

1.

a.

In the following segment of C++ code, each statement introduces one single new variable or constant, or changes one single variable. After each statement write down the value that the constant or variable will have.

Don't bother with a and b, they are trivial and would be a waste of ink. Start with the one numbered *i*.

```
const int a = 37;  
const double b = 37;
```

i int c = a/5;

ii int d = a/5*10;

iii const int e = 10*a/5;

iv double f = a/5;

v double g = a/5.0;

vi double h = f/2;

vii f += a/10;

viii f += b/10;

For the remaining parts, write down exactly what would be printed by each statement.

ix cout << 72364 - 72364 / 1000 * 1000;

x cout << 72364 - 72364.0 / 1000 * 1000;

xi cout << 72364 - 72364 / 1000.0 * 1000;

xii cout << 72364 - 72364 / 1000 * 1000.0;

xiii cout << 72364.0 - 72364 / 1000 * 1000;

b.

What would be printed by this program?
This is much easier than it looks.

```
#include <iostream>

int a=12, b=17;

void first(int x)
{ int a = 43;
  cout << "first: " << a << b;
  a += x;
  b += 1; }

void second(void)
{ const int d = 31;
  cout << "second(1): " << a << b;
  first(d);
  cout << "second(2): " << a << b;
  a += 1;
  b += 1; }

void main()
{ int b = 63, c = 72;
  cout << "main(1): " << a << b << c;
  second();
  cout << "main(2): " << a << b << c; }
```

2.

For this question, do not describe what the individual statements mean, do not tell me that a loop goes round, or that something is divided by ten. Analyse the functions so that you know what their overall effect is. Give me a description that would be meaningful to an educated person who knows nothing of programming (if there could be such a person).

a.

Here is a mystery function. Work out what it does. Be rational: it may not be immediately obvious, some little thought may be required.

```
int mystery(int y)
{ int x = 0;
  while (true)
  { if (x*x == y)
    return x;
    if (x*x > y)
    return -1;
    x+=1;
  }
}
```

What it does can be described very simply in English. Just write that simple explanation. If you are not totally confident in your answer, show how you arrived at it.

b.

Here is another mysterious function. Just as for part a, tell me what it does. The function will never be used with a negative value for n.

```
int elephant(int n)
{ if (n < 10)
  return 1;
  else
  return 1 + elephant(n/10); }
```

3.

For this question, your answers will be functions that produce plain text output, not graphics. You may use either `cout` or `print`, your choice. Remember that with ordinary text output, every character has exactly the same width, and a line of text has a fixed maximum width.

Consider a line to be exactly 99 characters wide. To print a single character at the right-hand edge, you would need to print 98 spaces first.

a.

Write a function that receives one integer parameter, let's call it N . The function should print a row of exactly N stars, perfectly centered on a line of text. N will always be odd. For example, if the parameter is 5, we should see

```
*****
```

b.

Write a function *that uses your answer to part a* to print an upright triangle X lines high and $2X-1$ characters wide, perfectly centered. X represents the function's one parameter. For example, if X is 4 we should see

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

c.

Write a function *that uses your answer to part b* to print a series of triangles of increasing size (starting from 1 and ending with H , the function's one parameter) perfectly aligned in a vertical stack. For example, if H is 5 we should see

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

It is rather like a christmas tree, sort of.

d. (part d is less important than the others. Spend your time wisely)

Write a function *that uses your answer to parts c and a* to print a simple “pine” tree exactly as for part d, with H stages. Beneath it, there should be a thin trunk of length L, and it should be growing out of a rectangular box of height BH and width BW, all made of stars, all perfectly aligned. H, L, BH, and BW are the four parameters to this function.

For example, if H is 5, L is 2, BH is 3, and BW is 5, we should see

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

You may assume that the parameter BW will always be odd. If it isn't, nobody cares what the result will look like.