<https://www.onlinegdb.com/>

To count the number of running processes in a system using C++, you would generally need to interact with the operating system's process management facilities. This approach varies depending on the OS. Here's a simple example for a Unix-like system, using the **popen** function to execute the **ps** command and count the lines of output, which represent the processes:

cppCopy code

#include <iostream>

#include <cstdio>

#include <memory>

#include <stdexcept>

#include <string>

#include <array>

int countProcesses() {

std::array<char, 128> buffer;

std::string result;

std::unique\_ptr<FILE, decltype(&pclose)> pipe(popen("ps -A | wc -l", "r"), pclose);

if (!pipe) {

throw std::runtime\_error("popen() failed!");

}

while (fgets(buffer.data(), buffer.size(), pipe.get()) != nullptr) {

result += buffer.data();

}

return std::stoi(result);

}

int main() {

try {

std::cout << "Number of running processes: " << countProcesses() << std::endl;

} catch (const std::exception& e) {

std::cerr << "Error: " << e.what() << std::endl;

return 1;

}

return 0;

}

This code runs the **ps -A** command, which lists all running processes, pipes the output to **wc -l**, which counts the number of lines (i.e., processes), and then reads and returns this number in C++. Remember, this is specific to Unix-like systems (like Linux and macOS); the approach would be different for other operating systems like Windows.

**References:**

1. [stackoverflow.com - How to count number of running process with given name?](https://stackoverflow.com/questions/64657941/how-to-count-number-of-running-process-with-given-name)
2. [cplusplus.com - Count specific process name](https://cplusplus.com/forum/general/253506/)
3. [somacon.com - Count Instances of Process or Application in C++](https://www.somacon.com/p112.php)
4. [codeproject.com - display total count of process running with the same name](https://www.codeproject.com/Questions/991014/display-total-count-of-process-running-with-the-sa)
5. [reddit.com - How do I calculate how many total process a program has](https://www.reddit.com/r/C_Programming/comments/11f1swt/how_do_i_calculate_how_many_total_process_a/)
6. [cplusplus.com - print number of active process in linux](https://cplusplus.com/forum/unices/221977/)

In a Windows system, counting the number of running processes can be achieved using the Windows API. The following is a basic example in C++ that uses the **CreateToolhelp32Snapshot**, **Process32First**, and **Process32Next** functions to enumerate all processes and count them:

#include <windows.h>

#include <tlhelp32.h>

#include <iostream>

int countProcesses() {

HANDLE hProcessSnap;

PROCESSENTRY32 pe32;

int count = 0;

// Take a snapshot of all processes in the system.

hProcessSnap = CreateToolhelp32Snapshot(TH32CS\_SNAPPROCESS, 0);

if (hProcessSnap == INVALID\_HANDLE\_VALUE) {

std::cerr << "CreateToolhelp32Snapshot failed." << std::endl;

return -1;

}

// Set the size of the structure before using it.

pe32.dwSize = sizeof(PROCESSENTRY32);

// Retrieve information about the first process.

if (!Process32First(hProcessSnap, &pe32)) {

std::cerr << "Process32First failed." << std::endl;

CloseHandle(hProcessSnap);

return -1;

}

// Now walk the snapshot of processes, and increment the count for each process.

do {

count++;

} while (Process32Next(hProcessSnap, &pe32));

CloseHandle(hProcessSnap);

return count;

}

int main() {

std::cout << "Number of running processes: " << countProcesses() << std::endl;

return 0;

}

This code creates a snapshot of all processes in the system, then iterates through them using **Process32First** and **Process32Next**, counting each one. Note that this approach is specific to Windows, as it uses Windows-specific APIs.

**References:**

1. [support.microsoft.com - Which version of Windows operating system am I running?](https://support.microsoft.com/en-us/windows/which-version-of-windows-operating-system-am-i-running-628bec99-476a-2c13-5296-9dd081cdd808)
2. [en.wikipedia.org - Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows)
3. [edu.gcfglobal.org - Windows Basics: All About Windows](https://edu.gcfglobal.org/en/windowsbasics/all-about-windows/1/)
4. [microsoft.com - Experience the Power of Windows 11 OS, Computers, & ...](https://www.microsoft.com/en-us/windows)
5. [intel.com - How to Determine Your Operating System](https://www.intel.com/content/www/us/en/support/articles/000018246/emerging-technologies.html)
6. [support.microsoft.com - View your system info - Windows 10](https://support.microsoft.com/en-us/windows/view-your-system-info-a965a8f2-0773-1d65-472a-1e747c9ebe00)