**1.** What is a computation?

Computation is an act of producing outputs based on inputs.

**2.** What do we mean by inputs and outputs to a computation? Give examples.

Input is used to produce outputs. E.g. in fuction call “square(7)” 7 is an input (argument) and “49” is an output (result)

**3.** What are the three requirements a programmer should keep in mind when expressing computations?

Expressed computation should be correct, simple and efficient.

1. Correctness - the computation has to do what it it supposed to do
2. Simplicity - the program needs to be as much simple as it can be in case other programmers will need to modify it
3. Efficient - the computation should be done in the fastest and the most efficient way

**4.** What does an expression do?

Expression performs an act of computation based on two operands (inputs)

**5.** What is the difference between a statement and an expression, as described in this chapter?

Expression is a single act of computation, where statement is a block of expression statements or declarations.

**6.** What is an lvalue? List the operators that require an lvalue. Why do these operators, and not the others, require an lvalue?

Lvalue is a value that can appear on the left side of an assignment. Lval is used by assignment, increment and decrement operator.

**7.** What is a constant expression?

Constant expression is a value that can’t be changed after it was assigned to the object.

**8.** What is a literal?

A literal is a value represented directly in the code ( a fixed value).

**9.** What is a symbolic constant and why do we use them?

Symbolic constant is a named object to which we can’t give a new value after it was initialized.

**10.** What is a magic constant? Give examples.

A magic constant is a value that is represented directly in the code without a description of what it supposed to do. For example: we can set a number of pages in our program to 25 but if we want to change it later, it would be hard because we will have to use “find and replace” method or search it manual. To prevent this, we can setup a variable with the desirable number of pages and then simply change the value of it when we want.

**11.** What are some operators that we can use for integers and floating-point values?

It is +,-,\*,:, >, <, =>, <=, =, ==, !== compound assignment, <<, >>

**12.** What operators can be used on integers but not on floating-point numbers?

% modulo (remainder)

**13.** What are some operators that can be used for strings?

<<, >>, >,<,=>,<=, + (concatenation), +=

**14.** When would a programmer prefer a switch-statement to an if-statement?

When we need to compare a value against several constants.

**15.** What are some common problems with switch-statements?

We can’t compare string values and we can’t use variables in case labels. There must be only constant expressions.

**16.** What is the function of each part of the header line in a for-loop, and in what sequence are they executed?

for (int i = 0 (initialization of counter); i < 40 (the condition of finishing the loop); i++ (the change of counter performed on every iteration).

**17.** When should the for-loop be used and when should the while-loop be used?

The for-loop provides much clearer declaration than while-loop. However, when we don’t know exact number of iterations, we should use while-loop.

**18.** How do you print the numeric value of a char?

It is possible to do by assigning the value of char object to integer object.

**19.** Describe what the line char foo(int x) means in a function definition.

char is a type of result of function’s execution, foo is a name of function, int is a type of argument and x is an argument

**20.** When should you define a separate function for part of a program? List reasons.

When this part of code needs to be used in other place of program

When this code produces result that we want to use in other parts of our program

When this code needs to be logical separated

**21.** What can you do to an int that you cannot do to a string?

With string we can’t use such operators as -,%,:,\*,+. We can’t decrement and increment string object.

**22.** What can you do to a string that you cannot do to an int?

We can’t hold as much data in an object of type integer as we want. Also we can’t concatenate two integer values.

**23.** What is the index of the third element of a vector?

2

**24.** How do you write a for-loop that prints every element of a vector?

for (int i = 0; i < vector.size(); i++)

**25.** What does vector<char> alphabet(26); do?

It creates a vector with the name “alphabet” and type char containing 26 elements.

**26.** Describe what push\_back() does to a vector.

push\_back() is a member function that adds a new element in the end of the vector

**27.** What do vector’s member functions begin(), end(), and size() do?

begin() refers to the first element of vector and end() refers to the past-the-last element, which doesn’t exist but we imagine that this element follows the last element of vector. This functions are used to refer to the range of vector. Function size() outputs the number of elements in vector.

**28.** What makes vector so popular/useful?

We can hold a lot of data in vector which we can sort and output according to our needs.

**29.** How do you sort the elements of a vector?

With the function sort. sort(vector.begin(),vector.end())