1. (3 points) What is the output of this program?

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

int main()
{
    int a = 2, b = 3, c = 4, d = 5;
    cout << (a + b * c) << endl;  // (0)
    cout << (c / d) << endl;      // (1)
    cout << (b % d) << endl;      // (2)
    cout << (a--) << endl;        // (3)
    cout << (!(c > d) && a <= b) << endl; // (4)
    cout << (a == b ? c : d) << endl;  // (5)

    for ( int i = 100; i >= 50; i -= 10 ) cout << i << ' ';
    cout << endl;                  // (6)

    char letter = 'a';
    do {
        cout << letter++ << '~';
    } while ( letter < 'f' );
    cout << endl;                  // (7)

    return 0;
}
```
2. (3 points) **Summer Olympic Games**

The *Summer Olympic Games*, first held in 1896, is a major international multi-sport event held once every four years. The scheduled 1916 Summer Olympics were cancelled following the onset of World War I. Due to World War II, the Games of 1940 were cancelled. The Games of 1944 were due to be held in London but were also cancelled. Due to the COVID-19 pandemic, the 2020 Games were to be delayed until 2021, marking the first time that the Olympic Games have been postponed.

In this program, user is asked to enter a year and this program checks whether the year entered by user is Summer Olympic Games year or not.

**Solution:**

```cpp
int main()
{
    int year = 2000;
    if ( year == 1916 || year == 1940 || year == 1944 || year == 2020 )
    {
        cout << "Not" << endl;
    }
    else if ( year == 2021 || year % 4 == 0 )
    {
        cout << "Olympic year" << endl;
    }
    /* Another way: if ( year % 4 == 0 && year != 2020
        && !(year == 1940) .. ) */
    return 0;
}
```
3. **(3 points) Calculate \( \pi \)**

The value of \( \pi \) can be approximated using the following expression:

\[
\sum_{n=1}^{\infty} \frac{1}{n^2} = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \ldots + \frac{1}{n^2} = \frac{\pi^2}{6}
\]

where \( n \) is a positive integer. Write a program that uses this formula to approximate the value of \( \pi \). Test your program for \( n = 1000 \).

**Solution:**

```cpp
int main()
{
    double total = 0;

    for ( int i = 1; i <= 10000; i++ )
    {
        total += 1. / (i * i);
    }

    double pi = sqrt(6 * total);

    cout << pi << endl;
    return 0;
}
```
4. (3 points) **Power Table**

This program takes two positive integers from a user and displays a power table. For example, if the input number is 2 and 5, the program would print:

2 to the power 0 is 1
2 to the power 1 is 2
2 to the power 2 is 4
2 to the power 3 is 8
2 to the power 4 is 16
2 to the power 5 is 32

**Solution:**

```cpp
int main()
{
    cout << "Please input two positive integer: ";
    int x, n;

    cin >> x >> n;

    int total = 1;

    for ( int i = 0; i <= n; i++ )
    {
        cout << x << " to the power " << i << " is " << total << endl;
        total *= x;
    }
}
```
5. **(3 points) Cell Phone Bill**

Write a full program that computes the final cost for a cell phone bill.

AT&T has two choices for unlimited plans: Unlimited Choice and Unlimited Plus. Both unlimited plans allow up to 10 lines. For two lines, the Unlimited Choice plan starts at $125 per month, and Unlimited Plus starts at $155. Each additional line beyond the first two is an extra $20 per month. Federal Universal Service Fund fees are 18.8% (of the total) additional charge.

Write code that prompts the user for the plan (choice or plus), and prompts the user for the number of lines. The code prints out the total amount due each month.

**Solution:**

```cpp
int main()
{
    int choice = 0, nlines = 3;
    double total;
    cout << "...";
    cin >> choice;
    cin >> nlines;
    if (choice)
        total = (125 + 20 * (nlines - 2)) * 1.188;
    else
        total = (155 + 20 * (nlines - 2)) * 1.188;
    cout << total << endl;
    return 0;
}
```
6. (3 points) **Sic Bo**

Write a program that lets the user play a casino game called *Sic bo* against the computer. The program should work as follows:

1. When the program begins, three random dices is rolled. (Don’t display the values yet.)
2. The user enters his or her guessing at the keyboard. The user can chose from Big and Small.
3. The computer’s result is displayed.
4. The computer displays whether the user wins or loses by these rules:
   - “Three dices have a same value” is called a **triple**.
   - The total from 11 to 17 with the exception of a triple is **Big**.
   - The total from 4 to 10 with the exception of a triple is a **Small**.

(That means any triple makes you lose.)

**Solution:**

```cpp
int main()
{
    srand(time(0));
    int a = rand() % 6 + 1;
    int b = rand() % 6 + 1;
    int c = rand() % 6 + 1;
    int d = a + b + c;

    int guess;
    cin >> guess; // assume 0 for small, 1 for big.

    if ( a == b && b == c ) cout << "You lose."
    else if ( 11 <= d && d <= 17 && guess == 1 ) cout << "You win."
    else if ( 4 <= d && d <= 10 && guess == 0 ) cout << "You win."
    else cout << "You lose."

    return 0;
}
```