

OS Course – Assignment II

Abstract

Your assignment is to write a *Vector* class which stores *integer* objects. The class should enable addition and removal of objects along with several other methods (defined in the header file). The idea is that the number of objects is not predefined, so you can't allocate in advance the amount of memory you will need.

Guidelines

The purpose of the exercise is to explore several dynamic allocation schemes. Since implementation of the *Vector* class is very easy, you will be graded both on implementation and a test *main*, and on the dynamic allocation scheme you use.

You are required to develop one scheme which you think is best but you need to provide documentation for other schemes you have considered and ruled out (state what caused you to rule them out).

Your project needs to be efficient in both the number of allocations it performs as well as the method of reallocation (whether you copy the data each time you reallocate, how much space do you allocate each time more space is required etc.).

You should implement the *vector.h* header file and you can't use any non-standard objects from C++ libraries. However, if you think you need additional objects (which you're going to develop yourself); you can implement them after verifying it with me.

Submission (17/04/2005)

Submission will be done in groups of **2** or **3** ONLY. I will not accept a work that is submitted alone.

You should submit your source files, a printout of the code and a document explaining your chosen allocation scheme as well as others you have not implemented.

Grading will be based on: program readability (i.e. comments, meaningful variable names etc.), correctness and efficiency as described above.