

一、实验目标

- 1) 熟悉标准库的容器使用及容器的性能;
- 2) 熟悉标准库的算法;
- 3) 熟悉 C++ 的接口与实现相分离的做法与实践;

二、实验内容

在 WEB 上, 计算机是通过 IP 地址进行标识的, 一个 IP 地址是一个 32 位的整数, 不过通常我们会把 32 位的整数分成 4 个八位的数值, 例如我们学校的邮件服务器地址为

202.120.224.10

转换成整数就是 3396919306。而我们对数字不是很敏感, 更容易记住有意义的名字, 例如

Mail.fudan.edu.cn

这个就是学校的邮件服务器的名称。因此, WEB 上的任何一台机器, 通常有一个主机名(网络术语叫做域名), 还有对应的 IP 地址。整个 internet 网络要正常运行的话, 必须要把域名转换成 IP 地址——我们的每一次浏览器访问都可能发生好多次这样的转换, 如果有转换不成功的话, 我们就不能访问 internet。

本次实验设计一个域名查找程序, 在本地提供服务。我们的目标是设计一个高效率的域名程序, 使得查找、删除、创建效率都非常高。

本次实验, 提供给大家的文件包括:

文本文件: ns20k.txt, ns40k.txt, ns80k.txt, ns224k.txt

源文件: nstest.cpp ns_common.h nsvector.cpp nsmap.cpp nsumap.cpp

批处理文件: auto.bat

此外, 还使用了我们写好的 stopwatch 和 split 这两个库文件。

1) 文本文件的内容包含多行, 每一行中包含域名和对应的 IP 地址, 中间使用空格分隔, 如下所示

coloradovtc.com 427663021 # 在第 20000 行

文本文件中包含 20k/40k/80k/224k 分别表示 2 万个/4 万个/8 万个/22.4 万个名称与 IP 的对——实际的域名已经超过几千万个——因此效率至关重要。

我们的数据来自[1], 把 2020/4/14 和 2020/4/15 日的数据放在一起, 然后通过脚本语言生成了现在的文本文件。

2) ns_common.h 是一个接口文件, 考虑到我们会使用 vector/map/unordered_map 分别来实现域名服务程序, 我们规定不管那一种实现, 都必须实现这些接口函数, 具体如下

```
using HostName = std::string;
using IPAddress = unsigned int;
const IPAddress NON_EXISTING_ADDRESS = 0;

void create(std::istream& in);
bool contain(const HostName&);
void insert(const HostName&, const IPAddress&);
bool remove(const HostName&);
```

```
IPAddress lookup(const HostName&);
```

此外，我们把所有的代码放在名字空间 EXP4 中，以保证我们使用的名字和其他的名字不冲突。

3) nsvector.cpp, nsmap.cpp 和 nsumap.cpp 中已经有部分代码，请根据下面的要求补充完整。

nsvector.cpp 使用 vector 实现名称服务程序。具体地，把名字和 IP 地址作为一个 pair 放在 vector 中。在实现 contain/remove/lookup 时，不能使用循环，要使用 STL 库函数 find_if，而且第三个参数使用 lambda 函数的形式。关于 lambda 函数的更多信息请参考[2]。

nsmap.cpp 使用 map 实现名称服务程序。其中 map 的 key 是名称，value 是 IP 地址。需要实现 ns_common.h 中的所有函数。

nsumap.cpp 使用 unordered_map 实现名称服务程序，其中关于 unordered_map 的详细信息参考[3]。需要实现 ns_common.h 中的所有函数。

4) nstest.cpp 包含测试程序，主要流程是：读入数据；查找在数据库中的主机名；查找不在数据库中的主机名。在每一个步骤都输出相应的信息。

5) auto.bat 是批处理程序（**只能在 windows 下面运行**），执行该程序，会自动进行编译链接并运行，最终的效果如下图：

```
Z:\course\2020S\00P\hw\00P4\done>nsvector 0<ns40k.txt 1>NUL
test create:      using time 14916ms
test contains:    using time 1486ms
test not contains:using time 1286ms

Z:\course\2020S\00P\hw\00P4\done>nsmap      0<ns40k.txt 1>NUL
test create:      using time 334ms
test contains:    using time 2ms
test not contains:using time 3ms

Z:\course\2020S\00P\hw\00P4\done>nsumap    0<ns40k.txt 1>NUL
test create:      using time 311ms
test contains:    using time 1ms
test not contains:using time 1ms
```

提交内容：nstest.cpp, ns_common.h 已经是完整的，不用修改，也不用提交；大家需要提交的是 nsvector.cpp, nsmap.cpp 和 nsumap.cpp 三个文件。在提交前，确保每个程序都可以正常运行。

参考资料

1. 新注册的域名。参考 <https://whoisds.com/newly-registered-domains>
2. Lambda 表达式。参考 <https://zh.cppreference.com/w/cpp/language/lambda>
3. unordered_map。参考 https://zh.cppreference.com/w/cpp/container/unordered_map