

Ef þú átt í vandræðum með að finna tiltekna stafi á lyklaborðinu þínu í prófinu þá eru hér nokkrir til að afrita / *If you have problems finding specific characters on your keyboard, here are a few to copy:* () { } [] < > = & | / \ ! "

1. Set (50%)

Lýsing á íslensku

Í þessu verkefni eigið þið að útfæra hugræna gagnatagið `Set` sem einfaldan tengdan lista af hnútum. Hnútarnir eru af taginu `Node` (sjá `Node.h`) sem er sniðmátsklasi og því er `Set` líka sniðmátsklasi. `Set` á að vera með header (sentinel) hnút sem geymir engin gögn en bendir þá á fyrsta hnútinn í tengda listanum.

Þið þurfið eingöngu að útfæra föllin í skránni `Set.cpp` (`Set.h` er gefin) og megið ekki breyta öðrum skráum. Þrjú prófunartilvik eru í skránni `input.txt` og rétt úttak í skránni `output.txt`.

Til að athuga hvort forritið ykkar er laust við minnisvillur getið þið keyrt skrána `valgrind.sh` sem er í rótarmöppu verkefnisins. Opnið `File Browser`, tvísmellið á `valgrind.sh` og veljið `Run in Terminal`.

English description

In this project, you need to implement the abstract data type `Set` as a simply linked list of nodes. The nodes are of the type `Node` (sjá `Node.h`) which is a template class and therefore `Set` is also a template class. `Set` should have a header (sentinel) node which does not contain any data, but points to the first node in the linked list.

You only need to implement the function in the file `Set.cpp` (`Set.h` is given) and you are not allowed to change other files.

Three test cases are given in the file `input.txt` and correct output in the file `output.txt`.

In order to check if your program is free of memory errors, you can run the file `valgrind.sh` which is in the root directory of the project. Open a `File Browser`, double-click `valgrind.sh` and select `Run in Terminal`.

2. PQueue (50%)

Lýsing á íslensku

Í þessu verkefni eigið þið að útfæra hugrænt gagnatag sem kallast forgangsbiðröð (PQueue). Forgangsbiðröð er biðröð þar sem hægt er að gefa hnútum ákveðinn forgang þegar þeim er bætt í biðröðina. Hnúturinn sem hefur hæstan forgang er fjarlægður fyrst úr biðröðinni. Ef tveir eða fleiri hnútar hafa jafn háan forgang þá er sá hnútur fjarlægður sem bætt var fyrst inn.

Hnúturnir í forgangsbiðröðinni eru af taginu `Pair` (sjá `Pair.h`). Fyrri tag parsins er `string` og seinna tag parsins er `integer`, sem táknar forgang.

Þið þurfið eingöngu að útfæra föllin í skránni `PQueue.cpp` (`PQueue.h` er gefin) og megið ekki breyta öðrum skráum. Prófunartilvik eru í skránni `input.txt` og rétt úttak í skránni `output.txt`.

Til að athuga hvort forritið ykkar er laust við minnisvillur getið þið keyrt skrána `valgrind.sh` sem er í rótarmöppu verkefnisins. Opnið `File Browser`, tvísmellið á `valgrind.sh` og veljið `Run in Terminal`.

English description

In this project, you need to implement an abstract data type called `priority queue` (PQueue). A priority queue is a queue where you can specify a priority when adding nodes. The nodes that have the highest priority are dequeued first. If two or more nodes have equally high priority, then the node dequeued is the one that was enqueued first.

The nodes are of the type `Pair` (see `Pair.h`). The first type of the pair is a `string` while the second type of the pair is an `integer`, denoting the priority.

You only need to implement the function in the file `PQueue.cpp` (`PQueue.h` is given) and you are not allowed to change other files.

Test cases are given in the file `input.txt` and correct output in the file `output.txt`.

In order to check if your program is free of memory errors, you can run the file `valgrind.sh` which is in the root directory of the project. Open a `File Browser`, double-click `valgrind.sh` and select `Run in Terminal`.