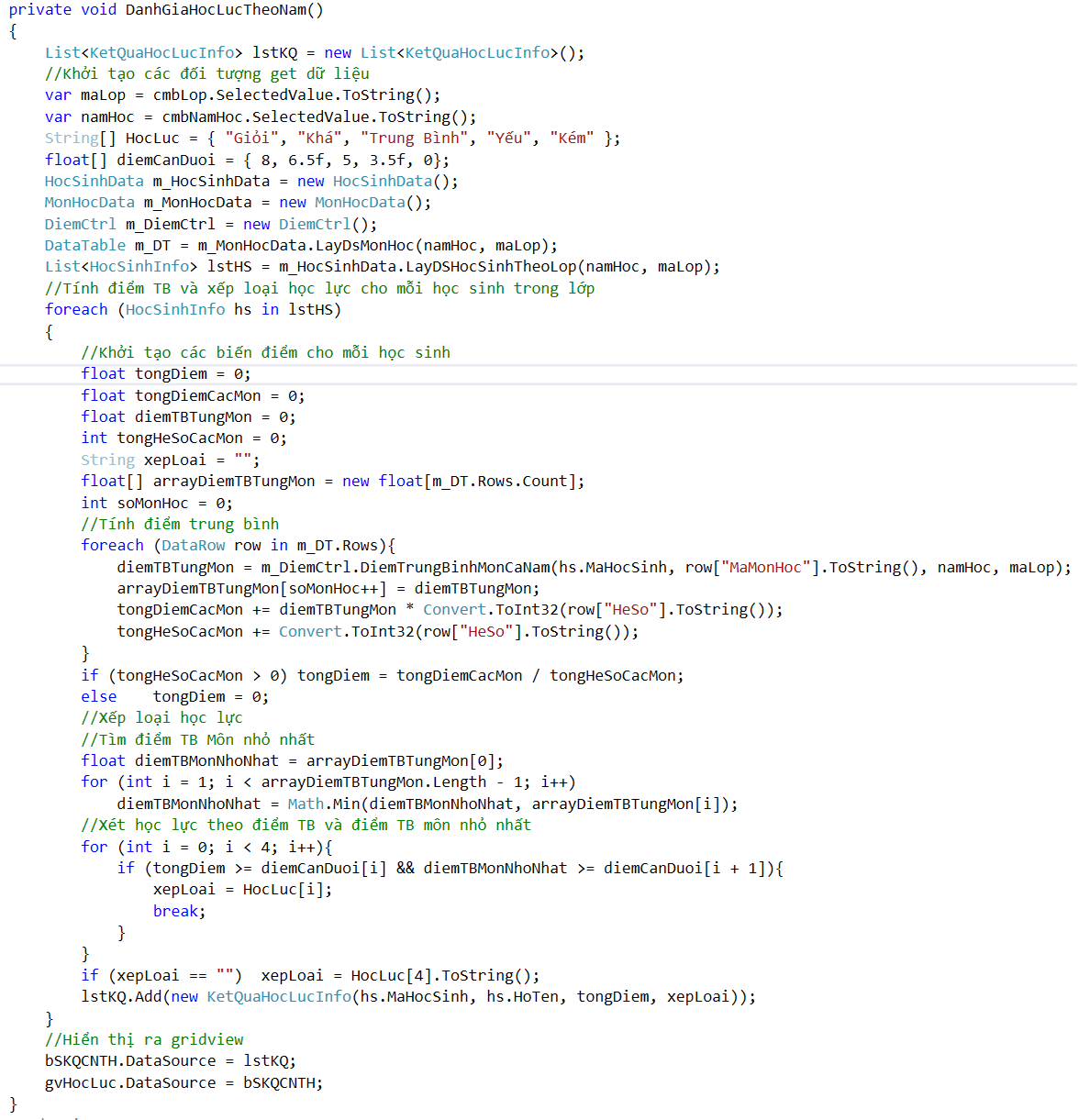
1. **Test chức năng Đánh Giá Học Lực Theo Năm**
2. **Giao diện module chức năng đánh giá học lực theo năm**

****

1. **Test module chức năng đánh giá học lực theo năm**

Gọi mệnh đề thể hiện tính chất dữ liệu vào của chương trình {A} và mệnh đề  
thể hiện tính chất dữ liệu ra cần có {B}, ta có:

**{A: maLop ∈ string, namHoc ∈ string}** và **{B: lstKQ ∈ List<KetQuaHocLucInfo>}**

**Trong đó:**

* **maLop** là mã lớp được lấy từ combobox theo kiểu string
* **namHoc** là năm học được lấy từ combobox theo kiểu string
* **lstKQ**: là danh sách kết quả học lực của cả lớp theo năm theo kiểu KetQuaHocLucInfo

Chứng minh **{A}P{B}**

**B1: Dự Đoán**

* Dự đoán {**C1**} là bất biến của đoạn trình **Q1**

{**C1:**  lstHS ∈ List<HocSinhInfo>, HS0..HSn ∈ HocSinhInfo, i, n ∈ int; n = lstHS.count, HSi= lstHS[i]; TinhHocLucVaXepLoai(HS0, HSn) = DaTinhHocLucVaXepLoai (HS0, HSi-1) U TinhHocLucVaXepLoai(HSi, HSn)}

* Dự đoán {**C2**} là bất biến của đoạn trình **Q2**

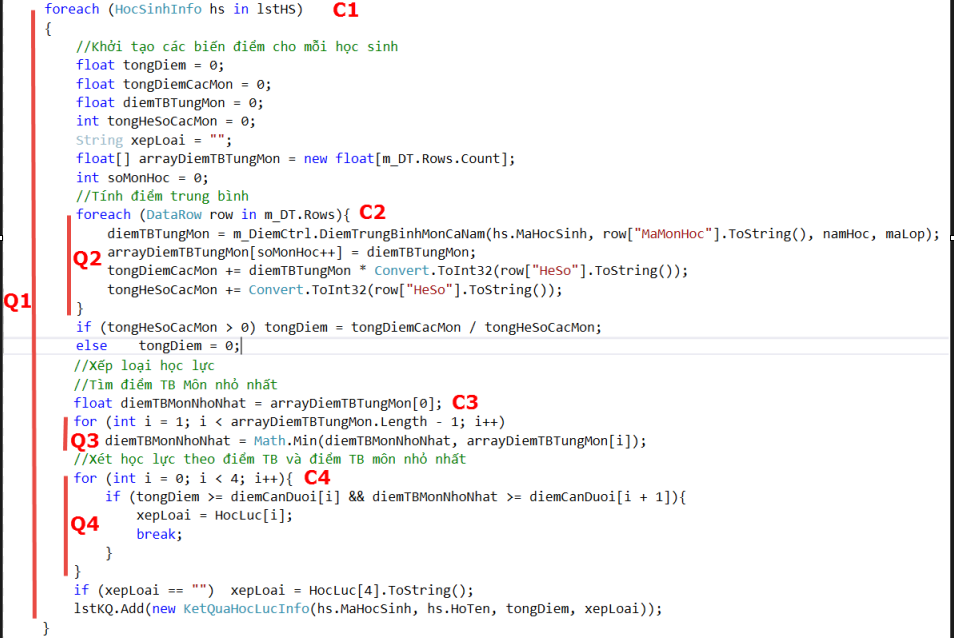
{**C2:**  MH0..MHn ∈ List<MonHoc>; TinhTongDiem(MH0, MHn) = DaTinhTongDiem (MH0, MHi-1) U TinhTongDiem (MHi, MHn)}

* Dự đoán {**C3**} là bất biến của đoạn trình **Q3**

{**C3:**  DiemTBMon0..DiemTBMonn ∈ Array<float>; TimDiemTBNhoNhat(DiemTBMon0 ,DiemTBMonn) = DaTimDiemTBNhoNhat (DiemTBMon0, DiemTBMoni-1) U TimDiemTBNhoNhat (DiemTBMoni, DiemTBMonn)}

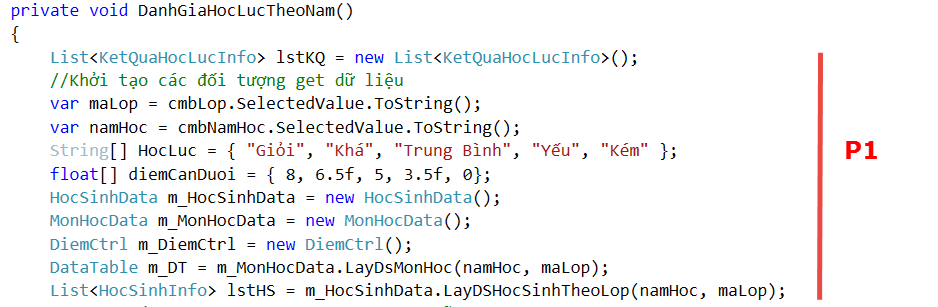
* Dự đoán {**C4}** là bất biến của đoạn trình **Q4**

{**C4:**  i = [0..4] ∈ int; XepHocLuc(0, n) = DaXepHocLuc (0, i-1) U XepHocLuc (i, n)}



**B1: Chứng minh**

* {**A**} **P1**{**C­1**}, với đoạn trình **P1** như sau:



Ta có:  
{**A**} List<KetQuaHocLucInfo> lstKQ = new List<KetQuaHocLucInfo>();{**A­1**}

{**A­1**}String[] HocLuc = { "Giỏi", "Khá", "Trung Bình", "Yếu", "Kém" };{**A2**}

{**A­2**}float[] diemCanDuoi = { 8, 6.5f, 5, 3.5f, 0};{**A­3**}

{**A­3**}HocSinhData m\_HocSinhData = new HocSinhData();{**A4**}

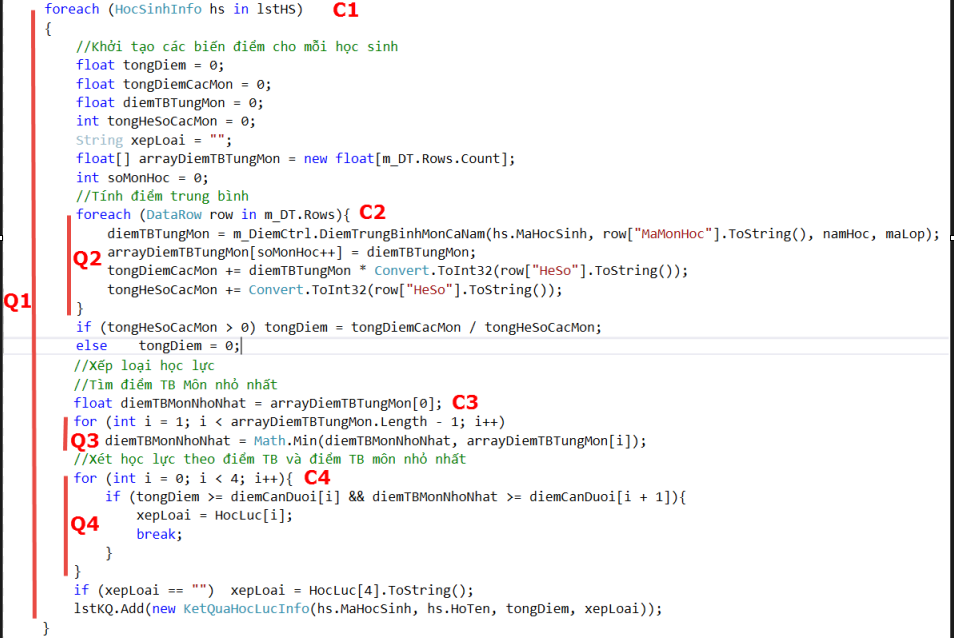
{**A­4**}MonHocData m\_MonHocData = new MonHocData();{**A5**}

{**A­5**}DiemCtrl m\_DiemCtrl = new DiemCtrl();{**A6**}

{**A­6**}DataTable m\_DT = m\_MonHocData.LayDsMonHoc(namHoc, maLop); {**A­7**}

{**A­7**}List<HocSinhInfo> lstHS = m\_HocSinhData.LayDSHocSinhTheoLop(namHoc, maLop); {**C1**}

* Chứng minh {**C**1} là bất biến của đoạn trình **Q1**, với đoạn trình **Q1** như sau:



Ta cần có: {**C1, E1**: lstHS ∈ List<HocSinhInfo>, HS0..HSn ∈ HocSinhInfo, i, n ∈ int; n = lstHS.count, HSi= lstHS[i]; TinhHocLucVaXepLoai(HS0, HSn) = DaTinhHocLucVaXepLoai (HS0, HSi-1) U TinhHocLucVaXepLoai(HSi, HSn); i < n;} **Q1 {C1}.**

Ta có:

{C1}float tongDiem = 0; {C11}

{C11}float tongDiemCacMon = 0; {C12}

{C12}float diemTBTungMon = 0; {C13}

{C13}int tongHeSoCacMon = 0; {C14}

{C14}String xepLoai = "";{C15}

{C15}float[] arrayDiemTBTungMon = new float[m\_DT.Rows.Count]; {C16}

{C16}int soMonHoc = 0; {C2}

* Chứng minh {**C**2} là bất biến của đoạn trình **Q2**.

Ta cần có: {**C2, E2**: lstMH ∈ List<MonHoc>, MH0..MHn ∈ MonHoc, i, n ∈ int; n = lstMH.count, MHi= lstMH[i]; TinhTongDiem (MH0, MHn) = DaTinhTongDiem(MH0, MHi-1) U TinhTongDiem(MHi, MHn); TinhTongHeSo(MH0, MHn) = DaTinhTongHeSo(MH0, MHi-1) U TinhTongHeSo(MHi, MHn); i < n;} **Q2 {C2}**

Theo tính chất của phép gán, ta có:

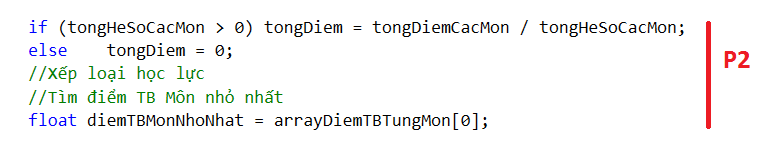
{**C21**} ≡ {C2[TinhTongHeSo|TinhTongHeSo +HeSo]: lstMH ∈ List<MonHoc>, MH0..MHn ∈ MonHoc, i, n ∈ int; n = lstMH.count, MHi= lstMH[i]; TinhTongDiem (MH0, MHn) = DaTinhTongDiem(MH0, MHi-1) U TinhTongDiem(MHi, MHn); **TinhTongHeSo(MH0, MHn) = DaTinhTongHeSo(MH0, MHi-1) U (TinhTongHeSo(MHi, MHn) + HeSo(MHi, MHn));**}

{**C22**} ≡ {C21[TinhTongDiem | TinhTongDiem + DiemTBTungMon \* HeSo]: lstMH ∈ List<MonHoc>, MH0..MHn ∈ MonHoc, i, n ∈ int; n = lstMH.count, MHi= lstMH[i]; **TinhTongDiem (MH0, MHn) = DaTinhTongDiem(MH0, MHi-1) U TinhTongDiem(MHi, MHn) + DiemTBTungMon(MHi, MHn) \* HeSo(MHi, MHn));** TinhTongHeSo(MH0, MHn) = DaTinhTongHeSo(MH0, MHi-1) U (TinhTongHeSo(MHi, MHn) + HeSo(MHi, MHn));}

**{C2, E2}Q2{C2}** nên kết thúc Q2 ta có mệnh đề **{C2, !E2}:**

**{C2, !E2:** lstMH ∈ List<MonHoc>, MH0..MHn ∈ MonHoc, i, n ∈ int; n = lstMH.count, MHi= lstMH[i]; **TinhTongDiem (MH0, MHn) = DaTinhTongDiem(MH0, MHi-1) U TinhTongDiem(MHi, MHn); TinhTongHeSo(MH0, MHn) = DaTinhTongHeSo(MH0, MHi-1) U TinhTongHeSo(MHi, MHn);** i >=n;**}**

* **Chứng minh {C2, !E2}P2{C3}** với P2 là đoạn trình như sau:

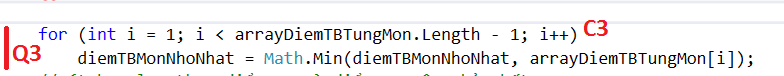
****

**{C2, !E2}** if (tongHeSoCacMon > 0) tongDiem = tongDiemCacMon / tongHeSoCacMon;

else tongDiem = 0**;{C21}**

Dễ dàng ta thấy:

* **{C2, !E2, E3: }** tongDiem = tongDiemCacMon / tongHeSoCacMon;{C21}
* **{C2, !E2, !E3: }** tongDiem = 0;{C21}
* **{C21}** diemTBMonNhoNhat = arrayDiemTBTungMon[0];{C3}
* Chứng minh {**C**3} là bất biến **Q3**, với Q3 là đoạn trình như sau:

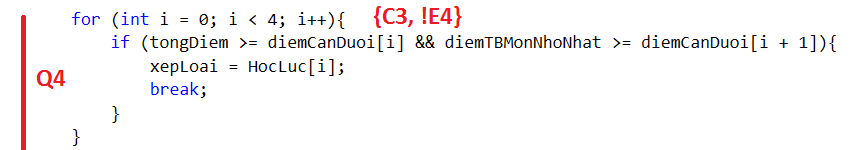
****

Ta có: {**C3, E4:** arrayDiemTBTungMon ∈ Array<float>, DiemTBMon0..DiemTBMonn ∈ float; i, n ∈ int; i = 1; n = arrayDiemTBTungMon.Length - 1; TimDiemTBNhoNhat(DiemTBMon0 ,DiemTBMonn) = DaTimDiemTBNhoNhat (DiemTBMon0, DiemTBMoni-1) U TimDiemTBNhoNhat (DiemTBMoni, DiemTBMonn); i < n - 1} **Q3 {C3}**

Vậy {C3} là bất biến của Q3. Nên kết thúc Q3, ta có mệnh đề **{C3, !E4}:**

{**C3, !E4:** arrayDiemTBTungMon ∈ Array<float>, DiemTBMon0..DiemTBMonn ∈ float; i, n ∈ int; i = 1; n = arrayDiemTBTungMon.Length - 1; TimDiemTBNhoNhat(DiemTBMon0 ,DiemTBMonn) = DaTimDiemTBNhoNhat (DiemTBMon0, DiemTBMoni-1) U TimDiemTBNhoNhat (DiemTBMoni, DiemTBMonn); i >= n - 1} **Q3 {C3}**

* Chứng minh **{C3, !E4}** là bất biến **Q4**, đặt **{C3, !E4} là {C4}** với Q4 là đoạn trình như sau:

****

Ta cần có: {**C4 , E5:** i = [0..4] ∈ int; i = 0; XepHocLuc(0, n) = DaXepHocLuc (0, i-1) U XepHocLuc (i, n); i < 4} **Q4 {C4}**

**Ta cần chứng tỏ:**

{**C4, E5**} if (tongDiem >= diemCanDuoi[i] && diemTBMonNhoNhat >= diemCanDuoi[i + 1]){

xepLoai = HocLuc[i]; break;} **{C4}**

Dễ dàng ta có

{**C4**, E5, E6: tongDiem >= diemCanDuoi[i] && diemTBMonNhoNhat >= diemCanDuoi[i + 1]} xepLoai = HocLuc[i]; break;{**C4**, **E5**}

{**C4**, E5, !E6: tongDiem < diemCanDuoi[i] || diemTBMonNhoNhat < diemCanDuoi[i + 1]} {**C4**, **E5**}

**Vậy {C4, E5} là bất biến của Q4. Kết thúc Q4, ta có mệnh đề {C4, !E5}**

* **Chứng minh {C4, !E5}P3{C41}** với P3 là đoạn trình như sau:

****

**Ta cần chứng tỏ:**

{**C4, !E5:** i = [0..4] ∈ int; i = 0; XepHocLuc(0, n) = DaXepHocLuc (0, i-1) U XepHocLuc (i, n); i >= 4**}** if (xepLoai == "") xepLoai = HocLuc[4].ToString();**{C41}**

**+ Dễ dàng ta có**

{**C4, !E5**, E7: xepLoai == ""} xepLoai = HocLuc[4].ToString();{**C4**, !E5}

{**C4**, !E5, !E7: xepLoai != ""}{**C4**, **!E5**}

* {**C4 , !E5}**lstKQ.Add(new KetQuaHocLucInfo(hs.MaHocSinh, hs.HoTen, tongDiem, xepLoai));**{C1}**

**Vậy** **{C1}** là bất biến của **Q1**. Nên kết thúc Q1, ta có mệnh đề **{C1, !E1}.**

**Với {C1, !E1:** lstHS ∈ List<HocSinhInfo>, HS0..HSn ∈ HocSinhInfo, i, n ∈ int; n = lstHS.count, HSi= lstHS[i]; TinhHocLucVaXepLoai(HS0, HSn) = DaTinhHocLucVaXepLoai (HS0, HSi-1) U TinhHocLucVaXepLoai(HSi, HSn); i >= n;}  
+ Dễ dàng chứng tỏ: **{C1, !E1}****{B}**

Vậy ta có **{A}P{B},** hay chương trình trên là đúng