```cpp
#include <iostream>
#include <string>
#include <fstream>
#include <iomanip>
using namespace std;
const int SIZE = 20;

void AddNew(fstream& data);
void ViewStatus(fstream& data, int);
void read(fstream& data, int&);
void payFee(fstream& data, int);
int search(int[]);
void deleteRecord(fstream& data, int);

struct info {
    string name[SIZE];
    int matno[SIZE];
    string course[SIZE];
    double fee[SIZE];
    double notPaid[SIZE];
    double paid[SIZE];
    int bilangan;
};
info student;  //declare struct

int main()
{
    int choice = 0;
    fstream data;


    do
    {

        system("cls");
        cout << "Welcome To The Maple High School Finance Website" << endl;
        cout << "_____________________________________________________" << endl;
        cout << "_____________________________________________________" << endl;

        cout << "PLEASE SELECT THE OPTION BELOW" << endl;
        cout << "1. Registration" << endl;
        cout << "2. View Financial Statement" << endl;
        cout << "3. Make Payment " << endl;
        cout << "4. Delete record" << endl;
        cout << "5. Exit " << endl;

        cout << endl << "Input your Choice : " << endl;
        cin >> choice;

```

```
50          for (choice; choice < 1 || choice > 5;)
51          {
52              cout << "Invalid.Please enter a valid choice : ";
53              cin >> choice;
54          }
55          switch (choice)
56          {
57          case 1: AddNew(data); break;
58          case 2: read(data, student.bilangan);
59              ViewStatus(data, student.bilangan); break;
60          case 3: read(data, student.bilangan);
61              payFee(data, student.bilangan); break;
62          case 4: read(data, student.bilangan);
63              deleteRecord(data, student.bilangan); break;
64          default: cout << "Thank You :)" << endl;
65          }
66
67      } while (choice != 5);
68
69
70      return 0;
71
72  }
73
74
75  void AddNew(fstream& data) //addnewname
76  {
77      system("cls");
78      cout << "---Add new student data---" << endl << endl;
79      data.open("Database.txt", ios::out | ios::app);
80      string name[SIZE];
81      int matricno[SIZE], i = 0, select, j;
82      double fee[SIZE];
83      double paid[SIZE];
84      double notPaid[SIZE];
85      string course[SIZE];
86      bool cont = false;
87      char respond;
88
89
90      do
91      {
92          cin.ignore();
93          cout << "Please enter your name : ";
94          getline(cin, name[i]);
95          cout << endl;
96
97          cout << "Please enter your matric number : ";
98          cin >> matricno[i];
```

```
 99          cout << endl;
100
101          cout << " Please select your course " << endl;
102          cout << "  1. Engine (Fee = RM 2000) " << endl;
103          cout << "  2. Law (Fee = RM 2500) " << endl;
104          cout << "  3. Econs (Fee = RM 1500) " << endl;
105          cout << "  4. ICT (Fee = RM 1000) " << endl;
106          cout << "  5. Medic (Fee = RM 3000) " << endl;
107          cout << "  6. Education (Fee = RM 2600) " << endl;
108
109          cin >> select;
110          while (select < 1 || select > 6)
111          {
112              cout << "The course that you entered does not exist! Please re-
                   enter : " << endl;
113              cout << " 1. Engine" << endl;
114              cout << " 2. Law" << endl;
115              cout << " 3. Econs" << endl;
116              cout << " 4. Ict" << endl;
117              cout << " 5. Medic" << endl;
118              cout << " 6. Education" << endl;
119              cin >> select;
120          }
121
122          if (select == 1) {
123
124              fee[i] = 2000;
125              course[i] = "Engin";
126          }
127          else if (select == 2) {
128
129              fee[i] = 2500;
130              course[i] = "Law";
131          }
132          else if (select == 3) {
133
134              fee[i] = 2500;
135              course[i] = "Econs";
136          }
137          else if (select == 4){
138
139              fee[i] = 1000;
140              course[i] = "ICT";
141          }
142          else if (select == 5){
143
144              fee[i] = 3000;
145              course[i] = "Medic";
146          }
```

```
147         else{
148
149             fee[i] = 2600;
150             course[i] = "Edu";
151         }
152
153
154         paid[i] = 0;
155         notPaid[i] = fee[i] - paid[i];
156         i++;
157
158         cout << "Do you wish to add another student ? (Y/N)" << endl;
159         cin >> respond;
160
161         switch (respond)
162         {
163         case 'y':
164         case 'Y': cont = true; break;
165         case 'n':
166         case 'N' or 'n': cont = false; break;
167         }
168
169     } while (cont && (i < SIZE));
170
171     j = i;
172
173
174
175     for (i = 0; i < j; i++)
176     {
177         data << name[i] << '$' << matricno[i] << " " << course[i] << " " << fee
              [i] << " " << paid[i] << " " << notPaid[i] << endl;
178     }
179     cout << "Data Successfully Recorded " << endl;
180     system("pause");
181     data.close();
182
183 }
184
185 void read(fstream& data, int& totalist) {  //read from file
186     data.open("Database.txt", ios::in);
187     system("cls");
188     string name[SIZE];
189     int matno[SIZE];
190     double fee[SIZE];
191     string course[SIZE];
192
193
194     int i = 0;
```

```
195
196     getline(data, student.name[i], '$');
197     data >> student.matno[i] >> student.course[i] >> student.fee[i] >>
          student.paid[i] >> student.notPaid[i];
198     while (data) {
199         i++;
200         data.ignore();
201         getline(data, student.name[i], '$');
202         data >> student.matno[i] >> student.course[i] >> student.fee[i] >>
              student.paid[i] >> student.notPaid[i];
203     }
204     totalist = i;
205     data.close();
206 }
207
208 void ViewStatus(fstream& data, int i) { //financial statement
209
210     cout << "---Financial status for all students---" << endl << endl;
211     cout << setprecision(2) << showpoint << fixed;
212     cout << setw(15) << "Name" << setw(23) << "Matric Number" << setw(13) <<
          "Course" << setw(15) << "Fee" << setw(19) << "Amount Paid" << setw(19) <<
           "\tAmount Not Paid" << endl;
213     for (int j = 0; j < i; j++) { //print
214         cout << endl;
215         cout << (j + 1) << ". ";
216         cout << setw(23) << left << student.name[j] << setw(15) << left <<
              student.matno[j] << setw(13) << left << student.course[j] << "RM " <<
               setw(14) << left << student.fee[j];
217         cout << "RM " << setw(18) << left << student.paid[j] << "RM " << setw
              (13) << left << student.notPaid[j];
218     }
219     cout << endl;
220     cout << "\n--------END OF THE LIST--------" << endl << endl;
221     system("pause");
222 }
223
224
225 void payFee(fstream& data, int j) {
226     data.open("Database.txt", ios::out); //rewrite the file
227     double payment;
228     cout << "----Make Payment----\n" << endl;
229     int pos = search(student.matno);
230
231     if (pos > -1) {
232         cout << "Initiating payment for student...\n" << endl;
233         cout << "\tName: " << student.name[pos] << endl;
234         cout << "\tCourse: " << student.course[pos] << endl << endl;
235         cout << setprecision(2) << showpoint << fixed;
236         cout << "Total fee:       RM " << student.fee[pos] << endl;
```

```
237         cout << "Outstanding fee: RM " << student.notPaid[pos] << endl;
238         cout << "Amount to pay?:  RM ";
239         cin >> payment;
240
241         while (payment > student.notPaid[pos]) {
242             cout << "Error!" << endl;
243             cout << "The amount you entered is higher than the amount you need
                  to pay." << endl;
244             cout << "Please re-enter amount: ";
245             cin >> payment;
246         }
247         char respond;
248         cout << endl <<"You have entered RM " << payment << "\tConfirm? (Y/N)";
249         cin >> respond;
250         switch (respond) {
251         case 'Y':
252         case 'y':
253             student.paid[pos] = student.paid[pos] + payment;
254             student.notPaid[pos] = student.notPaid[pos] - payment;
255             cout << "\nPayment successfully recorded" << endl << endl; break;
256         case 'n':
257         case 'N': cout << "Payment cancelled." << endl; break;
258         }
259
260     }
261     else
262         cout << "Sorry. Matric number entered not matching with any students."
              << endl;
263
264
265     for (int i = 0; i < j; i++)
266     {
267         data << student.name[i] << '$' << student.matno[i] << " " <<
              student.course[i] << " " << student.fee[i] << " " << student.paid[i]
              << " " << student.notPaid[i] << endl;
268     }
269
270     system("pause");
271     data.close();
272 }
273
274 int search(int arrayMatricNum[]) { //return position of array
275     int index = 0;
276     int position = -1;
277     bool found = false;
278     int value;
279
280     cout << "Please enter matric number of the student: ";
281     cin >> value;
```

```
282     while (index < SIZE && !found) {
283         if (arrayMatricNum[index] == value) {
284             found = true;
285             position = index;
286             cout << "Record found!\n" << endl;
287         }
288         index++;
289     }
290     return position;
291 }
292 
293 void deleteRecord(fstream& data, int z) {
294     cout << "---Delete student record from the system---" << endl << endl;
295     cout << "Note: This is aplicable to student that had complete \n      their ↵
        fee payment" << endl << endl;
296     char respond;
297     int pos = search(student.matno);
298 
299     if (pos > -1) {
300         cout << endl;
301         cout << "\tName: " << student.name[pos] << endl;
302         cout << "\tCourse: " << student.course[pos] << endl << endl;
303 
304         if (student.notPaid[pos] == 0) {
305             cout << "This student has pay full fees" << endl;
306             cout << "Confirm to delete this student record? (Y/N): ";
307             cin >> respond;
308             switch (respond) {
309             case 'Y': case 'y': {
310                 data.open("Database.txt", ios::out); //rewrite the file
311                 for (int i = 0; i < z; i++)
312                 {
313                     if (i != pos) {
314                         data << student.name[i] << '$' << student.matno[i] << " ↵
                         " << student.course[i] << " " << student.fee[i] << " " << ↵
                        student.paid[i] << " " << student.notPaid[i] << endl;
315                     }
316                     else
317                         continue;   //<- debugging
318                 }
319                 cout << "\nSuccess.\nRecord deleted." << endl;
320                 data.close();
321                 student.bilangan = student.bilangan - 1;
322             }break;
323 
324             case 'N': case 'n':
325                 cout << "Operation cancelled." << endl;
326                 break;
327             }
```

```
328         }
329         else {
330             cout << "Cannot delete student data" << endl;
331             cout << "This student still has outstanding fees." << endl;
332         }
333 
334     }
335     else {
336         cout << endl;
337         cout << "Sorry. Matric number entered not matching with any students."
              << endl;
338         for (int i = 0; i < z; i++)
339         {
340             data << student.name[i] << '$' << student.matno[i] << " " <<
                  student.course[i] << " " << student.fee[i] << " " << student.paid
                  [i] << " " << student.notPaid[i] << endl;
341         }
342     }
343 
344     cout << endl;
345     system("pause");
346 }
347 
```