PROGRAMMING ASSIGNMENT HELP.

Programming assignment help

This article for those who are just starting or plans to start a programmer's career. The things described below will seem to someone obvious, however, this information will be useful for beginners. Technical English at the basic level

Almost all technical documentation is written in English. The source code, in most industries, write in English. The most relevant events in the field of IT are highlighted, first of all, in English-speaking resources. If you want to become a good specialist in demand in the labor market, then English needs to know at least at the level of "read the documentation" and "write a bug report". Version control systems

Each programmer is obliged to understand the principle and be able to work at least with one versions control system. Not bad introductory material - the first three chapters of the Pro Git book. Blind text set method

When you sit 8 hours in the workplace, wind your head from the keyboard to the monitor is a rather tedious occupation. The ability to dial blindly helps to better focus on the task being solved. Hands write code "on the machine" and the brain does not spend extra time on switching the context between the set of text and thinking about the algorithm. There are good keyboard simulators that will help you learn to print blindly. For example, solo on the keyboard or VSE10.RU. Mathematics

This item is not required for many industries. But in some directions without mathematics can not do. For example, the development of gaming engines requires good knowledge of geometry (work with vectors, intersection of figures), mathematical analysis (search for integrals and derivatives), physics. When building optimization systems, the calculation of optimal routes, we will not go far without knowing discrete mathematics. In engine learning and financial sector, this is also the necessary requirement in order to become a full-fledged specialist in the field. The ability to separate the main thing from the secondary

Imagine that you create a site for sale tickets. In the process of product development, there is a desire to add some new features. Whether it is 10 additional payment systems, a finished personal account or notifications about new tickets. The further moving in the creation of something, the more similar desires arise to improve the functionality. Initially, you planned to spend 4 months to develop, then roll out the site in combat mode. Due to the occurrence of new requirements, the timing can easily move two or three times, and you will simply be rushing in the stack of unfinished and tasks and the floor of the working code.

At the same time, another developer will collect the minimum working system for 2 months, where only the functionality of the ticket order and online payment via Webmoney is implemented. Perhaps a month later, his service will appear two or three regular customers who will begin to bring the project to self-sufficiency. Already in the process of working with live people, the developer will understand which features on the site people are really needed, and will gradually add useful functions to your service.

And at this time you will program the functionality, which may not be so in demand, as you expected.

Often, the developer begins to be taken too far and forgets about the main thing. For example, it can be made to the ideal form of comments on the site, when the development of some important sections has not even started.