Problem 1 (points) Write a complete C++ program that prints the numbers from 28 to 387 with 10 numbers (separated by spaces) on each line.
The output from your program should begin

28 29 30 31 32 33 34 35 36 37
38 39 40 41 42 43 44 45 46 47

Answer:

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

int main() {
    for (int n = 28; n < 388; n++) {
        cout << n << " ";
        if (n % 10 == 7) cout << endl;
    }
    return 0;
}
```
Problem 2  (points)
Write C++ statements to carry out the following tasks. Do not write complete programs, just give a few lines of
C++ code. No answer can use more than two lines. Assume the following variables have been declared and have
legal values

int x = 18;

(a) Print to the user’s screen the sentence: *In C++ an endl makes a new line.*
Answer:

    cout << "In C++ an endl makes a new line." << endl;

(b) Print the square of \(x\).
Answer:

    cout << x * x << endl;

(c) Print a random number with 4 digits.
Answer:

    cout << rand() % 9000 + 1000 << endl;

(d) Print all numbers less that 1000 that are either divisible 7 or are even and greater than 400.
Answer:

    for (int n = 1; n < 1000; n++)
        if ((n % 7 == 0) || ((n % 2 == 0) && (n > 400))) cout << n << endl;

(e) Print the square root of \(3/8\).
Answer:

    cout << sqrt(3.0 / 8) << endl;
Problem 3  (points) Consider the following C++ program.

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

int main() {
    double x = 9.0, y = 16.0, z = 25.0;
    string a = "b", b = "a";
    cout << sqrt(z) << endl;    // line (a)
    cout << sqrt(sqrt(y)) << endl; // line (b)
    if ((x + y) != z) cout << b << endl; // line (c)
    cout << a << "a" << "b" << b << endl; // line (d)
    if (a == "b") cout << z; else cout << x; // line (e)
    cout << endl;
}
```

(a) What is the output at line (a)?

**Answer:**

5

(b) What is the output at line (b)?

**Answer:**

2

(c) What is the output at line (c)?

**Answer:**

(d) What is the output at line (d)?

**Answer:**

baba

(e) What is the output at line (e)?

**Answer:**

25
Problem 4  (points) Write a complete C++ program that asks the user for a number $n$ and prints 2 large copies of an X pattern (each with height $n$) in a horizontal sequence.

For example, if the user specified 5 for $n$, the program would print as follows:

```
  *   **  *
**   *   **
  * **   *
**   *   **
  *   **  *
```

(Each X pattern should begin in the column after the previous one ends. Do not try to check whether the user input is legal or sensible.)

Answer:

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

int main() {
    int n;
    cout << "Enter a number: ";
    cin >> n;

    for (int row = 1; row <= n; row++) {
        for (int pattern = 1; pattern <= 2; pattern++) {
            for (int c = 1; c <= n; c++) {
                if (row == c || (row + c) == (n + 1))
                    cout << "*";
                else cout << " ";
            }
            cout << endl;
        }
    }
    return 0;
}
```
Problem 1  (points) Write a complete C++ program that prints the numbers from 980 down to 666 with 6 numbers (separated by spaces) on each line.
The output from your program should begin

980 979 978 977 976 975  
974 973 972 971 970 969  

Answer:

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

int main() {
    for (int n = 980; n >= 666; n--) {
        cout << n << " ";
        if (n % 6 == 3) cout << endl;
    }
    cout << endl;
    return 0;
}
```
Problem 2  (points)
Write C++ statements to carry out the following tasks. Do not write complete programs, just give a few lines of C++ code. No answer can use more than two lines. Assume the following variables have been declared and have legal values

```cpp
int y = 12;
```

(a) Print to the user’s screen the sentence: C++ output uses cout.
Answer:

```cpp
cout << "C++ output uses cout." << endl;
```

(b) Print the square root of $y$.
Answer:

```cpp
cout << sqrt((double) y) << endl;
```

(c) Print a random 3 digit even number.
Answer:

```cpp
cout << 2 * (rand() % 450) + 100 << endl;
```

(d) Print all numbers less than 1000 that end in a 7 and are divisible by 3.
Answer:

```cpp
for (int n = 1; n < 1000; n++)
    if ((n % 10 == 7) && (n % 3 == 0)) cout << n << endl;
```

(e) Print the square of $3/8$.
Answer:

```cpp
cout << (3.0 / 8) * (3.0 / 8) << endl;
```
Problem 3  (points) Consider the following C++ program.

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

int main() {
    double x = 9.0, y = 16.0, z = 25.0;
    string a = "a", b = "a";
    cout << sqrt(y) << endl;  // line (a)
    cout << sqrt(y) + sqrt(x) << endl;  // line (b)
    if ((x + y) == z) cout << b << endl;  // line (c)
    cout << a << "a" << "b" << b << endl;  // line (d)
    if (a == "b") cout << z; else cout << x;  // line (e)
    cout << endl;
}
```

(a) What is the output at line (a)?
**Answer:**

4

(b) What is the output at line (b)?
**Answer:**

7

(c) What is the output at line (c)?
**Answer:**

a

(d) What is the output at line (d)?
**Answer:**

aaba

(e) What is the output at line (e)?
**Answer:**

9
Problem 4  (points) Write a complete C++ program that asks the user for a number \( n \) and prints 3 large copies of an L pattern (each with height \( n \)) in a horizontal sequence. 

For example, if the user specified 4 for \( n \), the program would print as follows:

```
*   *   *
*   *   *
*   *   *
**   **   **
```

(Each L pattern should begin after a gap of one column after the previous one ends. Do not try to check whether the user input is legal or sensible.)

Answer:

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

int main() {
    int n;
    cout << "Enter a number: ";
    cin >> n;

    for (int row = 1; row <= n; row++) {
        for (int pattern = 1; pattern <= 3; pattern++) {
            for (int c = 1; c <= n; c++) {
                if (c == 1 || row == n)
                    cout << "*";
                else cout << " ";
            }
            cout << " ";
        }
        cout << endl;
    }
    return 0;
}
```
Problem 1  (points) Write a complete C++ program that prints the numbers from 28 to 387 with 10 numbers (separated by commas) on each line.

The output from your program should begin

28,29,30,31,32,33,34,35,36,37
38,39,40,41,42,43,44,45,46,47

Answer:

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

int main() {
    for (int n = 28; n < 388; n++) {
        cout << n;
        if (n % 10 == 7) cout << endl;
        else cout << ",";
    }
    return 0;
}
```
Problem 2  ( points)
Write C++ statements to carry out the following tasks. Do not write complete programs, just give a few lines of C++ code. No answer can use more than two lines. Assume the following variables have been declared and have legal values

    int z = 5;

(a) Print to the user’s screen the words: \textit{endl makes a line and for makes a loop}
\textbf{Answer:}

    cout << "endl makes a line and for makes a loop" << endl;

(b) Print the cube of \(z + 1\).
\textbf{Answer:}

    cout << (z + 1) * (z + 1) * (z + 1) << endl;

(c) Print a random 2 digit number to the user’s screen.
\textbf{Answer:}

    cout << rand() % 90 + 10 << endl;

(d) Print all three digit numbers that either end in a 7 or are even and divisible by 7.
\textbf{Answer:}

    for (int n = 100; n < 1000; n++)
        if ((n % 10 == 7) || ((n % 2 == 0) && (n % 7 == 0))) cout << n << endl;

(e) Print the square root of \(3/7\).
\textbf{Answer:}

    cout << sqrt(3.0/7) << endl;
Problem 3  (points) Consider the following C++ program.

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

int main() {
    double x = 49.0, y = 81.0, z = 25.0;
    string a = "ab", b = "ba";
    cout << sqrt(x) << endl;  // line (a)
    cout << sqrt(sqrt(y)) << endl;  // line (b)
    if ((x + y) != z) cout << a << endl;  // line (c)
    cout << a << "a" << "b" << b << endl;  // line (d)
    if (a == "b") cout << x; else cout << y;  // line (e)
    cout << endl;
}
```

(a) What is the output at line (a)?
**Answer:**

```
7
```

(b) What is the output at line (b)?
**Answer:**

```
3
```

(c) What is the output at line (c)?
**Answer:**

```
ab
```

(d) What is the output at line (d)?
**Answer:**

```
ababba
```

(e) What is the output at line (e)?
**Answer:**

```
81
```
Problem 4  (points) Write a complete C++ program that asks the user for a number \( n \) and prints 2 large copies of an E pattern (each with height \( n \) that is odd) in a horizontal sequence.

For example, if the user specified 5 for \( n \), the program would print as follows:

```
***** *****
*   *
***** *****
*   *
***** *****
```

(Each E pattern should begin after a gap of one column after the previous one ends. Do not try to check whether the user input is legal or sensible.)

Answer:

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

int main() {
    int n;
    cout << "Enter an odd number: ";
    cin >> n;

    for (int row = 1; row <= n; row++) {
        for (int pattern = 1; pattern <= 2; pattern++) {
            for (int c = 1; c <= n; c++) {
                if (c == 1 || row == 1 || row == n || row == (n + 1) / 2)
                    cout << "*";
                else cout << " ";
            }
            cout << " ";
        }
        cout << endl;
    }
    return 0;
}
```
Problem 1 (points) Write a complete C++ program that prints the numbers from 980 down to 669 with 6 numbers (separated by periods) on each line. The output from your program should begin

980.979.978.977.976.975
974.973.972.971.970.969

Answer:

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

int main() {
    for (int n = 980; n >= 669; n--) {
        cout << n;
        if (n % 6 == 3) cout << endl;
        else cout << ".";
    }
    cout << endl;
    return 0;
}
```
Problem 2  (points)
Write C++ statements to carry out the following tasks. Do not write complete programs, just give a few lines of C++ code. No answer can use more than two lines. Assume the following variables have been declared and have legal values

```cpp
int y = 12;
```

(a) Print to the user's screen the sentence: Quote Hello but do not quote cout.
Answer:

```cpp
cout << "Quote Hello but do not quote cout." << endl;
```

(b) Print the square root of the square root of \( y \).
Answer:

```cpp
cout << sqrt(sqrt((double) y)) << endl;
```

(c) Print a random 3 digit number that is divisible by 3.
Answer:

```cpp
cout << 3 * (rand() % 300) + 102 << endl;
```

(d) Print all numbers less that 1000 that end in a 7 and are divisible by 7.
Answer:

```cpp
for (int n = 1; n < 1000; n++)
    if ((n % 10 == 7) && ((n % 7) == 0)) cout << n << endl;
```

(e) Print the square of 5/8.
Answer:

```cpp
cout << (5.0 / 8) * (5.0 / 8) << endl;
```
Problem 3  (points) Consider the following C++ program.

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

int main() {
    double x = 81.0, y = 49.0, z = 36.0;
    string a = "az", b = "za";
    cout << sqrt(y) << endl; // line (a)
    cout << sqrt(y) + sqrt(x) << endl; // line (b)
    if ((x + y) == z) cout << b << endl; // line (c)
    cout << a << "a" << "b" << b << endl; // line (d)
    if (a == "b") cout << z; else cout << x; // line (e)
    cout << endl;
}
```

(a) What is the output at line (a)?

Answer:

7

(b) What is the output at line (b)?

Answer:

16

(c) What is the output at line (c)?

Answer:

(d) What is the output at line (d)?

Answer:

azabza

(e) What is the output at line (e)?

Answer:

81
Problem 4  (points) Write a complete C++ program that asks the user for a number \( n \) and prints 3 large copies of a T pattern (each with height \( n \) that is odd) in a horizontal sequence.

For example, if the user specified 5 for \( n \), the program would print as follows:

```
***** ***** *****
*   *   *
*   *   *
*   *   *
*   *   *
```

(Each T pattern should begin after a gap of one column after the previous one ends. Do not try to check whether the user input is legal or sensible.)

Answer:

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

int main() {
    int n;
    cout << "Enter an odd number: ";
    cin >> n;
    for (int row = 1; row <= n; row++) {
        for (int pattern = 1; pattern <= 3; pattern++) {
            for (int c = 1; c <= n; c++) {
                if (row == 1 || c == (n + 1) / 2)
                    cout << "*";
                else cout << " ";
            }
            cout << " ";
        }
        cout << endl;
    }
    return 0;
}
```