

# 高考志愿填报系统分析报告

## 高考志愿填报系统分析报告

依赖项

第一题

思路

重要代码

将文理科分开，分别计算每个专业历年平均分和最低分的有效平均值（有效指的是只考虑有效数值），并将其与专业用 `map` 进行映射。

令 `map` 按值排序。

输出前十个元素与后十个元素

结果与结论

第二题

思路

重要代码

创建`map`：年份，分数，分段累计人数

将分数替换为排名

输出

结果

第三题

思路

重要代码

`map`：专业，学院，学科，历年排名

`map`：专业，学院，学科，历年分数

`map`：往年的专业以及今年的专业

得到考生排名

遍历各专业历年排名，如果有一年平均值80%-120%之间的排名包含考生排名，那么值得选择

得到对应的大类

结果

## 依赖项

高考志愿填报系统主要基于C++ STL。用到的Library如下：

```
#include <iostream>
#include <map>
#include <vector>
#include <fstream>
#include <string>
#include <set>
#include <algorithm>
#include <iomanip>
```

## 第一题

第一题要求我们分析中山大学考生填报的“十大热门专业”和“十大冷门专业”并提供相关的指标。

## 思路

我的思路分为如下几步：

1. 清理数据。原数据存在如下几个问题
  1. 原格式为 UTF16，并不是 C++ 接受的格式。
  2. 大多数专业的数据并不完整，需要标记空白数据。因此我将 txt 格式文件转换为 csv 格式文件以后使用 Python pandas 进行了数据的清洗（并未体现在代码中）。
2. 将文理科分开，分别计算每个专业历年平均分和最低分的有效平均值（有效指的是只考虑有效数值），并将其与专业用 map 进行映射。
3. 按 value 对 map 进行排序，输出热门专业和冷门专业。

## 重要代码

部分重要函数如下：

**将文理科分开，分别计算每个专业历年平均分和最低分的有效平均值（有效指的是只考虑有效数值），并将其与专业用 map 进行映射。**

```
map<string, int> ProfessionAndAverageScore_Liberal,
    ProfessionAndAverageScore_Science;
ifstream finput("中山大学2018-2020年本科批次录取统计（广东）.txt");
string profession = "", academy = "", subject = "";
vector<int> score;
int average_score = 0, temp1 = 0, sum = 0;

//输入数据，创建键值对
while (!finput.eof())
{
    if (finput.fail()) break;
    //处理文科,共31个学院
    for (int i = 1; i <= 31; i++)
    {
        //每阅读一行以后将profession academy subject average_score以及score初始化
        profession = academy = subject = "";
        average_score = 0;
        score.clear();
        temp1 = sum = 0;

        //依次输入专业，学院，文理科情况
        finput >> profession >> academy >> subject;
        //将历年分数输入并计算出有效平均值
        for (int j = 1; j <= 6; j++)
        {
            finput >> temp1;
            score.emplace_back(temp1);
        }
        int count = 0;
        /*for (int l = 1; l <= 6; l++)
        {
            if (score[l] != 0)
            {
                count++;
            }
        }*/
    }
}
```

```

    }*/
    for (auto k = score.begin(); k != score.end(); k++)
    {
        if ((*k) != 0)
            count++;
        sum += (*k);
    }
    if (count == 0)
    {
        cout << profession;
    }
    else{ average_score = sum / count; }

    //添加键值对
    ProfessionAndAverageScore_Liberal.insert({ profession,average_score
});
}
//处理理科，共86个专业
for (int i = 1; i <= 86; i++)
{
    //每阅读一行以后将profession academy subject average_score以及score初始化
    profession = academy = subject = "";
    average_score = 0;
    score.clear();
    temp1 = sum = 0;

    //依次输入专业，学院，文理科情况
    finput >> profession >> academy >> subject;
    //将历年分数输入并计算出有效平均值
    for (int j = 1; j <= 6; j++)
    {
        finput >> temp1;
        score.emplace_back(temp1);
    }
    int count = 0;
    for (auto k = score.begin(); k != score.end(); k++)
    {
        if (*k != 0)
            count++;
        sum += (*k);
    }
    if (count == 0)
    {
        cout << profession;
    }
    else { average_score = sum / count; }
    //添加键值对
    ProfessionAndAverageScore_Science.insert({ profession,average_score
});
}

```

## 令map按值排序。

```
//便于令创建的map按值排序
bool cmp(pair<string, int> a, pair<string, int> b)
{
    return a.second < b.second;
}
//按value而不是key排序
vector< pair<string, int> > SortByValue_Up(map<string, int> mp)
{
    vector< pair<string, int> > vec;
    for (map<string, int>::iterator it = mp.begin(); it != mp.end(); it++)
    {
        vec.push_back(pair<string, int>(it->first, it->second));
    }
    sort(vec.begin(), vec.end(), cmp);
    return vec;
}
```

## 输出前十个元素与后十个元素

```
//输出前十个元素
void Output(vector< pair<string, int> > vec)
{
    int i = 1;
    auto it = vec.begin();
    it++;
    while (it != vec.end() && i <= 10)
    {
        cout << "    " << "    " << i << ":" << (*it).first << " 六项有效平均分: "
        << (*it).second << endl;
        it++;
        i++;
    }
}
//输出后十个元素
void Output_Reverse(vector< pair<string, int> > vec)
{
    int i = 1;
    auto it = vec.rbegin();
    while (it != vec.rend() && i <= 10)
    {
        cout << "    " << "    " << i << ":" << (*it).first << " 六项有效平均分: "
        << (*it).second << endl;
        it++;
        i++;
    }
}
```

## 结果与结论

文科热门冷门专业情况如下：

#### 十大热门专业

- 1:经济学类（含金融学、经济学专业） 六项有效平均分：635
- 2:工商管理 六项有效平均分：628
- 3:经济学类（含国际商务、管理科学、金融学、经济学专业） 六项有效平均分：626
- 4:金融学 六项有效平均分：621
- 5:工商管理类（含工商管理、会计学专业） 六项有效平均分：621
- 6:英语 六项有效平均分：620
- 7:会计学 六项有效平均分：620
- 8:新闻传播学类（含新闻学、传播学专业） 六项有效平均分：619
- 9:社会学 六项有效平均分：619
- 10:法学 六项有效平均分：619

#### 十大冷门专业

- 1:旅游管理类（含旅游管理、会展经济与管理专业） 六项有效平均分：609
- 2:图书情报与档案管理类（含信息管理与信息系统、图书馆学、档案学专业） 六项有效平均分：609
- 3:外国语言文学类（含英语、阿拉伯语、西班牙语、朝鲜语、俄语专业） 六项有效平均分：609
- 4:国际政治 六项有效平均分：610
- 5:哲学类（含哲学、逻辑学专业） 六项有效平均分：611
- 6:社会学类（含社会学、人类学、考古学专业） 六项有效平均分：612
- 7:外国语言文学类（含英语、德语、法语、日语专业） 六项有效平均分：612
- 8:经济学类（含经济学、金融学、工商管理、会计学专业） 六项有效平均分：613
- 9:公共管理类（含行政管理、政治学与行政学专业） 六项有效平均分：614
- 10:历史学 六项有效平均分：614

理科热门冷门专业情况如下：

#### 十大热门专业

- 1:临床医学（八年制） 六项有效平均分：660
- 2:经济学类（含国际商务、管理科学、金融学、经济学专业） 六项有效平均分：659
- 3:口腔医学（5+3） 六项有效平均分：653
- 4:计算机类（含计算机科学与技术、网络空间安全、信息与计算科学、软件工程专业） 六项有效平均分：652
- 5:经济学类（含金融学、经济学专业） 六项有效平均分：650
- 6:工商管理类（含工商管理、会计学专业） 六项有效平均分：648
- 7:口腔医学 六项有效平均分：647
- 8:法学 六项有效平均分：647
- 9:管理科学与工程类（含国际商务、管理科学专业） 六项有效平均分：646
- 10:软件工程 六项有效平均分：645

#### 十大冷门专业

- 1:地理信息科学 六项有效平均分：600
- 2:生物科学 六项有效平均分：600
- 3:土木类（含土木工程、水文与水资源工程专业） 六项有效平均分：601
- 4:水文与水资源工程 六项有效平均分：610
- 5:土木工程 六项有效平均分：610
- 6:海洋工程与技术 六项有效平均分：611
- 7:理论与应用力学 六项有效平均分：612
- 8:交通工程 六项有效平均分：612
- 9:航空航天工程 六项有效平均分：612
- 10:化学工程与工艺 六项有效平均分：613

C:\Users\Legion\Desktop\CEL-Volunteering\Debug\CEL-Volunteering.exe (进程 1268)已退出，代码为 0。  
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。  
按任意键关闭此窗口。...

## 第二题

### 思路

第二题要求我们通过“2018-2020每年的一分一段”数据，完整补充“中山大学2018-2020年本科批次录取统计（广东）”中缺少的平均分和最低分的排位信息。由于一分一段表中的每一个分数的排名都已经给出，因此我们只需要用排名替换“中山大学2018-2020年本科批次录取统计（广东）”中的分数，本质上就是一个查找的过程。

最后我们将结果输出到一个txt文件中，方便第三问使用。

### 重要代码

#### 创建map：年份，分数，分段累计人数

```
void Calculate(ifstream& finput,int year, map<int, map<int, int>>& mp)
{
    int score, rank;
    int num;
    //map<int, map<int, int>> mp;
    while (!finput.eof())
    {
        finput >> score >> num >> rank;
        if (finput.fail()) break;
```

```

        mp[year][score] = rank;
    }
    mp[year][0] = 0;
}

```

## 将分数替换为排名

```

map<string, vector<int>> professionAndRankings;
map<string, string> professionAndAcademy;
map<string, string> professionAndSubject;
vector<int> rank;

while (!finput.eof())
{
    if (finput.fail()) break;
    for (int i = 1; i <= 31; i++)
    {
        finput >> profession >> academy >> subject
            >> min_2020 >> avg_2020 >> min_2019 >> avg_2019
            >> min_2018 >> avg_2018;
    }
    for (int i = 1; i <= 86; i++)
    {
        rank.clear();

        finput >> profession >> academy >> subject
            >> min_2020 >> avg_2020 >> min_2019 >> avg_2019
            >> min_2018 >> avg_2018;

        rank.push_back(scoresAndRankings[2020][min_2020]);
        rank.push_back(scoresAndRankings[2020][avg_2020]);
        rank.push_back(scoresAndRankings[2019][min_2019]);
        rank.push_back(scoresAndRankings[2019][avg_2019]);
        rank.push_back(scoresAndRankings[2018][min_2018]);
        rank.push_back(scoresAndRankings[2018][avg_2018]);
        professionAndAcademy[profession] = academy;
        professionAndSubject[profession] = subject;
        professionAndRankings[profession] = rank;
    }
}

```

## 输出

```

ofstream foutput("理科排名-中山大学2018-2020年本科批次录取统计(广东).txt");
foutput << "专业" << "\t" << "学院" << "\t" << "分科" << "\t"
    << "2020年最低排位" << "\t" << "2020年平均排位" << "\t"
    << "2019年最低排位" << "\t" << "2019年平均排位" << "\t"
    << "2018年最低排位" << "\t" << "2018年平均排位" << endl;

for (auto it = professionAndRankings.begin(); it !=
professionAndRankings.end(); it++)
{
    foutput << it->first << "\t" << professionAndAcademy[it->first] << "\t"
    << professionAndSubject[it->first] << "\t";
    for (int i = 0; i < 6; i++)

```

```
{
    foutput << it->second[i] << "\\t";
}
foutput << endl;
}

cout << "已输出到CEL-Volunteering/CEL-Volunteering文件夹中";
```

## 结果

理科排名-中山大学2018-2020年本科批次录取统计（广东） - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

专业	学院	分科	2020年最低排位	2020年平均排位	2019年最低排位	2019年平均排位	2018年最低排位	2018年平均排位
保密管理	数据科学与计算机学院（软件学院）	理科	3749	3057	3804	2974	0	0
材料类（含材料物理、材料化学、高分子材料与工程专业）	材料科学与工程学院	理科	6008	5127	5748	4248	5398	4292
材料类（含材料物理、材料化学专业）	材料学院	理科	8426	7314	6621	5974	6631	5780
城乡规划	地理科学与规划学院	理科	8747	7030	0	0	0	0
大气科学类（含大气科学、应用气象学专业）	大气科学学院	理科	8426	7030	7785	5974	6863	6188
地理科学类（含自然地理与资源环境、人文地理与城乡规划、地理信息科学、地理科学、城乡规划专业）	地理科学与规划学院	理科	4673	3218	4975	3954	0	0
地理科学类（含自然地理与资源环境、人文地理与城乡规划、地理信息科学、地理科学专业）	地理科学与规划学院	理科	0	0	4422	0	0	0
地理信息科学	测绘科学与技术学院	理科	0	22215	6834	0	0	0
地质学类（含地质学、地质工程专业）	地球科学与工程学院	理科	7856	7314	7785	6834	7079	6631
电子信息类（含电子信息科学与技术、通信工程、微电子科学与工程、光电信息科学与工程（工学）专业）	电子与信息工程学院（微电子学院）	理科	0	0	2372	1840	0	0
电子信息类（含电子信息科学与技术、通信工程专业）	电子与信息工程学院（微电子学院）	理科	0	0	2372	1840	0	0
法学	法学院	理科	1923	1612	2141	1579	1918	1581
法医学	中山医学院	理科	4891	3397	6834	5537	4973	4292
工商管理	管理学院	理科	0	0	3111	1517	0	0
工商管理类（含工商管理、会计学专业）	国际金融学院	理科	0	0	0	0	5159	2974
公共管理类（含行政管理、政治学与行政学专业）	政治与公共事务管理学院	理科	3920	3057	3370	3111	3393	2992
管理科学与工程类（含国际商务、管理科学专业）	岭南学院	理科	0	0	1447	856	0	0
光电信息科学与工程（工学）	电子与信息工程学院（微电子学院）	理科	0	0	3499	2841	0	0
光电信息科学与工程（理学）	物理学院	理科	0	0	3804	0	0	0
国际政治	国际关系学院	理科	4673	5974	4975	5967	5398	0
海洋工程与技术	海洋工程与技术学院	理科	8426	8145	9391	8858	6863	6188
海洋科学	海洋科学学院	理科	8426	6769	9391	7785	6863	6188
汉语言文学	中国语言文学系	理科	2178	1923	2841	1840	3122	2022
航空航天工程	航空航天大学	理科	0	0	5748	4975	0	0
航空航天类（含航空航天工程、理论与应用力学专业）	航空航天大学	理科	8426	6769	6621	5537	0	0
核工程类（中法合作办学）（核工程与核技术专业）	中法核工程与技术学院	理科	8426	6769	9114	6834	6863	5585
化学工程与工艺	化学工程与技术学院	理科	7856	7030	9114	7538	7079	6188
化学类（含化学、高分子材料与工程专业）	化学学院	理科	4464	6399	4422	5398	4133	0
环境科学与工程类（含环境科学、环境工程专业）	环境科学与工程学院	理科	6490	5338	5748	4422	5585	4292
会计学	管理学院	理科	0	0	2606	1374	0	0
基础医学	中山医学院	理科	2178	2178	4099	3804	3680	3680
计算机类（含计算机科学与技术、网络空间安全、信息与计算科学、软件工程专业）	数据科学与计算机学院	理科	4891	3749	4422	2974	0	0
计算机类（含计算机科学与技术、网络空间安全专业）	数据科学与计算机学院	理科	0	0	1447	1084	0	0
计算机类（含智能科学与技术、交通工程专业）	智能工程学院	理科	0	0	6199	4422	0	0
交通工程	智能工程学院	理科	0	0	2974	1840	0	0
金融学	国际金融学院	理科	0	0	1057	841	1154	913
经济学类（含国际商务、管理科学、金融学、经济学专业）	岭南学院	理科	0	0	1224	673	0	0
经济学类（含金融学、经济学专业）	岭南学院	理科	0	0	1224	673	0	0
经济学类（含经济学、金融学、工商管理、会计学专业）	国际金融学院	理科	3218	2774	3954	2974	0	0
口腔医学	光华口腔医学院	理科	1519	1238	1840	1579	2299	1918
口腔医学（5+3）	光华口腔医学院	理科	1057	841	1293	1084	2109	1295
理论与应用力学	航空航天大学	理科	0	0	5974	4796	0	0
历史学	历史学系（珠海）	理科	7314	6490	8273	6621	0	0
临床医学	中山医学院	理科	3057	2052	3254	2372	3535	2299
临床医学（八年制）	中山医学院	理科	915	655	1154	771	917	660
旅游管理类（含旅游管理、会展经济与管理专业）	旅游学院	理科	6008	5562	9114	5974	4454	3824
逻辑学	哲学系	理科	0	0	3804	3111	0	0
软件工程	数据科学与计算机学院（软件学院）	理科	0	0	1741	1517	0	0
生态学	生态学院	理科	6008	5338	9114	6834	0	0
生物科学	农学院	理科	0	22215	7070	0	0	0
生物科学类（含生物科学、生物技术、生态学专业）	生命科学学院	理科	4673	3397	4975	3370	5398	3824
生物医学工程	生物医学工程学院	理科	8426	6769	5974	5336	6863	5585
数学类（含数学与应用数学、统计学专业）	数学学院	理科	2178	1717	2489	1840	2658	2109
数学类（含数学与应用数学、信息与计算科学专业）	数学学院（珠海）	理科	3920	3218	3954	3254	4292	3393
水文与水资源工程	土木工程学院	理科	0	0	6199	5336	0	0
图书情报与档案管理类（含图书馆学、档案学专业）	资讯管理学院	理科	0	0	3111	2974	0	0
图书情报与档案管理类（含信息管理与信息系统、图书馆学、档案学专业）	资讯管理学院	理科	6008	5127	9114	5159	0	0
土木、水利与海洋工程	土木工程学院	理科	8145	7030	0	0	0	0
土木工程	土木工程学院	理科	0	6199	5159	0	0	0
土木类（含土木工程、水文与水资源工程专业）	土木工程学院	理科	0	21673	6621	0	0	0
外国语言文学类（含英语、阿拉伯语、西班牙语、朝鲜语、俄语专业）	国际翻译学院	理科	4464	3567	4975	3499	3263	2870
微电子科学与工程	微电子科学与技术学院	理科	5773	4673	6834	4099	0	0

## 第三题

第三题要求我们结合“2021年广东省高考普通类（物理）一分一段”和“中山大学2021年分专业招生计划（广东）”，输入考生的分数或排名，输出推荐填报的2021年中山大学大类专业志愿以及参考依据。

## 思路

我的思路分为以下几点：

- 分析输入：题目要求我们输入高考成绩或者是排名，鉴于高考分数会随着题目难度升降，我选择将高考分数也转换成排名。
- 分析评价“是否推荐的指标”：一般来说，省内985高校各专业的分数线比较稳定（因为招生多）。于是在这里我选取的评价指标是：如果考生排名在该专业历年排名的80%-120%之间则视为可以填报。这样既不会浪费分数也有较大概率能够进入该专业。
- 具体操作：

1. 参考 T2 的操作，建立 map: 2021年高考分数->2021年高考排名
2. 建立 map: 专业, 历年排名, 历年分数, 学院, 学科, 今年大类
3. 遍历 map: 专业->历年排名选取合适的专业。
4. 建立 set 得到各个专业对应的大类（不允许重复）
5. 将推荐填报的大类专业输出到控制台，理由输出到文件中。

## 重要代码

### map: 专业, 学院, 学科, 历年排名

```
Files_CloseAndOpen("理科排名-中山大学2018-2020年本科批次录取统计（广东）.txt", finput);
vector<int> rank;
string profession = "", academy = "", subject = "";
string garbish;
int min_2020, avg_2020, min_2019, avg_2019, min_2018, avg_2018;
while (!finput.eof())
{
    getline(finput, garbish);

    for (int i = 1; i <= 75; i++)
    {
        rank.clear();
        finput >> profession >> academy >> subject
            >> min_2020 >> avg_2020 >> min_2019 >> avg_2019
            >> min_2018 >> avg_2018;

        rank.push_back(min_2020);
        rank.push_back(avg_2020);
        rank.push_back(min_2019);
        rank.push_back(avg_2019);
        rank.push_back(min_2018);
        rank.push_back(avg_2018);
        professionAndAcademy[profession] = academy;
        professionAndSubject[profession] = subject;
        professionAndRankings[profession] = rank;
    }
}
```

### map: 专业, 学院, 学科, 历年分数

```
Files_CloseAndOpen("中山大学2018-2020年本科批次录取统计（广东）.txt", finput);
vector<int> scores;
while (!finput.eof())
{
    scores.clear();
    finput >> profession >> academy >> subject
        >> min_2020 >> avg_2020 >> min_2019 >> avg_2019
        >> min_2018 >> avg_2018;
    if (finput.fail()) break;
    scores.push_back(min_2020);
    scores.push_back(avg_2020);
    scores.push_back(min_2019);
    scores.push_back(avg_2019);
    scores.push_back(min_2018);
}
```



```

        scores.push_back(avg_2018);
        professionAndScores[profession] = scores;
    }

```

## map:往年的专业以及今年的专业

```

Files_CloseAndOpen("往年专业与2021年专业对应表.txt", finput);
string type;
while (!finput.eof())
{
    finput >> profession >> academy >> subject
        >> min_2020 >> avg_2020 >> min_2019 >> avg_2019
        >> min_2018 >> avg_2018 >> type;
    if (finput.fail()) break;
    professionAndTypes[profession] = type;
}

```

## 得到考生排名

```

int distinguish;
int ranks;
int score;
int inf, sup;
cout << "高考志愿填报系统" << endl
    << "\t请选择输入方式: " << endl
    << "\t\t1. 输入高考成绩" << endl
    << "\t\t2. 输入高考排名" << endl;
cin >> distinguish;
if (distinguish == 1)
{
    cout << "请输入成绩: " << endl;
    cin >> score;
    ranks = Cumulate_ScoreAndRank_2021[2021][score];
}
else
{
    cout << "请输入排名: " << endl;
    cin >> ranks;
}

```

## 遍历各专业历年排名，如果有一年平均值80%-120%之间的排名包含考生排名，那么值得选择

```

vector<string> Candidate_Profession;
for (auto it = professionAndRankings.begin(); it !=
professionAndRankings.end(); it++)
{
    int sup, inf;
    int flag = 0;
    for (int i = 1; i <= 5; i += 2)
    {
        sup = (*it).second[i] * 1.2;
        inf = (*it).second[i] * 0.8;
    }
}

```

```

        if ((inf <= ranks ) && (ranks <= sup))
        {
            flag = 1;
            break;
        }
    }
    if (flag == 1) Candidate_Profession.push_back(it->first);
}

```

## 得到对应的大类

```

set<string> Candidate_Types;
for (auto it2 = Candidate_Profession.begin(); it2 !=
Candidate_Profession.end(); it2++)
{
    Candidate_Types.emplace(*it2);
}
if (Candidate_Types.empty())
{
    cout << "中山大学没有适合你的大类，请选择其他学校!" << endl;
}
else
{
    //输出对应结果
    cout << endl << "建议填报的大类如下: " << endl;
    for (auto it3 = Candidate_Types.begin(); it3 != Candidate_Types.end();
it3++)
    {
        cout << "\t" << *it3 << endl;
    }

    ofstream foutput("建议填报理由.txt");
    cout << "理由(近三年对应专业的录取分数和排名)见文件“建议填报理由.txt”" << endl;
    foutput << "专业" << "\t" << "学院" << "\t" << "分科" << "\t" << "大类" <<
"\t"
        << "2020年最低排位" << "\t" << "2020年平均排位" << "\t"
        << "2019年最低排位" << "\t" << "2019年平均排位" << "\t"
        << "2018年最低排位" << "\t" << "2018年平均排位" << endl;
    for (auto it4 = Candidate_Profession.begin(); it4 !=
Candidate_Profession.end(); it4++)
    {
        foutput << *it4 << "\t" << professionAndAcademy[*it4] << "\t" <<
professionAndSubject[*it4] << "\t" << professionAndTypes[*it4] << "\t"
            << professionAndRankings[*it4][0] << "\t" <<
professionAndRankings[*it4][1] << "\t"
            << professionAndRankings[*it4][2] << "\t" <<
professionAndRankings[*it4][3] << "\t"
            << professionAndRankings[*it4][4] << "\t" <<
professionAndRankings[*it4][5] << endl;
    }
}
}

```

结果

Microsoft Visual Studio 调试控制台

高考志愿填报系统  
请选择输入方式：  
1. 输入高考成绩  
2. 输入高考排名  
1  
请输入成绩：  
657  
建议填报的大类如下：  
电子信息类（含电子信息科学与技术、通信工程、微电子科学与工程、光电信息科学与工程（工学）专业）  
电子信息类（含电子信息科学与技术、通信工程专业）  
汉语言文学  
基础医学  
金融学  
口腔医学  
临床医学  
数学类（含数学与应用数学、统计学专业）  
新闻传播学类（含新闻学、传播学专业）  
智能科学与技术  
理由(近三年对应专业的录取分数和排名)见文件“建议填报理由.txt”  
C:\Users\Legion\Desktop\CEL-Volunteering\Debug\CEL-Volunteering.exe (进程 21544) 已退出，代码为 0。  
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。  
按任意键关闭此窗口。 . . .

建议填报理由 - 记事本														
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)														
专业	学院	分科	大类	2020年最低排位	2020年平均排位	2019年最低排位	2019年平均排位	2018年最低排位	2018年平均排位					
电子信息类（含电子信息科学与技术、通信工程、微电子科学与工程、光电信息科学与工程（工学）专业）														
电子信息类（含电子信息科学与技术、通信工程专业）	电子与信息工程学院（微电子学院）	理科												
汉语言文学	中国语言文学系	理科		2178	2178	1923	2841	1840	3122	2022				
基础医学	中山医学院	理科		2178	2178	4099	3804	3680	3680					
金融学	国际金融学院	理科		0	0	2974	1840	0	0					
口腔医学	光华口腔医学院	理科		1519	1238	1840	1579	2299	1918					
临床医学	中山医学院	理科		3057	2052	3254	2372	3535	2299					
数学类（含数学与应用数学、统计学专业）	数学学院	理科					2178	1717	2489	1840	2658	2109		
新闻传播学类（含新闻学、传播学专业）	传播与设计学院	理科					2774	2307	3653	3111	3680	2765		
智能科学与技术	智能工程学院	理科		0	0	2489	2040	0	0					