

Randomly Generated Integers

For this computer assignment, you are to write a C++ program to generate random integers in the range [LOW = 1, HIGH = 10000] and to store them in a vector < int > of size VEC_SIZE = 250. Then, sort the contents of the vector (in ascending order) and display it on stdout.

To sort the contents of a vector, use the sort () function in the STL. In addition to the main () routine, implement the following subroutines in your program:

- void genRndNums (vector < int > & v) : This routine generates VEC_SIZE integers and puts them in vector v. Initializes the random number generator (RNG) by calling the function srand () with the seed value SEED = 1, and generates random integers by calling the function rand ().
- void printVec (const vector < int > & v) : This routine displays the contents of vector v on stdout, printing exactly NO_ITEMS = 12 numbers on a single line, except perhaps the last line. The sorted numbers need to be properly aligned on the output. For each printed number, allocate ITEM_W = 5 spaces on stdout.

Programming Notes:

- You are not allowed to use any I/O functions from the C library, such as scanf or printf. Instead, use the I/O functions from the C++ library, such as cin or cout.
- Name your source file as prog1.cc and your header file as prog1.h. Guard the statements in your header file using the following format. (This is necessary because you don't want the statements in a header file are processed more than once.)

```
#ifndef CONST-VALUE // which is not defined any place else
#define CONST-VALUE // same const value as for ifndef directive
```

```
// put all statements for your header file here
```

```
#endif
```

- Include all system header files (that you need in your program) in your header file. For example, to gain access to the iostream library, which defines a set of simple I/O operations, insert the line #include <iostream> in your header file, to gain access the I/O manipulators such as setw, insert the line #include <iomanip> in your header file; to access the vector container in the STL, insert the line #include <vector> in your header file; and to use the functions in the STL such as sort, insert the line #include <algorithm> in your header file. At the top of your source file, insert your header file by the following statement: #include "prog1.h".

- To compile and link your program with the system library routines execute: `Make N=1`. (Assuming you've already a link to the script `Make` in directory: `~cs340/bin` from your bin directory.) To test your program, execute: `Make execute N=1`. This command executes your program and displays its output as well as any error messages both on the terminal screen and in `prog1.out`. You can find all related files for this computer assignment in directory: `~cs340/progs/17f/p1`. After you are done with the program, you wouldn't need its object and executable files any more. To delete them, execute: `Make clean`.
- Let `v` be a vector of integers, then the call: `sort (v.begin (), v.end ())` sorts the elements of `v` in ascending order. The detailed description of the `sort ()` routine can be found on the course website and in the course textbook.
- Execute the `srand ()` function (only once) before generating the first random integer with the seed value `SEED`. The `rand ()` function generates a random integer in the range `[0, RAND_MAX]`, where the constant value `RAND_MAX` is the largest random integer returned by the `rand ()` function and its value is system dependent. To normalize the return value to a value in the range `[LOW, HIGH]`, execute: `rand () % (HIGH - LOW + 1) + LOW`.
- You can find the correct output of this program in file `prog1.out` in directory: `~cs340/progs/17f/p1`.
- When your program is ready, submit your source and header file to your TA by executing: `mail_prog.340 prog1.cc prog1.h`.