Q1: Concepts and Definitions

What is the difference between computer hardware and computer software?

What is an input device. List three example computer input devices.

What is the difference between main memory and secondary memory? Between ROM and RAM?

Difference between algorithm and a program? Between high-level versus low-level languages?

List any 5 of the various phases in software development.
Very briefly, explain what type of activity is performed at each of the phases you listed

List three advantages of using the formal software development methodology

Difference between objects and classes in object-oriented programming? Between methods and data? Between superclass and subclass?

Difference between C and C++?

State three reasons for declaring a data type for our program variables?

List three of the available data types in C++. Given an example declaration for each of the data types listed.

Which of the following are NOT valid identifiers in C++?

myName my_name my_Name my–Name myName2 2ndName
my.name Myname MYNAME 1234 abcd x1 x

Which of the following are NOT reserved keywords in C++?

Float Int case equal equals char else Min min max default constant
(b) Given the following declarations, answer the following questions:

```c
int a;
float b;
double c;
```

Explain the type conversions that will occur as the following statement is executed:

```c
c=a+b;
```

Explain the type conversions that will occur as the following statement is executed:

```c
a=b+c;
```

Explain the type conversions that will occur as the following statement is executed:

```c
b=a+c;
```

What will be the data type for the result of the following statement?

```c
a*b*c;
```

Q2: Errors in C++ programs or C++ statements

Valid statements, programs with errors, etc.

Example

In each of the following program segments, find the error(s) in the program.

```c
int Main()
{
    int Values;
    cout >> “Please enter the values” >> endl ;
    cin << value
    cout >> “Value entered is : “ >> value >> endl;
    return 0
}
```
//function definition for function that determines the minimum between two numbers

function minimum(number 1, number2)
{
    int min;
    min = number1;
    if (number2 < min)
    {
        min = number2;
        return min;
    }
}

Q3: True or False Questions

For each of the following, indicate whether the assertion is TRUE or FALSE.

<table>
<thead>
<tr>
<th>s/n</th>
<th>Assertion</th>
<th>True or False</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>In C++, a function allowed to call other functions in a different file.</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>A reserved keyword is also an identifier in C++</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>In C++, only one selection control structure is allowed - the if … else selection structure.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>There are only two basic constructs for writing structured programs - the selection structure and the repetition structure.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>It is illegal to place variable declarations anywhere in a C++ program – i.e. before or after the variable is used.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Some C++ Language operators are evaluated from left to right, while others are evaluated from right to left.</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>It is illegal to pre-processor directives to include other files in a C++ program</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>The default case is optional and hence may be absent in a switch…case statement</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>In C++, it is legal for two variables in the same function to have the same name (i.e. the same identifier).</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Any valid ASCII character is also a valid symbol for a C++ identifier</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Variable names used in a prototype declaration should always correspond to the variable names used in the definition of the function.</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>A variable can be initialized at the time it is declared</td>
<td></td>
</tr>
</tbody>
</table>

Some more True and False

<table>
<thead>
<tr>
<th>s/n</th>
<th>Assertion</th>
<th>True or False</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>The at function in the string class returns the position where an input search string is located</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>const variables are variables whose values cannot change during program execution</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Variables declared as int usually require more storage than double variables</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>In general, C++ functions are only accessible from within the file where they are defined</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>A function prototype is always necessary before a function can be invoked in a program, unless the function is defined before it is invoked</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Any selection structure (if, if/else, and switch) can always be replaced by one or more simple if structures</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>The assign function is a valid function (i.e. method) in the string class</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>A function prototypes are the same as a function definition</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>The function main() must always come before always other functions in the program text</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>The parenthesis operator () has the highest precedence of all C operators</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>In general, the unary forms of the arithmetic operators (+, -) always have a higher precedence than the corresponding binary operators</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Every C++ program must contain the function called main()</td>
<td></td>
</tr>
</tbody>
</table>
**Question 4: Program Outputs**

Generally, results of simple expressions – such as those given in the notes, in the class, or in the exercises

Another example:
List the results that will be produced if the following program is executed: (Ignore any syntax error)

```cpp
#include <iostream>

int average(int m, int n);
float average2(int m, int n);

int main()
{
    int m, n;
    m=5; n=2;
    cout << "First average : ";
    cout << average(m,n) << endl ;
    cout << endl;
    cout << "Second average : ";
    cout << average2(m,n) << endl;
}

int average(int m, int n)
{
    int result;
    result = (m + n)/2;
    return result;
}

float average2(int m, int n)
{
    float result;
    result = (m + n)/2;
    return result;
}
```

**Question 5: Writing a Program**

Generally, you will be given a simple formula and will be asked to write a short program to implement the formula. Sample formulae would be those for distances, volume of solids, area, simple interest, etc.

For this problem, you will not be required to do the problem analysis stage. You will only need to write a code to implement the given formula.