Examination, Programming, bridging course, 2010-12-17
Time, 8.00-13.00

Materials allowed: None

Each problem is worth 5 points.

For grade 4:

C++ students: Do problem 1, 3 and 5
Java students: Do problem 1, 3 and 7
Fortran students: Do problem 1, 3 and the grade 4 version of 9 and 10

For grade 5:

C++ students: Do problem 1, 2, 3, 4, 5 and 6
Java students: Do problem 1, 2, 3, 4, 7 and 8
Fortran students: Do problem 1, 2, 3, 4 and the grade 5 version of 9 and 10

To get the grade 4 you should complete 2/3 of the problems marked for grade four. To get the grade 5 you must complete 2/3 of all the problems.

If you achieve neither, you will get grade 3.
1. Python

(a) What is the output of the Python code below?

```python
for i in range(3):
    print i,

print '\n'
for i in range(3):
    print i,
    for j in range(2):
        print j,

print '\n'
print (1 <= 1)
print (1 >= 1)
print (1 != 1)
```

(b) Write a function that generates the Fibonacci series, which is defined by $F_n = F_{n-1} + F_{n-2}$ with starting values $F_0 = 0$ and $F_1 = 1$. Use 'yield' to return values.

2. Python

(a) Let’s say a record with grocery items and its prices should be created. Suggest an appropriate data structure for storing this. The record could look like this:

<table>
<thead>
<tr>
<th>Item</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soda</td>
<td>5</td>
</tr>
<tr>
<td>Bread</td>
<td>20</td>
</tr>
<tr>
<td>Butter</td>
<td>25</td>
</tr>
</tbody>
</table>

(b) List at least two programming tasks when it’s appropriate to use Python as a programming language and two tasks when it’s not appropriate.

(c) What are regular expressions?
3. C

A positive integer number is called a **perfect number** if the sum of all numbers that are evenly divisible by the number (including the number 1 but excluding the number itself) is equal to the number.

**Example:**
The number 28 is a perfect number because 28 is evenly divisible by the numbers 1, 2, 4, 7 and 14 and the sum of these is 28.

**Task:**
Write a C-function that has a positive integer number as parameter and that tests whether the number is perfect or not. The function should return the answer.
Write a main function that prints all perfect numbers less than one million (1000000) using the function described above.

4. C

The dot product between two arrays x and y with n elements is defined as

\[ x \cdot y = \sum_{i=1}^{n} x_i \cdot y_i \]

Write a function in C that takes two arrays as parameters (more parameters are allowed if needed) that calculates and returns the dot product of the arrays. The function must check that the size of the arrays are the same, if not, some kind of error should occur.
Write a main function that creates two arrays, fills them with values, calls the function described above and prints the result. The elements of the arrays should be of type `double`
Assume the we have a class `Sphere` defined

class Sphere {
private:
    double x,y,z; // (x,y,z) = centre point
    double r;    // radius
public:
    // Parameterless constructor
    Sphere() {
        ....
    }

    // Other constructor
    Sphere(double x, double y, double z, double r) {
        ....
    }

    // Print a Sphere
    void print() {
        ....
    }

    // Return the volume
    double volume() {
        ....
    }

    // Do we collide with another Sphere
    bool collide(const Sphere& other) {
        ....
    }
}

Write the following methods in the class above;

- The constructor with three parameters, declared above
- The method `volume`, that returns the volume of a sphere \( v = \frac{4\pi r^3}{3} \)
- The method `print`
- The method `collide`, that checks if the distance between the two centrepoints is less or equal to the sum of their radius. If so true is returned, otherwise false.
• A main-method that creates two spheres, calls the method `collide` with these spheres and prints the result of this call

6. **C++**

(a) Assume that you are writing a software module that acts as a message relay, receiving messages and passing them on to other units. Also assume that it might not always be possible to directly retransmit the messages. Which STL template would you recommend for temporarily storing the messages before they are retransmitted? Explain the reasoning behind your choice.

(b) What are iterators used for?

(c) Write a template class for storing a graph structure, but without defining any functions. That is, only the internal data structure should be defined. A graph consists of nodes and edges that connect the nodes. In this case a node will hold an object and an edge will just denote that two nodes are directly connected to each other. More specifically, a node will contain an instance of the type (MyClass in the example below) used to define the template class.

Below is an example of how the Graph template could be used later on, when an addNode function has been added. This example is just to illustrate how it’s supposed to be used, you don’t have to implement the addNode function.

```c++
// Objects that will be nodes in the graph.
MyClass a, b, c, d;
// Lists with neighbors to each node, for creating the edges.
vector<MyClass*> nb, nc, nd;

// Setup graph structure
Graph<MyClass> graph;

graph.addNode(&a, NULL);
   nb.push_back(&a);
   graph.addNode(&b, &nb);
   nc.push_back(&a);
   graph.addNode(&c, &nc);
   nd.push_back(&a);
   nd.push_back(&b);
   graph.addNode(&d, &nd);
```
7. Java

Assume the we have a class Sphere defined

```java
public class Sphere {
    private double x, y, z; // (x, y, z) = centre point
    private double r; // radius

    // Parameterless constructor
    public Sphere() {
        ....
    }

    // Other constructor
    public Sphere(double x, double y, double z, double r) {
        ....
    }

    // Textual representation of a Sphere
    public String toString() {
        ....
    }

    // Return the volume
    public double volume() {
        ....
    }

    // Do we collide with another Sphere
    public boolean collide(Sphere other) {
        ....
    }
}
```

Write the following methods in the class above;

- The constructor with three parameters, declared above
- The method `volume`, that returns the volume of a sphere \(v = 4\pi r^3/3\)
- The method `toString`
- The method `collide`, that checks if the distance between the two centrepoints is less or equal to the sum of their radius. If so true is returned, otherwise false.
- A main-method that creates two spheres, calls the method `collide` with these spheres and prints the result of this call
8. Java

The Least-Square method calculates a line \( y = ax + b \) from a number of \((x, y)\) samples so that we minimize the sum of the squares of the errors. This means that we should minimize

\[
s = \sum_{i=1}^{n} (y_i - (ax_i + b))^2
\]  

(1)

Doing this we get

\[
a = \frac{(n \sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i)}{D}
\]

\[
b = \frac{(\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i y_i \sum_{i=1}^{n} x_i)}{D}
\]

\[
D = n \sum_{i=1}^{n} x_i^2 - \left( \sum_{i=1}^{n} x_i \right)^2
\]

Using this, write a class \textbf{Samples} that has two arrays, \( x \) and \( y \), as attributes. More attributes are allowed if needed.

It should be possible to initialize the arrays using proper constructors. The class should contain a toString method that returns a textual representation of the object.

There should also be a method that applies the least square method described above and the returns the resulting coefficients \( a \) and \( b \).

The arrays and the resulting coefficients \( a \) and \( b \) are of type \texttt{double}
9. **Fortran**

Write a subroutine declared as

```fortran
subroutine findtrailing( a, b, c )
```

Here, `a` is an integer variable, (grade 5 see below), with a value $0 - 99$, and `b` is an array (possibly long) containing positive integers, all of which having values that are at least 1000. The parameter `c` is a pointer, that can point to an array of integers. The subroutine should go through the array `b` and check which elements have a value that (in the decimal system) end with the same two digits as `a`. The subscripts of these elements should be returned in an array pointed to by `c`. This array has to be allocated by the subroutine.

For 4 points on this problem (aiming for grade 4), the formulation of the problem is as above. For 5 points (aiming for grade 5), `a` should be a (short) array of values between 0 and 99. The resulting `c` should point to a two-dimensional array of subscripts, Row $j$ in this array should contain the subscripts of `b`-elements that end in the same digits as $a(j)$. Note that this two-dimensional array should be a normal array, i.e., all rows should have the same length, although not all of them might be filled with nonzero elements.

10. **Fortran**

A data file, named `secdata.dat`, is structured so that it contains data values in sections as shown in the following example:

```
'PART1' 4
 .
 4 lines each containing one real number
 .
'SECTDATA' 7
 .
 7 lines each containing one real number
 .
'END' 0
```

Note that the file above is just an example of how the file could look like, there can be an arbitrary number of sections. Each section is headed by a line containing a string, which is the name of the section followed by an integer giving the number of lines in the section. Each line in the section contains one real data value. After the last section comes a line with the name ‘END’ and an arbitrary integer value.

Write a main program that reads the file, and for each section of data finds the sum of the values contained in that section. The program should write its output to a file `secdata.out`. For each section the name of the section and the sum of the
values should be written on one line of the output file.

For 4 points, (aiming for grade 4), the problem can be solved according to the specifications above. For 5 points (aiming for grade 5), the program should handle the situation that the file does not contain a trailing line with 'END' xx, but instead end-of-file follows directly after the last section.