

Solutions

09.00am – 09.50am, Wednesday, October 18, 2017

Problem 1 (10 points) The following C++ program has errors at the lines marked a,b,c,d, and e. For each answer write a single line of C++ that fixes all errors in the corresponding line. Do not change anything that is correct.

```
#Include <iostream> // line a
using namespace std;

int main{ // line b

    int x = 1, y = 2;

    for (int x == 1; x < 10; x++); // line c

    cout << y % y endl; // line d

    return; // line e
}
```

(a) Correct line (a):

Answer:

```
#include <iostream>
```

(b) Correct line (b):

Answer:

```
int main() {
```

(c) Correct line (c):

Answer:

```
    for (int x = 1; x < 10; x++);
```

(d) Correct line (d):

Answer:

```
    cout << y % y << endl;
```

(e) Correct line (e):

Answer:

```
    return 0;
```

Problem 2 (10 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 that x and y have been declared.

```
int x; double y;
```

(a) Read a value of x typed on the screen by the user

Answer:

```
cin >> x;
```

(b) If x is negative, replace x by the 4th power of its value

Answer:

```
if (x < 0) x = x * x * x * x;
```

(c) Print on the screen the last 2 digits of x.

Answer:

```
cout << x % 100 << endl;
```

(d) Print on the screen a random number between 1 and x (inclusive).

Answer:

```
cout << 1 + rand() % x << endl;
```

(e) Set y to the exact value of the quotient of x by 100.

Answer:

```
y = x / 100.0;
```

Problem 3 (10 points) Consider the following C++ program. Suppose that a user runs the program and enters -10 as input.

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

int main() {
    int a;
    cout << "Enter a\npositive integer\n";           // line (a)
    cin >> a;

    cout << a << - a << a - a << endl;               // line (b)

    for (int n = a; n <= 0; n += 4) cout << n;       // line (c)
    cout << endl;

    cout << a + 20 / 3 << endl;                       // line (d)

    cout << (a + 20) % 3 << endl;                     // line (e)

    return 0;
}
```

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

Answer:

```
Enter a
positive integer
```

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

Answer:

```
-10100
```

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

Answer:

```
-10-6-2
```

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

Answer:

```
-4
```

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

Answer:

```
1
```

Problem 4 (10 points) Write a complete C++ program that asks the user to enter a positive integer n . If n is not positive the program should print a message *Not positive. Enter another:* and continue to do this until the user enters a positive integer. Then the program should print n squares of *s with decreasing size that begin with an $n \times n$ square and end with a 1×1 square. The right hand edges of the squares should line up. (Partial credit will be given for the parts of the program that you complete successfully.) Excessively long solutions lines might lose some credit.

For example, here is a sample run of the program:

```
Enter a positive integer n: -3
Not positive. Enter another: 4
****
****
****
****
***
***
**
**
*
```

Answer:

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

int main() {
    int n;
    cout << "Enter a positive integer n: ";
    cin >> n;
    while (n <= 0) {
        cout << "Not positive. Enter another: ";
        cin >> n;
    }
    for (int s = n; s >= 1; s--) {
        for (int r = 1; r <= s; r++) {
            for (int c = 1; c <= n; c++)
                if (c > n - s) cout << "*";
                else cout << " ";
            cout << endl;
        }
    }
    return 0;
}
```

Solutions

09.00am – 09.50am, Wednesday, October 18, 2017

Problem 1 (10 points) The following C++ program has errors at the lines marked a,b,c,d, and e. For each answer write a single line of C++ that fixes all errors in the corresponding line. Do not change anything that is correct.

```
#include <Iostream>           // line a
using namespace std;

Int main()                   // line b

    int z = 5; double y = 12.0;

    for (z = 1; z < 10; z--); // line c

    cout << y % z << endl;    // line d

    return z                 // line e
}
```

(a) Correct line (a):

Answer:

```
#include <iostream>
```

(b) Correct line (b):

Answer:

```
int main() {
```

(c) Correct line (c):

Answer:

```
    for (z = 1; z < 10; z++);
```

(d) Correct line (d):

Answer:

```
    cout << ((int) y) % z << endl;
```

(e) Correct line (e):

Answer:

```
    return z;
```

Problem 2 (10 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 that x and y have been declared.

```
int y; double x;
```

(a) Read a value of x typed on the screen by the user.

Answer:

```
cin >> x;
```

(b) If x is negative, replace x by the 2nd power of its value.

Answer:

```
if (x < 0) x = x * x;
```

(c) Make y be the closest integer to x (round up half).

Answer:

```
y = (int) (x + 0.5);
```

(d) Print to the screen a random number between 0 and y (inclusive).

Answer:

```
cout << rand() % (y + 1) << endl;
```

(e) Set x to the exact value of the quotient of y by 13.

Answer:

```
x = y / 13.0;
```

Problem 3 (10 points) Consider the following C++ program. Suppose that a user runs the program and enters 6 as input.

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

int main() {
    int a;
    cout << "Enter a\npositive integer\n";           // line (a)
    cin >> a;

    cout << a << - a << a - a << endl;                // line (b)

    for (int n = a; n <= 0; n += 4) cout << n;       // line (c)
    cout << endl;

    cout << a + 20 / 3 << endl;                        // line (d)

    cout << (a + 20) % 3 << endl;                     // line (e)

    return 0;
}
```

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

Answer:

```
Enter a
positive integer
```

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

Answer:

```
6-60
```

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

Answer:

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

Answer:

```
12
```

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

Answer:

```
2
```

Problem 4 (10 points) Write a complete C++ program that asks the user to enter an odd positive integer n . If n is illegal the program must terminate at once. Otherwise the program should print squares of *s. that begin with an $n \times n$ square and end with a 1×1 square and such that the sizes of squares decrease by 2 as they go down the page. The right hand edges of the squares should line up. (Partial credit will be given for the parts of the program that you complete successfully.) Excessively long solutions lines might lose some credit.

For example, here is a sample run of the program:

```
Enter an odd positive integer n: 5
*****
*****
*****
*****
*****
***
***
***
***
*
```

Answer:

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

int main() {
    int n;
    cout << "Enter an odd positive integer n: ";
    cin >> n;
    if (n <= 0 || n % 2 == 0) return 0;
    for (int s = n; s >= 1; s-=2) {
        for (int r = 1; r <= s; r++) {
            for (int c = 1; c <= n; c++)
                if (c > n - s) cout << "*";
                else cout << " ";
            cout << endl;
        }
    }
    return 0;
}
```

Solutions

02.45pm – 03.35pm, Wednesday, October 18, 2017

Problem 1 (10 points) The following C++ program has errors at the lines marked a,b,c,d, and e. For each answer write a single line of C++ that fixes all errors in the corresponding line. Do not change anything that is correct.

```
#INCLUDE <iostream>. // line a
using namespace std;

int main(){ // line b
    int x = 1; y = 2; // line c
    for (int x == 1; x == 1; x++); // line d
    cout << y % y << endl;
    Return 10; // line e
}
```

(a) Correct line (a):

Answer:

```
#include <iostream>.
```

(b) Correct line (b):

Answer:

```
int main() {
```

(c) Correct line (c):

Answer:

```
int x = 1, y = 2;
```

(d) Correct line (d):

Answer:

```
for (int x = 1; x == 1; x++);
```

(e) Correct line (e):

Answer:

```
return 10;
```

Problem 2 (10 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 that *name* and *age* have been declared.

```
int age; string name;
```

(a) Read values of name and age typed on the screen by the user

Answer:

```
cin >> name >> age;
```

(b) If the name is Freddy, divide the age by 10 and use that as age.

Answer:

```
if (name == "Freddy") age = age / 10;
```

(c) Print on the screen the last 2 digits of the age.

Answer:

```
cout << age % 100 << endl;
```

(d) Print on the screen a random number between 1 and age (inclusive).

Answer:

```
cout << 1 + rand() % age << endl;
```

(e) Print the users name 5 times on 5 lines of the screen.

Answer:

```
for (int i = 1; i <= 5; i++)  
cout << name << endl;
```

Problem 3 (10 points) Consider the following C++ program. Suppose that a user runs the program and enters -20 as input.

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

int main() {
    int a;
    cout << "Enter a\npositive integer\n";           // line (a)
    cin >> a;

    cout << a << - a << a - a << endl;             // line (b)

    for (int n = a; n <= 0; n += 4) cout << n;      // line (c)
    cout << endl;

    cout << a + 20 / 3 << endl;                     // line (d)

    cout << (a + 20) % 3 << endl;                   // line (e)

    return 0;
}
```

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

Answer:

```
Enter a
positive integer
```

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

Answer:

```
-20200
```

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

Answer:

```
-20-16-12-8-40
```

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

Answer:

```
-14
```

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

Answer:

```
0
```

Problem 4 (10 points) Write a complete C++ program that asks the user to enter a positive integer n . If n is not positive the program should print a message *Not positive. Enter another:* and continue to do this until the user enters a positive integer. Then the program should print a square of side n that is made up from a large letter L made of #’s in the left column and bottom row, then a slightly smaller L made made of o’s inside it, then a smaller L made of #’s and so on. (Partial credit will be given for the parts of the program that you complete successfully.) Excessively long solutions lines might lose some credit.

For example, here is a sample run of the program:

```
Enter a positive integer n: -5
Not positive. Enter another: 7
#o#o#o#
#o#o#oo
#o#o###
#o#oooo
#o#####
#oooooo
#####
```

Answer:

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

int main() {
    int n;
    cout << "Enter a positive integer n: ";
    cin >> n;
    while (n <= 0) {
        cout << "Not positive. Enter another: ";
        cin >> n;
    }
    for (int r = n; r >= 1; r--) {
        for (int c = 1; c <= n; c++) {
            int x = r;
            if (c <= r) x = c;
            if (x % 2 == 0) cout << "o";
            else cout << "#";
        }
        cout << endl;
    }
    return 0;
}
```

Solutions

02.45pm – 03.35pm, Wednesday, October 18, 2017

Problem 1 (10 points) The following C++ program has errors at the lines marked a,b,c,d, and e. For each answer write a single line of C++ that fixes all errors in the corresponding line. Do not change anything that is correct.

```
include <iostream>           // line a
using namespace std;

int main() {

    int z = 5, double y = 12.0; // line b

    for (z = 1; z !> 10; z++); // line c

    cout (int) y % z endl; // line d

    Return z. // line e
}
```

(a) Correct line (a):

Answer:

```
#include <iostream>
```

(b) Correct line (b):

Answer:

```
int z = 5; double y = 12.0;
```

(c) Correct line (c):

Answer:

```
for (z = 1; z <= 10; z++);
```

(d) Correct line (d):

Answer:

```
cout << (int) y % z << endl;
```

(e) Correct line (e):

Answer:

```
return z;
```

Problem 2 (10 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 that *score* and *name* have been declared.

```
string name; double score;
```

(a) Read the score and name typed on the screen by the user.

Answer:

```
cin >> score >> name;
```

(b) If the name is not Freddy add a bonus of 10 to the score.

Answer:

```
if (name != "Freddy") score += 10;
```

(c) Print the closest integer to the score (round up half).

Answer:

```
cout << (int) (score + 0.5);
```

(d) Print to the screen a random integer between 1 and score.

Answer:

```
cout << 1 + rand() % ((int) score) << endl;
```

(e) Print the user's name 20 times on one line of output.

Answer:

```
for (int i = 1; i <= 20; i++) cout << name << " ";  
cout << endl;
```

Problem 3 (10 points) Consider the following C++ program. Suppose that a user runs the program and enters 20 as input.

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

int main() {
    int a;
    cout << "Enter a\npositive integer\n";           // line (a)
    cin >> a;

    cout << a << - a << a - a << endl;             // line (b)

    for (int n = a; n <= 0; n += 4) cout << n;       // line (c)
    cout << endl;

    cout << a + 20 / 3 << endl;                     // line (d)

    cout << (a + 20) % 3 << endl;                   // line (e)

    return 0;
}
```

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

Answer:

```
Enter a
positive integer
```

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

Answer:

```
20-200
```

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

Answer:

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

Answer:

```
26
```

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

Answer:

```
1
```

Problem 4 (10 points) Write a complete C++ program that asks the user to enter a positive integer n . If n is not positive the program must terminate at once. Then the program should print a square of side n that is made up from a large backwards letter L made of #'s in the right column and bottom row, then a slightly smaller reversed L made made of o's inside it, then a smaller reversed L made of #'s and so on. (Partial credit will be given for the parts of the program that you complete successfully.) Excessively long solutions lines might lose some credit.

For example, here is a sample run of the program:

```
Enter a positive integer n: 7
#o#o#o#
oo#o#o#
###o#o#
oooo#o#
#####o#
oooooo#
#####
```

Answer:

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

int main() {
    int n;
    cout << "Enter a positive integer n: ";
    cin >> n;
    if (n <= 0) return 0;
    for (int r = 1; r <= n; r++) {
        for (int c = 1; c <= n; c++) {
            int x = r;
            if (c >= r) x = c;
            if (x % 2 == n % 2) cout << "#";
            else cout << "o";
        }
        cout << endl;
    }
    return 0;
}
```