Họ và tên: Trần Đăng Hiếu

Lớp: 64CNTT3

MSV: 2251061779

***Bài 2: Quản lý đăng ký học theo tín chỉ***

Khai báo lớp  ***Sinh viên (mã sinh viên, Họ tên, Địa chỉ, Số ĐT, Lớp),*** mã sinh viên là một số nguyên có 5 chữ số, tự động tăng.

Khai báo lớp ***Môn học*** gồm các thuộc tính ***(mã môn học, Tên môn, Tổng số tiết, Loại môn học)*** – trong đó Loại môn học có thể là: Đại cương, Cơ sở ngành, Chuyên ngành bắt buộc, Chuyên ngành tự chọn.  Mã môn học là một số nguyên có 3 chữ số, tự động tăng.

Khai báo lớp ***Bảng Đăng ký*** có quan hệ kết hợp (association) với lớp ***Sinh viên*** và lớp ***Môn học*** trong đó một sinh viên sẽ được đăng ký không quá 8 môn học một học kỳ. Ghi rõ thời gian đăng ký của sinh viên.

Viết chương trình trong ngôn ngữ C++ thực hiện các yêu cầu sau:

1. Nhập danh sách môn học mới. In ra danh sách các môn học đã có.
2. Nhập danh sách sinh viên.  In ra danh sách sinh viên đã có.
3. Lập Bảng Đăng ký cho từng sinh viên bằng cách nhập các môn học cho sinh viên đó và thời gian đăng ký (lấy từ thời gian hệ thống), và in ra danh sách ra màn hình *(chú ý: cùng 1 sinh viên với 1 môn học thì không thể xuất hiện 2 lần trong bảng này).*

Source code

File SinhVien.cpp khai báo lớp SinhVien

1. #include <iostream>
2. #include <iomanip>
3. #include <string>
5. **using** **namespace** std;
7. // khai bao lop sinh vien
8. **class** SinhVien{
9. **private**:
10. **static** **int** MaSVID;
11. **int** MaSV;
12. string HoTen;
13. string DiaChi;
14. string SoDT;
15. **public**:
16. SinhVien(){
17. **this**->MaSV = MaSVID++;
18. };
19. SinhVien(string HoTen, string DiaChi, string SoDT) : HoTen(HoTen), DiaChi(DiaChi), SoDT(SoDT){
20. **this**->MaSV = MaSVID++;
21. };
22. string getHoTen(){
23. **return** **this**->HoTen;
24. }
25. **void** Nhap();
26. **void** Xuat();
27. };
29. //Nhap sinh vien
30. **void** SinhVien::Nhap(){
31. cout << endl << "Nhap ho ten: ";
32. getline(cin, HoTen);
33. cout << "Nhap dia chi: ";
34. getline(cin, DiaChi);
35. cout << "Nhap so dien thoai: ";
36. getline(cin, SoDT);
37. }
38. //Xuat sinh vien
39. **void** SinhVien::Xuat(){
40. cout << left << setw(15) << MaSV << setw(40) << HoTen << setw(40) << DiaChi << setw(15) << SoDT << endl;
41. }
43. **int** SinhVien::MaSVID = 10000;

File MonHoc.cpp khai báo lớp MonHoc

1. #include <iostream>
2. #include <iomanip>
3. #include <string>
5. **using** **namespace** std;
7. // khai bao lop mon hoc
8. **class** MonHoc{
9. **private**:
10. **static** **int** MaMHID;
11. **int** MaMonHoc;
12. string TenMonHoc;
13. **int** TongSoTiet;
14. string LoaiMonHoc;
15. **public**:
16. MonHoc(){
17. **this**->MaMonHoc = MaMHID++;
18. };
19. MonHoc(string TenMonHoc, **int** TongSoTiet, string LoaiMonHoc) : TenMonHoc(TenMonHoc), TongSoTiet(TongSoTiet), LoaiMonHoc(LoaiMonHoc){
20. **this**->MaMonHoc = MaMHID++;
21. };
22. string getTenMonHoc(){
23. **return** **this**->TenMonHoc;
24. }
25. **void** Nhap();
26. **void** Xuat();
27. };
29. //Nhap mon hoc
30. **void** MonHoc::Nhap(){
31. cout << endl << "Nhap ten mon hoc: ";
32. getline(cin, TenMonHoc);
33. cout << "Nhap tong so tiet: ";
34. cin >> TongSoTiet;
35. cin.ignore();
36. cout << "Nhap loai mon hoc: ";
37. getline(cin, LoaiMonHoc);
38. }
40. //Xuat mon hoc
41. **void** MonHoc::Xuat(){
42. cout << left << setw(15) << MaMonHoc << setw(30) << TenMonHoc << setw(15) << TongSoTiet << setw(30) << LoaiMonHoc << endl;
43. }
45. **int** MonHoc::MaMHID = 100;

File BangDangKy.cpp khai báo lớp BangDangKy có quan hệ kết hợp với 2 lớp SinhVien và MonHoc.

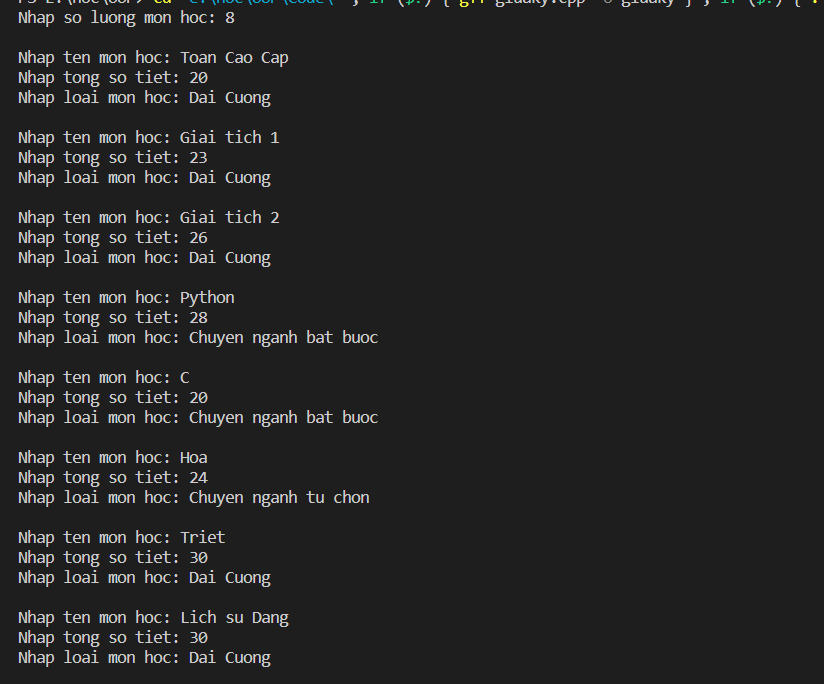
1. #include <iostream>
2. #include <iomanip>
3. #include <string>
4. #include <ctime>
5. #include "SinhVien.cpp"
6. #include "MonHoc.cpp"
8. **using** **namespace** std;
9. // khai bao lop bang dang ky
10. **class** BangDangKy{
11. **private**:
12. MonHoc monHoc;
13. SinhVien sinhVien;
14. **time\_t** thoiGianDangKy;
15. **public**:
16. BangDangKy(){};
17. BangDangKy(MonHoc monHoc, SinhVien sinhVien, **time\_t** thoiGianDangKy) : monHoc(monHoc), sinhVien(sinhVien), thoiGianDangKy(thoiGianDangKy){};
18. **void** Xuat();
19. };
21. //Xuat bang dang ky
22. **void** BangDangKy::Xuat(){
23. cout << left << setw(40) << sinhVien.getHoTen() << setw(30) << monHoc.getTenMonHoc() << setw(30) << ctime(&thoiGianDangKy) << endl;
24. }

File main.cpp xử lý yêu cầu đề bài

1. #include <iostream>
2. #include <vector>
3. #include <iomanip>
4. #include <string>
5. #include <algorithm>
6. #include <ctime>
7. #include "BangDangKy.cpp"
9. **using** **namespace** std;
11. // main
12. **int** main(){
13. //Khai bao
14. vector<SinhVien\*> danhSachSinhVien;
15. vector<MonHoc\*> danhSachMonHoc;
16. vector<BangDangKy\*> danhSachBangDangKy;
18. //bai 1: nhap danh sach mon hoc va in ra danh sach mon hoc
19. **int** slMonHoc;
20. cout << "Nhap so luong mon hoc: ";
21. cin >> slMonHoc;
22. cin.ignore();
23. //nhap danh sach mon hoc
24. **for**(**int** i=0; i<slMonHoc; i++){
25. MonHoc \*monHoc = **new** MonHoc();
26. monHoc->Nhap();
27. danhSachMonHoc.push\_back(monHoc);
28. }
29. //in ra danh sach mon hoc
30. cout << endl << right << setw(50) << "Danh sach mon hoc vua nhap: " << endl << endl;
31. cout << left << setw(15) << "Ma mon hoc" << setw(30) << "Ten mon hoc" << setw(15) << "So tiet" << setw(30) << "Loai mon hoc" << endl;
32. cout << setfill('-') << setw(85) << "-" << setfill(' ')<< endl;
33. **for**(**int** i=0; i<slMonHoc; i++){
34. danhSachMonHoc[i]->Xuat();
35. }
37. //bai 2: nhap danh sach sinh vien va in ra danh sach sinh vien
38. **int** slSinhVien;
39. cout << endl << "Nhap so luong sinh vien: ";
40. cin >> slSinhVien;
41. cin.ignore();
42. //nhap danh sach sinh vien
43. **for**(**int** i=0; i<slSinhVien; i++){
44. SinhVien \*sinhVien = **new** SinhVien();
45. sinhVien->Nhap();
46. danhSachSinhVien.push\_back(sinhVien);
47. }
48. //in ra danh sach sinh vien
49. cout << endl << right << setw(50) << "Danh sach sinh vien vua nhap: " << endl << endl;
50. cout << left << setw(15) << "Ma sinh vien" << setw(40) << "Ho ten" << setw(40) << "Dia chi" << setw(15) << "So dien thoai" << endl;
51. cout << setfill('-') << setw(110) << "-" << setfill(' ')<< endl;
52. **for**(**int** i=0; i<slSinhVien; i++){
53. danhSachSinhVien[i]->Xuat();
54. }
55. //bai 3:Lập Bảng Đăng ký cho từng sinh viên
56. //Dang ky mon hoc
57. **for**(**int** i=0; i<danhSachSinhVien.size(); i++){
58. **for**(**int** j=0; j<danhSachMonHoc.size(); j++){
59. cout << endl << "Ban co muon dang ky mon hoc " << danhSachMonHoc[j]->getTenMonHoc() << " cho sinh vien "<< danhSachSinhVien[i]->getHoTen() << " khong? (Y/N): ";
60. string check;
61. getline(cin, check);
62. **if**(check=="Y" || check=="y"){
63. **time\_t** thoigianDangKy = time(nullptr);
64. BangDangKy \*bangDangKy = **new** BangDangKy(\*danhSachMonHoc[j], \*danhSachSinhVien[i], thoigianDangKy);
65. danhSachBangDangKy.push\_back(bangDangKy);
66. }
67. }
68. }
69. //in ra bang dang ky
70. cout << endl << right << setw(50) << "Danh sach bang dang ky: " << endl << endl;
71. cout << left << setw(40) << "Ho ten" << setw(30) << "Ten mon hoc" << setw(30) << "Thoi gian dang ky" << endl;
72. cout << setfill('-') << setw(100) << "-" << setfill(' ')<< endl;
73. **for**(**int** i=0; i<danhSachBangDangKy.size(); i++){
74. danhSachBangDangKy[i]->Xuat();
75. }
76. **return** 0;
77. }

Result

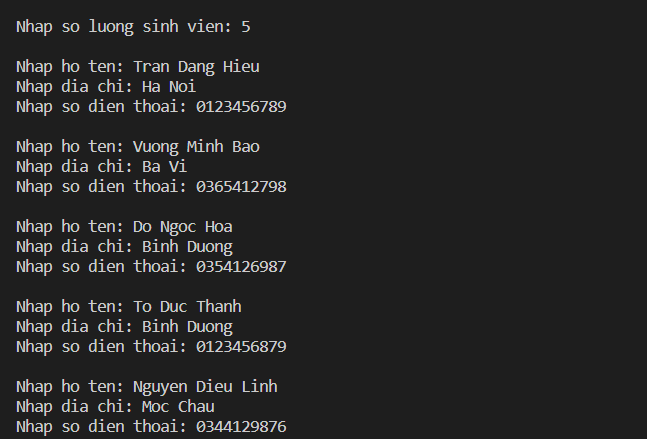
Bài 1:



A screenshot of a computer

Description automatically generated

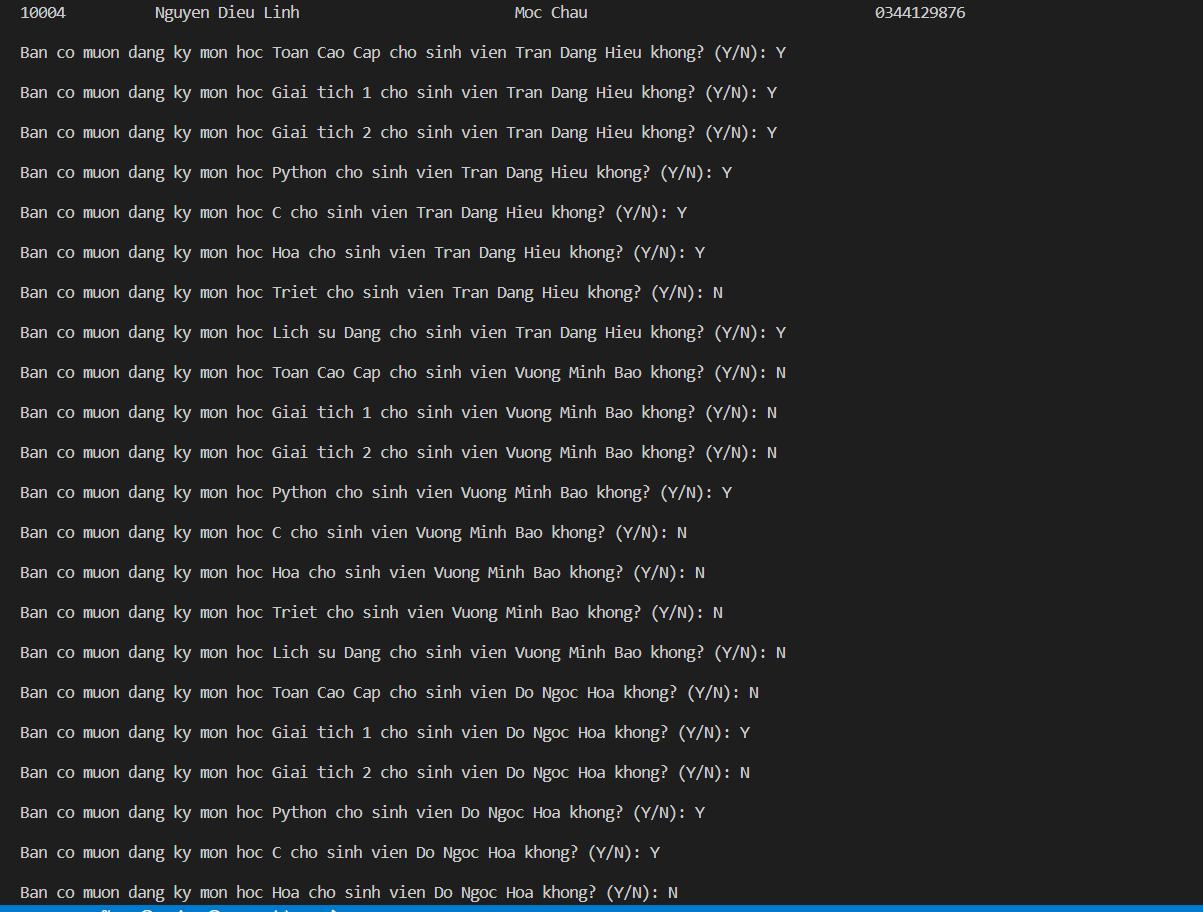
Bài 2:



A black screen with white text

Description automatically generated

Bài 3:



A black screen with white text

Description automatically generated

A screenshot of a computer

Description automatically generated