

HƯỚNG DẪN BÀI TẬP THỰC HÀNH THỪA KẾ VÀ ĐA HÌNH

Một trường đại học (TRUONGDH) cần quản lý thông tin về việc cấp thưởng cho các đối tượng xuất sắc (DTXS) của các khoa như sau:

- Giảng viên (GV): mã số, họ tên, khoa và số bài báo khoa học. Mỗi bài báo khoa học sẽ được cấp thưởng 20 triệu. Ngoài ra, nhà trường còn thưởng thêm để hỗ trợ chi phí đăng bài như sau:
 - $3 \leq \text{SoBB} \leq 5$: thưởng thêm 10 triệu.
 - $\text{SoBB} \geq 6$: thưởng thêm 15 triệu.
- Sinh viên (SV): mã số, họ tên, khoa, điểm trung bình và điểm rèn luyện. Điểm trung bình tính theo thang điểm 10 và điểm rèn luyện tính theo thang điểm 100. Các SV có điểm rèn luyện từ 85 trở lên mới được xem xét cấp thưởng theo quy định:
 - $8.5 \leq \text{Điểm trung bình} \leq 9$: thưởng 5 triệu.
 - Điểm trung bình > 9 : thưởng 7 triệu.
- Nhân viên (NV): mã số, họ tên, khoa và số sáng kiến kinh nghiệm. Mỗi sáng kiến kinh nghiệm sẽ được cấp thưởng 3 triệu, tăng 20% số tiền cấp thưởng nếu có từ 3 sáng kiến kinh nghiệm trở lên.

Yêu cầu:

1. Vẽ sơ đồ các lớp.
2. Khai báo và định nghĩa các lớp để quản lý theo yêu cầu của trường.
3. Liệt kê d/s các GV có 5 bài báo khoa học trở lên.
5. Đếm số lượng các SV có điểm rèn luyện tối đa.
6. Tính tổng tiền cấp thưởng cho các NV có 3 sáng kiến kinh nghiệm trở lên.
7. Tính điểm trung bình của các SV có điểm rèn luyện dưới 50.
8. Kiểm tra trường có GV nào chưa có bài báo khoa học không?
9. Tìm các NV có số sáng kiến kinh nghiệm nhiều nhất.
10. Sắp xếp d/s các DTXS tăng dần theo mã số, giảm dần theo tiền thưởng.
11. Tìm kiếm DTXS theo: mã số, họ tên, khoa, tiền thưởng.
12. Cập nhật thông tin DTXS có mã số do người dùng nhập vào.
13. Thêm 1 DTXS mới.
14. Xóa 1 DTXS có mã số do người dùng nhập vào.
15. Xóa toàn bộ d/s DTXS.
16. Tạo file.
17. Đọc file.
18. Đọc file và lưu d/s DTXS.
19. Lưu file d/s DTXS.

Hướng dẫn:

1. Vẽ sơ đồ các lớp.



Quy ước:

- Ký hiệu -: phạm vi truy cập **private**
- Ký hiệu #: phạm vi truy cập **protected**
- Ký hiệu +: phạm vi truy cập **public**

2. Định nghĩa các lớp để quản lý theo yêu cầu của trường.

```
DTSX.h  ▢ ×
TruongDH (Global Scope)
1  #pragma once
2  #ifndef _DTXS
3  #define _DTXS
4  #include<iostream>
5  #include<string>
6  using namespace std;
7  //Lớp trừu tượng (Abstract class)
8  class DTXS
9  {
10 protected:
11     string maso, hoten, khoa;
12 public:
13     DTXS(string = "", string = "", string = "");
14     string GetMaSo();
15     string GetHoTen();
16     string GetKhoa();
17     void SetHoten(string);
18     void SetKhoa(string);
19     //Các hàm ảo
20     virtual void Nhap();
21     virtual void Xuat();
22     //Hàm ảo thuần túy - thuần ảo
23     virtual string GetLoai() = 0;
24     virtual long TinhTienThuong() = 0;
25 };
26 #endif
```

```
GV.h  ▢ ×
TruongDH (Global Scope)
1  #pragma once
2  #ifndef _GV
3  #define _GV
4  #include"DTSX.h"
5  class GV : public DTXS
6  {
7  private:
8     int sobb;
9  public:
10     GV(string = "", string = "", string = "", int = 0);
11     int GetSoBB();
12     void SetSoBB(int);
13     void Nhap();
14     void Xuat();
15     long TinhTienThuong();
16     string GetLoai();
17 };
18 #endif
```

```

SV.h  + X
TruongDH  (Global Scope)
1      #pragma once
2      #ifndef _SV
3      #define _SV
4      #include "DTSX.h"
5      class SV : public DTXS
6      {
7      private:
8          double dtb, drl;
9      public:
10         SV(string = "", string = "", string = "", double = 0, double = 0);
11         double GetDTB();
12         double GetDRL();
13         void SetDTB(double);
14         void SetDRL(double);
15         void Nhap();
16         void Xuat();
17         long TinhTienThuong();
18         string GetLoai();
19     };
20     #endif

```

```

NV.h  + X
TruongDH  (Global Scope)
1      #pragma once
2      #ifndef _NV
3      #define _NV
4      #include "DTSX.h"
5      class NV : public DTXS
6      {
7      private:
8          int sosk;
9      public:
10         NV(string = "", string = "", string = "", int = 0);
11         int GetSoSK();
12         void SetSoSK(int);
13         void Nhap();
14         void Xuat();
15         long TinhTienThuong();
16         string GetLoai();
17     };
18     #endif

```

```
1  #pragma once
2  #ifndef _TRUONGDH
3  #define _TRUONGDH
4  #include "DTXS.h"
5  #include <fstream>
6  class TRUONGDH
7  {
8  private:
9      DTXS** pdtxs;
10     int sl;
11 public:
12     TRUONGDH();
13     ~TRUONGDH();
14     int GetSL();
15     void Nhap();
16     void Xuat();
17     DTXS* Nhap1();
18     void LietKeGVCoTu5BaiBao();
19     int DemSVCoDRLToiDa();
20     long TinhTongTienThuongNVCo3SKKNTroLen();
21     double TinhDTBCuaSVCoDRLDuoi50();
22     bool KiemTraGVChuaCoBaiBao();
23     void TimNVCoSKKNNhieuNhat();
24     string DoiChuoiInHoa(string);
25     string DoiChuoiInThuong(string);
26     void SapXepMaSoTangDan();
27     void TimDTXSThuocKhoa(string);
28     void CapNhatThongTinTheoMaSo(string);
29     void Them(DTXS*,int);
30     int Xoa(string);
31     void GhiFile(string);
32     void DocFile(string);
33     void DocFileDTXS(string);
34     void GhiFileDTXS(string);
35 };
36 #endif
```


3. Xây dựng các phương thức xử lý cho các lớp trên.

```
XuLyDTXS.cpp x
TruongDH (Global Scope)
1  #include "DTSX.h"
2  DTXS::DTXS(string ms, string ht, string khoa) : maso(ms), hoten(ht), khoa(khoa) {}
3  string DTXS::GetMaSo() { return maso; }
4  string DTXS::GetHoTen() { return hoten; }
5  string DTXS::GetKhoa() { return khoa; }
6  void DTXS::SetHoten(string ht) { hoten = ht; }
7  void DTXS::SetKhoa(string kh) { khoa = kh; }
8  void DTXS::Nhap() {
9      cin.ignore();
10     cout << "Nhap ma so: ";
11     getline(cin, maso);
12     cout << "Nhap ho ten: ";
13     getline(cin, hoten);
14     cout << "Nhap khoa: ";
15     getline(cin, khoa);
16 }
17 void DTXS::Xuat() { cout << "MS: " << maso << "\tTen: " << hoten << "\tKhoa: " << khoa <<
    << "\tLoai: " << GetLoai() << "\tThuong: " << TinhTienThuong(); }
```

```
XuLyGV.cpp x
TruongDH (Global Scope)
1  #include "GV.h"
2  GV::GV(string ms, string ht, string khoa, int bb) : DTXS(ms, ht, khoa), sobb(bb) {}
3  int GV::GetSoBB() { return sobb; }
4  void GV::SetSoBB(int bb) {
5      while (bb < 0) {
6          cout << "Nhap so bai bao: ";
7          cin >> bb;
8      }
9      sobb = bb;
10 }
11 void GV::Nhap() {
12     DTXS::Nhap();
13     do {
14         cout << "Nhap so bai bao: ";
15         cin >> this->sobb;
16     } while (sobb < 0);
17 }
18 void GV::Xuat() {
19     DTXS::Xuat();
20     cout << "\tSoBB: " << sobb << endl;
21 }
22 long GV::TinhTienThuong() {
23     long thuong, them;
24     thuong = sobb * 20000000;
25     them = (sobb >= 6 ? 15000000 : (sobb >= 3 ? 10000000 : 0));
26     return thuong + them;
27 }
28 string GV::GetLoai() {
29     return "GV";
30 }
```

```

1  #include "SV.h"
2  SV::SV(string ms, string ht, string khoa, double tb, double r1) :DTXS(ms, ht, khoa), dtb(tb), drl(r1) {}
3  double SV::GetDTB() { return dtb; }
4  double SV::GetDRL() { return drl; }
5  void SV::SetDTB(double tb) {
6      while (tb < 0 || tb > 10) {
7          cout << "Nhap diem trung binh: ";
8          cin >> tb;
9      }
10     dtb = tb;
11 }
12 void SV::SetDRL(double r1) {
13     while (r1 < 0 || r1 > 100) {
14         cout << "Nhap diem ren luyen: ";
15         cin >> r1;
16     }
17     drl = r1;
18 }
19 void SV::Nhap() {
20     DTXS::Nhap();
21     do {
22         cout << "Nhap diem trung binh: ";
23         cin >> this->dtb;
24     } while (dtb < 0 || dtb > 10);
25     do {
26         cout << "Nhap diem ren luyen: ";
27         cin >> this->drl;
28     } while (drl < 0 || drl > 100);
29 }
30 void SV::Xuat() {
31     DTXS::Xuat();
32     cout << "\tDTB: " << dtb << "\tDRL: " << drl << endl;
33 }
34 long SV::TinhTienThuong() {
35     long thuong = 0;
36     if (drl >= 85)
37         thuong = (dtb > 9 ? 7000000 : (dtb >= 8.5 ? 5000000 : 0));
38     return thuong;
39 }
40 string SV::GetLoai() {
41     return "SV";
42 }

```

XuLyNV.cpp [X]

TruongDH (Global Scope)

```

1  #include "NV.h"
2  NV::NV(string ms, string ht, string khoa, int sk) :DTXS(ms, ht, khoa), sosk(sk) {}
3  int NV::GetSoSK() { return sosk; }
4  void NV::SetSoSK(int sk) {
5      while (sk < 0) {
6          cout << "Nhap so sang kien: ";
7          cin >> sk;
8      }
9      sosk = sk;
10 }
11 void NV::Nhap() {
12     DTXS::Nhap();
13     do {
14         cout << "Nhap so sang kien: ";
15         cin >> this->sosk;
16     } while (sosk < 0);
17 }
18 void NV::Xuat() {
19     DTXS::Xuat();
20     cout << "\tSoSK: " << sosk << endl;
21 }
22 long NV::TinhTienThuong() {
23     long thuong = sosk * 3000000;
24     if (sosk >= 3)
25         thuong = thuong * 1.2;
26     return thuong;
27 }
28 string NV::GetLoai() {
29     return "NV";
30 }

```


TruongDH

```

1  #include "TRUONGDH.h"
2  #include "GV.h"
3  #include "SV.h"
4  #include "NV.h"
5  TRUONGDH::~TRUONGDH() {
6      if (pdtxs != NULL)
7      {
8          for (int i = 0; i < sl; i++)
9              delete pdtxs[i];
10         delete[] pdtxs;
11         pdtxs = NULL;
12         sl = 0;
13     }
14 }
15 TRUONGDH::TRUONGDH() {
16     pdtxs = NULL;
17     sl = 0;
18 }
19 int TRUONGDH::GetSL() {
20     return sl;
21 }
22 void TRUONGDH::Nhap() {
23     int loai; //loại DTXS
24     do {
25         cout << "Nhap so luong DTSX: ";
26         cin >> sl;
27     } while (sl < 1);
28     pdtxs = new DTXS * [sl];
29     for (int i = 0; i < sl; i++)
30     {
31         cout << "-----\n";
32         cout << "Nhap thong tin DTXS thu " << i + 1 << ":\n";
33         do {
34             cout << "Nhap loai (1: GV, 2: SV, 3: NV): ";
35             cin >> loai;
36         } while (loai < 1 || loai > 3);
37         if (loai == 1)
38             pdtxs[i] = new GV;
39         else if (loai == 2)
40             pdtxs[i] = new SV;
41         else
42             pdtxs[i] = new NV;
43         pdtxs[i]->Nhap(); //tính đa hình
44     }
45 }
46 DTXS* TRUONGDH::Nhap1() {
47     int loai; //loại DTXS
48     DTXS* pdt;
49     do {
50         cout << "Nhap loai (1: GV, 2: SV, 3: NV): ";
51         cin >> loai;
52     } while (loai < 1 || loai > 3);
53     if (loai == 1)
54         pdt = new GV;
55     else if (loai == 2)
56         pdt = new SV;
57     else
58         pdt = new NV;
59     pdt->Nhap(); //tính đa hình
60     return pdt;
61 }
62 void TRUONGDH::Xuat() {
63     if (pdtxs == NULL)
64         cout << "D/s DTXS rong\n";
65     else
66         for (int i = 0; i < sl; i++)
67             pdtxs[i]->Xuat(); //tính đa hình
68 }

```

```

69 void TRUONGDH::LietKeGVCoTu5BaiBao() {
70     int dem = 0;
71     for (int i = 0; i < sl; i++)
72     {
73         if (pdtxs[i]->GetLoai() == "GV" && ((GV*)pdtxs[i])->GetSoBB() >= 5)
74         {
75             pdtxs[i]->Xuat(); //đã hình
76             dem++;
77         }
78         if (dem == 0)
79             cout << "Khong co GV nao co tu 5 bai bao tro len\n";
80     }
81 int TRUONGDH::DemSVCoDRLToiDa() {
82     int dem = 0;
83     for (int i = 0; i < sl; i++)
84         if (pdtxs[i]->GetLoai() == "SV" && ((SV*)pdtxs[i])->GetDRL() == 100)
85             dem++;
86     return dem;
87 }
88 long TRUONGDH::TinhTongTienThuongNVCo3SKKNTroLen() {
89     long sum = 0;
90     for (int i = 0; i < sl; i++)
91         if (pdtxs[i]->GetLoai() == "NV" && ((NV*)pdtxs[i])->GetSoSK() >= 3)
92             sum += pdtxs[i]->TinhTienThuong();
93     return sum;
94 }
95 double TRUONGDH::TinhDTBCuaSVCoDRLDuoi50() {
96     double tongdtb = 0;
97     int dem = 0;
98     for (int i = 0; i < sl; i++)
99         if (pdtxs[i]->GetLoai() == "SV" && ((SV*)pdtxs[i])->GetDRL() < 50)
100         {
101             tongdtb += ((SV*)pdtxs[i])->GetDTB();
102             dem++;
103         }
104     return dem > 0 ? tongdtb / dem : 0;
105 }
106 bool TRUONGDH::KiemTraGVChuaCoBaiBao() {
107     bool kt = false; //Giả sử không có GV chưa có bài báo
108     for (int i = 0; i < sl; i++)
109         //Nếu tìm thấy GV thứ i chưa có bài báo
110         if (pdtxs[i]->GetLoai() == "GV" && ((GV*)pdtxs[i])->GetSoBB() == 0)
111         {
112             kt = true; //Ghi nhận tình trạng có GV chưa có bài báo
113             break;
114         }
115     return kt;
116 }
117 void TRUONGDH::TimNVCoSKKNNhiềuNhat() {
118     int skmax = -1, vt = -1; //Không có NV
119     //Tìm NV đầu tiên (nếu có)
120     for (int i = 0; i < sl; i++)
121         if (pdtxs[i]->GetLoai() == "NV") //Có NV
122         {
123             vt = i; //Ghi nhận vị trí NV đầu tiên
124             skmax = ((NV*)pdtxs[i])->GetSoSK(); //Ghi nhận số SKKN của NV đầu tiên là skmax
125             break;
126         }
127     if (vt != -1) //Nếu có NV đầu tiên
128     {
129         //Tìm số SKKN max
130         for (int i = vt + 1; i < sl; i++)
131             if (pdtxs[i]->GetLoai() == "NV" && ((NV*)pdtxs[i])->GetSoSK() > skmax)
132                 skmax = ((NV*)pdtxs[i])->GetSoSK(); //Ghi nhận số SKKN của NV nhiều hơn => skmax
133         //Liệt kê d/s các NV có số SKKN max
134         for (int i = 0; i < sl; i++)
135             if (pdtxs[i]->GetLoai() == "NV" && ((NV*)pdtxs[i])->GetSoSK() == skmax)
136                 pdtxs[i]->Xuat();
137     }
138     else //Không có NV
139         cout << "Truong khong co NV";
140 }

```

```

140 string TRUONGDH::DoiChuoiiInHoa(string str) {
141     //Đổi từng ký tự của chuỗi str thành ký tự IN HOA
142     for (int i = 0; i < str.length(); i++)
143         str[i] = toupper(str[i]);
144     return str;
145 }
146 string TRUONGDH::DoiChuoiiInThuong(string str) {
147     //Đổi từng ký tự của chuỗi str thành ký tự in thường
148     for (int i = 0; i < str.length(); i++)
149         str[i] = tolower(str[i]);
150     return str;
151 }
152 void TRUONGDH::SapXepMaSoTangDan() {
153     for (int i = 0; i < sl - 1; i++)
154         for (int j = i + 1; j < sl; j++)
155             if (DoiChuoiiInHoa(pdtxs[i]->GetMaSo()) > DoiChuoiiInHoa(pdtxs[j]->GetMaSo()))
156             {
157                 DTXS* ptam = pdtxs[i];
158                 pdtxs[i] = pdtxs[j];
159                 pdtxs[j] = ptam;
160             }
161 }
162 void TRUONGDH::TimDTXSThuocKhoa(string kh) {
163     int dem = 0;
164     for (int i = 0; i < sl; i++)
165         if (DoiChuoiiInHoa(pdtxs[i]->GetKhoa()) == DoiChuoiiInHoa(kh))
166         {
167             pdtxs[i]->Xuat();
168             dem++;
169         }
170     if (dem == 0)
171         cout << "Khong co DTXS thuoc Khoa " << kh << endl;
172 }
173 void TRUONGDH::CapNhatThongTinTheoMaSo(string ms) {
174     int timthay = -1; //Chưa tìm thấy DTXS có mã số ms
175     for (int i = 0; i < sl; i++)
176         if (DoiChuoiiInHoa(pdtxs[i]->GetMaSo()) == DoiChuoiiInHoa(ms))
177         {
178             timthay = 1; //đã tìm thấy DTXS có mã số ms
179             string ht, kh;
180             cin.ignore();
181             cout << "Nhap ho ten: ";
182             getline(cin, ht);
183             pdtxs[i]->SetHoten(ht); //Cập nhật họ tên
184             cout << "Nhap khoa: ";
185             getline(cin, kh);
186             pdtxs[i]->SetKhoa(kh); //Cập nhật khoa
187             if (pdtxs[i]->GetLoai() == "GV") //Nếu DTXS đang xét là GV
188             {
189                 int bb;
190                 cout << "Nhap so bai bao: ";
191                 cin >> bb;
192                 ((GV*)pdtxs[i])->SetSoBB(bb); //Cập nhật số bài báo
193             }
194             else if (pdtxs[i]->GetLoai() == "SV") //Nếu DTXS đang xét là SV
195             {
196                 double tb, r1;
197                 cout << "Nhap DTB: ";
198                 cin >> tb;
199                 ((SV*)pdtxs[i])->SetDTB(tb); //Cập nhật DTB
200                 cout << "Nhap DRL: ";
201                 cin >> r1;
202                 ((SV*)pdtxs[i])->SetDRL(r1); //Cập nhật DRL
203             }
204             else //DTXS đang xét là NV
205             {
206                 int sk;
207                 cout << "Nhap so SKKN: ";
208                 cin >> sk;
209                 ((NV*)pdtxs[i])->SetSoSK(sk); //Cập nhật số SKKN
210             }
211             cout << "Da cap nhat thong tin DTXS co ma so la " << ms << endl;
212             Xuat(); //Xuất lại d/s DTXS sau khi đã cập nhật
213             break;
214         }
215     if (timthay == -1)
216         cout << "Khong tim thay DTXS co ma so la " << ms << endl;
217 }

```

```

218 void TRUONGDH::Them(DTXS* pdt, int vt) {
219     DTXS** pnew = new DTXS * [sl + 1];
220     for (int i = 0; i < vt - 1; i++)
221         pnew[i] = pdtxs[i];
222     pnew[vt - 1] = pdt;
223     for (int i = vt; i < sl + 1; i++)
224         pnew[i] = pdtxs[i - 1];
225     delete[]pdtxs;
226     pdtxs = pnew;
227     sl++;
228 }
229 int TRUONGDH::Xoa(string ms) {
230     int vt = -1; //Chưa tìm thấy DTXS có mã số ms
231     for (int i = 0; i < sl; i++)
232         if (DoiChuoiInHoa(pdtxs[i]->GetMaSo()) == DoiChuoiInHoa(ms))
233         {
234             vt = i;
235             break;
236         }
237     if (vt != -1)
238     {
239         DTXS** pnew = new DTXS * [sl - 1];
240         for (int i = 0; i < vt; i++)
241             pnew[i] = pdtxs[i];
242         for (int i = vt; i < sl - 1; i++)
243             pnew[i] = pdtxs[i + 1];
244         delete[]pdtxs;
245         pdtxs = pnew;
246         sl--;
247     }
248     else
249         cout << "Khong tim thay DTXS co ma so " << ms << endl;
250     return vt;
251 }
252 void TRUONGDH::GhiFile(string tenfile) {
253     string nd;
254     ofstream outputFile(tenfile);
255     if (!outputFile.fail())
256     {
257         cout << "Ban nhap noi dung file (ket thuc bang dau ~):\n";
258         while (1)
259         {
260             getline(cin, nd);
261             if (nd.back() != '~')
262                 outputFile << nd << endl;
263             else
264             {
265                 nd.back() = '\0';
266                 outputFile << nd;
267                 break;
268             }
269         }
270         outputFile.close();
271         cout << "Da ghi file thanh cong\n";
272     }
273     else
274         cout << "Loi tao file\n";
275 }

```



```

276 void TRUONGDH::DocFile(string tenfile) {
277     string nd;
278     ifstream inputFile(tenfile);
279     if (!inputFile.fail())
280     {
281         while (getline(inputFile, nd))
282             cout << nd << endl;
283         inputFile.close();
284         cout << "Da doc file thanh cong\n";
285     }
286     else
287         cout << "Loi doc file\n";
288 }
289 void TRUONGDH::GhiFileDTXS(string tenfile) {
290     ofstream outputFile(tenfile);
291     if (!outputFile.fail())
292     {
293         outputFile << s1 << endl;
294         string l;
295         for (int i = 0; i < s1; i++)
296         {
297             l = pdtxs[i]->GetLoai();
298             outputFile << pdtxs[i]->GetMaSo() << endl << pdtxs[i]->GetHoTen() << endl << pdtxs[i]->GetKhoa() << endl << l << endl;
299             if (l == "GV")
300                 outputFile << ((GV*)pdtxs[i])->GetSoBB() << endl;
301             else if (l == "SV")
302                 outputFile << ((SV*)pdtxs[i])->GetDTB() << endl << ((SV*)pdtxs[i])->GetDRL() << endl;
303             else
304                 outputFile << ((NV*)pdtxs[i])->GetSoSK() << endl;
305             outputFile << pdtxs[i]->TinhTienThuong() << endl;
306         }
307         outputFile.close();
308         cout << "Da ghi file thanh cong\n";
309     }
310     else
311         cout << "Loi ghi file\n";
312 }
313 void TRUONGDH::DocFileDTXS(string tenfile) {
314     ifstream inputFile(tenfile);
315     if (!inputFile.fail())
316     {
317         inputFile >> s1;
318         if (s1 > 0)
319         {
320             pdtxs = new DTXS * [s1];
321             string ms, ht, kh, l;
322             int bb, sk;
323             double tb, rl;
324             long tt;
325             for (int i = 0; i < s1; i++)
326                 //while (!inputFile.eof())
327                 {
328                     inputFile.ignore();
329                     getline(inputFile, ms);
330                     getline(inputFile, ht);
331                     getline(inputFile, kh);
332                     getline(inputFile, l);
333                     if (DoiChuoiInHoa(l) == "GV") {
334                         inputFile >> bb;
335                         pdtxs[i] = new GV(ms, ht, kh, bb);
336                     }
337                     else if (DoiChuoiInHoa(l) == "SV")
338                     {
339                         inputFile >> tb;
340                         inputFile >> rl;
341                         pdtxs[i] = new SV(ms, ht, kh, tb, rl);
342                     }
343                     else
344                     {
345                         inputFile >> sk;
346                         pdtxs[i] = new NV(ms, ht, kh, sk);
347                     }
348                     inputFile >> tt;
349                 }
350             cout << "Da doc file va luu d/s DTXS thanh cong\n";
351         }
352         else
353             cout << "So luong DTXS phai > 0\n";
354         inputFile.close();
355     }
356     else
357         cout << "Loi doc file\n";
358 }

```



```

1  #include "TRUONGDH.h"
2  void main() {
3      TRUONGDH tr;
4      int chon;
5      do {
6          cout << "==== CT QUAN LY DOI TUONG XUAT SAC CUA TRUONG =====\n";
7          cout << "1. Nhap d/s DTXS\n";
8          cout << "2. Xuat d/s DTXS\n";
9          cout << "3. Liet ke d/s GV co tu 5 bai bao khoa hoc tro len\n";
10         cout << "4. Dem SV co DRL toi da\n";
11         cout << "5. Tinh tong tien thuong cua NV co 3 SKKN tro len\n";
12         cout << "6. Tinh DTB cua SV co DRL duoi 50\n";
13         cout << "7. Kiem tra GV chua co bai bao khoa hoc\n";
14         cout << "8. Tim NV co SKKN nhieu nhat\n";
15         cout << "9. Sap xep d/s DTXS tang dan theo ma so\n";
16         cout << "10. Tim DTXS thuoc Khoa\n";
17         cout << "11. Cap nhat thong tin DTXS co ma so tuy y\n";
18         cout << "12. Them DTXS moi tai vi tri bat ky\n";
19         cout << "13. Xoa DTXS co ma so tuy y\n";
20         cout << "14. Xoa d/s DTXS\n";
21         cout << "15. Tao file\n";
22         cout << "16. Doc file\n";
23         cout << "17. Doc file va luu d/s DTXS\n";
24         cout << "18. Luu file d/s DTXS\n";
25         cout << "0. Thoat CT\n";
26         cout << "=====\n";
27         cout << "Ban chon: ";
28         cin >> chon;
29         switch (chon) {
30             case 0:
31                 if (tr.GetSL() > 0)
32                     tr.~TRUONGDH();
33                 cout << "Dang thoat CT...";
34                 break;
35             case 1:
36                 if (tr.GetSL() > 0)
37                     tr.~TRUONGDH();
38                 tr.Nhap();
39                 break;
40             case 2:
41                 if (tr.GetSL() > 0)
42                     tr.Xuat();
43                 else
44                     cout << "Chua nhap thong tin\n";
45                 break;
46             case 3:
47                 if (tr.GetSL() > 0)
48                     tr.LietKeGVCoTu5BaiBao();
49                 else
50                     cout << "Chua nhap thong tin\n";
51                 break;
52             case 4:
53                 if (tr.GetSL() > 0) {
54                     int dem = tr.DemSVCoDRLToiDa();
55                     if (dem > 0)
56                         cout << "Co " << dem << " SV co DRL toi da\n";
57                     else
58                         cout << "Khong co SV co DRL toi da\n";
59                 }
60                 else
61                     cout << "Chua nhap thong tin\n";
62                 break;

```

```

63     case 5:
64         if (tr.GetSL() > 0) {
65             int tongtien = tr.TinhTongTienThuongNVCo3SKKNTroLen();
66             if (tongtien > 0)
67                 cout << "Tong tien thuong cap cho NV co 3 SKKN tro len la " << tongtien << endl;
68             else
69                 cout << "Khong co NV co 3 SKKN tro len\n";
70         }
71         else
72             cout << "Chua nhap thong tin\n";
73         break;
74     case 6:
75         if (tr.GetSL() > 0) {
76             double dtb = tr.TinhDTBCuaSVCoDRLDuoai50();
77             if (dtb == 0)
78                 cout << "Khong co SV co DRL duoi 50\n";
79             else
80                 cout << "DTB cua SV co DRL duoi 50 la " << dtb << endl;
81         }
82         else
83             cout << "Chua nhap thong tin\n";
84         break;
85     case 7:
86         if (tr.GetSL() > 0) {
87             if (tr.KiemTraGVChuaCoBaiBao())
88                 cout << "Truong co GV chua co bai bao\n";
89             else
90                 cout << "Truong khong co GV chua co bai bao\n";
91         }
92         else
93             cout << "Chua nhap thong tin\n";
94         break;
95     case 8:
96         if (tr.GetSL() > 0)
97             tr.TimNVCoSKKNNhieuNhat();
98         else
99             cout << "Chua nhap thong tin\n";
100        break;
101    case 9:
102        if (tr.GetSL() > 0) {
103            tr.SapXepMaSoTangDan();
104            tr.Xuat();
105        }
106        else
107            cout << "Chua nhap thong tin\n";
108        break;
109    case 10:
110        if (tr.GetSL() > 0) {
111            string kh;
112            cin.ignore();
113            cout << "Nhap Khoa can tim: ";
114            getline(cin, kh);
115            tr.TimDTXSThuocKhoa(kh);
116        }
117        else
118            cout << "Chua nhap thong tin\n";
119        break;
120    case 11:
121        if (tr.GetSL() > 0) {
122            string ms;
123            cin.ignore();
124            cout << "Nhap ma so: ";
125            getline(cin, ms);
126            tr.CapNhatThongTinTheoMaSo(ms);
127        }
128        else
129            cout << "Chua nhap thong tin\n";
130        break;

```

```
131     case 12: {
132         DTXS* pdt;
133         int vt = 1;
134         pdt = tr.Nhap1();
135         if (tr.GetSL() > 0) {
136             do {
137                 cout << "Nhap vi tri muon them DTXS tu 1 den " << tr.GetSL() + 1 << ": ";
138                 cin >> vt;
139             } while (vt < 1 || vt > tr.GetSL() + 1);
140         }
141         tr.Them(pdt, vt);
142         cout << "D/s DTXS sau khi them DTXS moi tai vi tri thu " << vt << " la:\n";
143         tr.Xuat();
144         break;
145     }
146     case 13:
147         if (tr.GetSL() > 0) {
148             string ms;
149             cout << "Nhap ma so DTXS muon xoa: ";
150             cin >> ms;
151             int kq = tr.Xoa(ms);
152             if (kq != -1) {
153                 if (tr.GetSL() > 0) {
154                     cout << "D/s DTXS sau khi xoa DTXS co ma so " << ms << " la:\n";
155                     tr.Xuat();
156                 }
157                 else
158                     cout << "D/s DTXS rong\n";
159             }
160         }
161         else
162             cout << "Chua nhap thong tin\n";
163         break;
164     case 14:
165         if (tr.GetSL() > 0) {
166             tr.~TRUONGDH();
167             cout << "Da xoa toan bo d/s DTXS\n";
168         }
169         else
170             cout << "Chua nhap thong tin\n";
171         break;
172     case 15: {
173         string tenfile;
174         cin.ignore();
175         cout << "Nhap ten file: ";
176         getline(cin, tenfile);
177         tr.GhiFile(tenfile);
178         break;
179     }
180     case 16: {
181         string tenfile;
182         cin.ignore();
183         cout << "Nhap ten file: ";
184         getline(cin, tenfile);
185         tr.DocFile(tenfile);
186         break;
187     }
```

```
main.cpp X
TRUONGGDH
188 case 17: {
189     if (tr.GetSL() > 0)
190         tr.~TRUONGGDH();
191     string tenfile;
192     cin.ignore();
193     cout << "Nhap ten file: ";
194     getline(cin, tenfile);
195     tr.DocFileDTXS(tenfile);
196     break;
197 }
198 case 18:
199     if (tr.GetSL() > 0) {
200         string tenfile;
201         cin.ignore();
202         cout << "Nhap ten file: ";
203         getline(cin, tenfile);
204         tr.GhiFileDTXS(tenfile);
205     }
206     else
207         cout << "Chua nhap thong tin\n";
208     break;
209 default:
210     cout << "Ban chon sai. Moi chon lai.\n";
211     break;
212 }
213 } while (chon != 0);
214 }
```

Yêu cầu:

1. Kiểm tra không được trùng mã số khi nhập liệu.
2. Đọc – ghi file nhị phân để mã hóa dữ liệu.