______________.
(e) The __________ of an identifier is the portion of the program in which the identifier can be
used.
(f) An enumeration, introduced by the key word enum and followed by a new type name, is a set of
integer constants represented by identifiers. True False (circle correct ans)
(g) The binary number 1101 = __________ in decimal notation.
(h) The decimal number 9 = __________ in binary notation.
(i) The case labels in a switch statement can either be constants like `case 'e';`, `case 13:` or they can be variables like

```c
int a, b, c;
...
case a:
...
case b:
    True False
```

(j) __________ functions enable a single function definition for functions that perform a task on multiple data types.

2. (15 pts) Show the column by column output produced by the program below. Put your output in the boxes. Each box is one column on the page.

```
1 2 3 4 5 6 7 8 9 10
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
<td>--</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

```c
#include <iostream.h>
main()
{
    int dayw = 3, daym, daym_max = 28;
    for(int i = 1; i < dayw; i++)
        cout << " ";           //This string contains 3 spaces
    for(daym = 1; daym <= daym_max; daym++) {  
        cout << setw(2) << daym; // Print the day of the month
        if (dayw == 7) {          // Goto next day of the week
            dayw = 1;
            cout << endl;
        }
        else {
            dayw++;  
            cout << ', ';
        }
    } // end of loop
}
```

3. (10 pts) Consider the following program.
#include<iostream.h>

int tvb( int arg );
void tbr( int& arg );

main()
{
    int n = 10;
    cout << "First result = " << tvb( n ) << endl;
tbr( n );
    cout << "Second result = " << n << endl;
}

int tvb( int arg )
{
    arg *= 3;
    return arg;
}

void tbr( int& arg )
{
    arg *= 3;
}

(a) What is printed by the program?

(b) Explain how the program works.

4. (10 pts) Write a main() program that reads integers until EOF and prints the sum and average value of the numbers.

5. (15 pts) Using for loops (but no arrays) write a function for n ≥ 1,

   ```cpp
   void printSquare(int n)
   ```

   to print a hollow square. For example, if n == 4, the function would print

   ```
   ****
   *   *
   *   *
   ****
   ```
If $n == 1$, the function should print

*

Assume that $n$ *’s will fit on a single line on the output page.

6. (15 pts) A numerical palindrome is a number that reads the same backwards or forwards, for example, 12321, 51115, or 248842. Write a function

```c
int palindrome(int n)
```

that determines if the non-negative integer $n$ is a palindrome and returns an `int 1` (representing true) if $n$ is a palindrome and an `int 0` (representing false) otherwise.

Hint: Make use of the function `int reverse(int n)` that reverses the digits of $n$ as was done for homework. If you use this function, however, you must give the code for it. The functions should not print anything.