**1.** Why would you want a graphical user interface?

When the user should be able to manipulate the objects on the screen by means of pointing, clicking, dragging and dropping, hovering. The program with graphical interface can be much easier in use than a program without GUI, but command-line programs are often faster as the software layer between the user and program logic tends to be much lighter.

**2.** When would you want a non-graphical user interface?

When the input and output of the program is only textual and there is no need for graphics to represent these data.

**3.** What is a software layer?

A program that connects different parts of the system to produce a reasonable output.

**4.** Why would you want to layer software?

Layers can be modified separately so there is no need to recreate the whole system if some part of it should be changed. Often, different layers are written by different people, in different time and with different programming languages. In order to connect this all in one system, the concept of layers should be used.

**5.** What is the fundamental problem when communicating with an operating system from C++?

Programming language, which was used to write this operating system, can be different from C++, so the OS won’t recognise the concepts of classes, class methods, members etc.

**6.** What is a callback?

A function that is called when a particular event occurs (e.g. button click, window resizing etc.)

**7.** What is a widget?

A form of interaction with a program (Button, Input box, Checkbox etc.)

**8.** What is another name for widget?

Control

**9.** What does the acronym FLTK mean?

The Fast, Light Toolkit

**10.** How do you pronounce FLTK?

fulltick

**11.** What other GUI toolkits have you heard of?

GTK+, Qt, wxWidgets

**12.** Which systems use the term *widget* and which prefer *control*?

There is no direct difference between those 2 notions as their meaning is similar and interchangeable.

**13.** What are examples of widgets?

Button, Menu, Inbox, Outbox, Checkbox, Radio Button

**14.** When would you use an inbox?

When we need to get data from the user.

**15.** What is the type of the value stored in an inbox?

The value of type string. However, the inbox can also return the value of type integer.

**16.** When would you use a button?

To provide a form of control to the user.

**17.** When would you use a menu?

When we need to define a set of buttons that can be logically grouped.

**18.** What is control inversion?

It means that the order of execution is determined not by the program, but by the user.

**19.** What is the basic strategy for debugging a GUI program?

Use a debugger and be extra careful while writing the code. The development approach should be iterative and each small step of program creation should be carefully tested. In case of errors, the code can be compared with already working one. The code should be simple and clear enough to be understood by the others.

**20.** Why is debugging a GUI program harder than debugging an “ordinary program using streams for I/O”?

Because the order execution is determined by the user and a programmer can barely imagine all possible execution orders. Also, it can be hard to keep track of connection between widgets and data, especially in big applications.