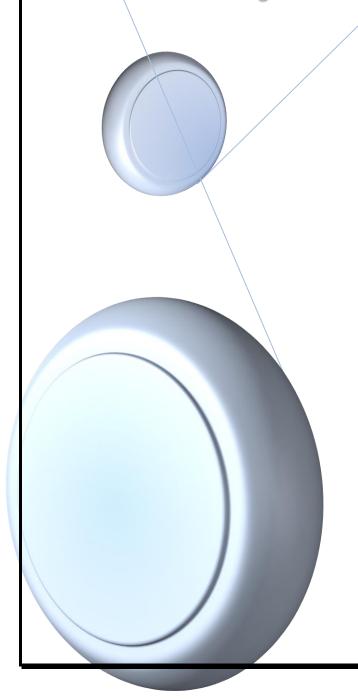


University Management System



إشراف الأستاذة / نجلاء أبو طالب

إعداد الطائبات:

بابل السُّبيبة بيان الرحبي

مها الجبوبي أسماء البحم

هلا الجيري وقية الجيري

شهد الجمائي

توثيق مشروع النظام جامعي

تم تطوير هذا النظام بلغة ++c لإدارة عمليات القبول والتسجيل في الجامعة

ويتضمن الميزات الاتية

١ ـ المكاتب المستخدمة في البرنامج

#include <iostream> #include <string> #include <vector> #include <map> //#include <limits>

استخدمنا مكتبة الإدخال والإخراج مكتبة السلاسل النصية ومكتبة vector تقوم بتخزین البیانات بداخلها علی شکل مصفوفات و مکتبة تخزن البيانات فيها بشكل مرتب تصاعدي

using namespace std;

أ عياكل البيانات

1. تركيب خاص في التخصصات الخاصة بالجامعة ويحتوي على

- اسم التخصص
- وصف التخصص
 - مدة الدراسة
- قائمة المقررات الدراسية

struct Specialization {

string name;

string description;

int duration;

vector<string> courses;

٢. تركيب خاص في كليات الجامعات ويحتوي على

- اسم الكلية
- متطلبات الكلية (خريطة من نوع مفتاح و قيمة)
 - قائمة التخصصات المتاحة
- vector<Specialization> specializations;

map<string, string> admissionRequirements;

س تركيب خاص في بيانات الطلاب ويحتوي على

- اسم الطالب
- الرقم الجامعي
- الكلية المسجل بها
- التخصص المختار

truct Student {

struct College {

string name;

string name;

string id;

string college;

string specialization;

bool parallelProgram;

}:

الصفحة ١

		 النظام الموازي
	الصفحة ٢	

٢ ـ قواعد بيانات النظام

vector<College> colleges;

vector<Student> registeredStudents;

استخدمنا قاعدتين بيانات لتخزين البيانات

- ١. قاعدة ببانات خاصة بالكلبات
- ٢. قاعدة بيانات خاصة بتخزين بيانات الطالب

ع دالة لتصنة السانات

قمنا بتهيئة النظام عن طريق إنشاء كليتين من نوع التركيب الخاص بالكليات

void initializeData() {

// Computer Science College

College cs;

cs.name = "College of Computer

csSpec1.name = "Computer Science";

programming fundamentals, data structures,

csSpec1.description = "Program covering

Science and Information";

cs.admissionRequirements = {

{"High School", "Minimum 80%

١. كلية الحاسيات

استخدمنا المتغير (cs) لتسمية الكلية عن طريف الاتي:

- اسم الكلية " كلية الحاسبات"
- متطلبات القبول {{" الثانوية العامة, المعدل على الأقل ٨٠%"} {"اختبار القبول المعدل على الأقل ٥٧%"}}

التخصص الأول لكلية الحاسبات

رمزنا له بالرمز (CSSpec1) ويحتوي على

- اسم التخصص " علوم الحاسب"
- وصف التخصص" هو برنامج يغطى اساسيات البرمجة و هياكل البيانات والذكاء الصناعي"
 - مدة الدر اسة= ٤ سنو ات

and AI\n\n"; csSpec1.duration = 4;

Specialization csSpec1;

- المواد الأساسية" البرمجة و هياكل البياثات و قاعدة البياثات. "csSpec1.courses = {"Programming 1",

"Data Structures", "Algorithms", "Database Systems\n\n"}:

التخصص الثاني لكلية الحاسب

رمزنا له بالرمز (CSSPEC2) ويحتوى على

- اسم التخصص " هندسة البرمجيات"
- وصف التخصص" هو برنامج يركز على تطوير البرمجيات وإداريها"
 - مدة الدراسة= ٤ سنوات
 - المواد الأساسية" هندسة البرمجيات و تطوير الويب و إدارة المشاريع"

Specialization csSpec2;

csSpec2.name = "Software Engineering";

csSpec2.description = "Program focused on

software development and management\n\n";

cs.specializations.push back(csSpec1);

csSpec2.duration = 4;

csSpec2.courses = {"Software Engineering",

"Web Development", "Software Testing", "Project

Management\n\n"}:

ويعد التهبئة قمنا باضافة البيانات الى قائمة التخصصات باستخدام دالة push back cs.specializations.push back(csSpec2);

colleges.push_back(cs);

وبعد ذلك نقوم بإضافة جميع البيانات الى قاعدة البيانات الخاصة بالكليات باستخدام نفس الدالة Push back

College eng; eng.name = "College of ٢ . كلية الهندسة Engineering"; استخدمنا المتغير (eng) لتسمية الكلية عن طريف الاتي: eng.admissionRequirements = { اسم الكلية " كلية هندسة الحاسوب" متطلبات القبول {{" الثانوية العامة, المعدل على الأقل ٥٥%"، ¡High School", "Minimum 85% in"} scientific track"}, {"اختبار القبول, المعدل على الأقل ٨٠ %"}} {"Aptitude Test", "Minimum 80%"}, ❖ التخصص الأول لكلية الهندسة رمزنا له بالرمز (engspec1) ويحتوي على Specialization engSpec1; engSpec1.name = "Computer Engineering"; - اسم التخصص "هندسة الحاسوب" - وصف التخصص" هو برنامج يجمع بين هندسة الإلكترونيات Program combining وصف التخصص" هو برنامج يجمع بين هندسة الإلكترونيات electrical engineering and computer و علوم الحاسوب" science\n\n"; - مدة الدر اسة= o سنوات engSpec1.duration = 5; - المواد الأساسية" الدوائر الإلكترونية و الإلكترونيات engSpec1.courses = {"Electrical Circuits", الرقمية ومعالجة الإشارات و شبكات الحاسوب..." "Digital Electronics", "Signal Processing", "Computer Networks\n\n"}; التخصص الثاني لكلية الهندسة رمزنا له بالرمز (engspec2) ويحتوي على Specialization engSpec2; - اسم التخصص " الهندسة الصناعية" engSpec2.name = "Industrial Engineering"; - وصف التخصص" هو برنامج يركز على تحسين engSpec2.description = "Program focused on النظم والعمليات" systems and process optimization\n\n"; - مدة الدراسة= ٥ سنوات engSpec2.duration = 5; - المواد الأساسية" بحوث العمليات و نظم التصنيع engSpec2.courses = {"Operations Research", وهندسة العوامل البشرية و إدارة الجودة....." "Human Factors Engineering", "Manufacturing Systems", "Quality Management\n\n"};

وبعد التهيئة قمنا بإضافة البيانات الى قائمة التخصصات eng.specializations.push_back(engSpec1); push_back باستخدام دالة push_back (engSpec2);

colleges.push back(eng);

وبعد ذلك نقوم بإضافة جميع البيانات الى قاعدة البيانات الخاصة بالكليات باستخدام نفس الدالة Push_back

واستخدمنا دالة ()cin.ignore لتنظيف الادخال من الاحرف الزائدة ومنع

لتنظيف الادخال من الاحرف الزائدة ومنع حدوث أخطاء في التبديل بين cin و ()getline وضمان قراءة الادخال بشكل صحيح من المستخدم دون سلوك غير متوقع

و دالة عرض القائمة الرئيسية

هذه الدالة نقوم من خلالها بعرض لقائمة التي تمكننا من الوصول الي أي جزء من النظام الجامعي

- ١. عرض التخصصات
 - ٢. متطلبات القبول
 - ٣. عملية التسجيل
 - ٤. النظام الموازي
- ه. عرض بيانات الطلاب
 - ٦. الخروج

```
void showMainMenu() {
   cout << "
               University Admission System\n";
   cout << "=======\n";

    View Available Specializations\n";

   cout << "
               Admission Requirements\n";
   cout <<
               Registration Process\n";
   cout <<
   cout << "
               Parallel Program Information\n";
    cout << '
                5. View Registered Students\n";
   cout << "
   cout << "=======\n";
                    Enter your choice: ";
```

٦ دالة عرض التخصصات المتاحة

هذه الدالة تعرض جميع الكليات والتخصصات مع تفاصيل كل تخصص

```
عملنا دوارة قمنا بتعريف داخلها
                                                                              متغیر (college) لتخزین اسم
void showSpecializations() {
    cout << "Available University Specializations:\n";</pre>
    for (const College& college : colleges) {
        cout << "\nCollege: " << college.name << "\n\n";</pre>
        cout << "Specializations:\n\n";</pre>
        for (const Specialization& spec :
college.specializations) {
            cout << "- " << spec.name <<
spec.duration << " years)\n";</pre>
            cout << "
                           *Description*:
spec.description << "\n";</pre>
            cout << "
                           *Core Courses*: ";
             for (const string& course : spec.courses) {
                cout << course << ", ,"
            cout << "\n";</pre>
    cout << "\nPress any key to return to main menu...";</pre>
    cin.ignore();
    cin.get();
```

الكلية المأخوذة من قاعدة البيانات(colleges)

وعرفنا ايضا دوارة بداخل الدوارة الاولى وقمنا بتعریف متغیر (spec)داخلها لتخزين اسم و وصف التخصص ومدة دراسته من قائمة التخصصات

وأيضا عرفنا دوارة خاصه بالمواد بداخلها متغير (course) لتخزين المواد من قائمة التخصص

}

٧ دالة عرض متطلبات القبول

```
هذه الدالة تظهر كل متطلبات القبول لكل كلبة
void showAdmissionRequirements()
    cout << "College Admission Requirements:\n\n";</pre>
                                                                           عملنا دوارة قمنا بتعريف داخلها
                                                                          متغیر (college) لتخزین اسم
    for (const College& college : colleges) {
                 "\nCollege: " << college.name << "\n";
                                                                                                الكلبة
                       *Requirements*:\n";
                                                                   المأخوذة من قاعدة البيانات(colleges)
        for (const auto& reg : college.admissionReguirements)
                                                                      عرفنا دوارة لعرض المتطلبات وعرفنا
<< "\n";
                                                                             بداخلها متغیر من نوع auto
                                                                                  وخزنا بداخلة المتطلبات
    cout << "\n*Note*\n: Requirements may</pre>
university policy each year.\n";
                                                                 وأيضا طبعنا ملاحظة تنص على ان الشروط
    cout << "\nPress any key to return to main menu...";</pre>
                                                                                  تتغير من كلية الى أخرى
   cin.ignore();
   cin.get();
                                                                 }
                                                                            مطومات عن MAP, AUTO
          Map: هي حاوية تخزن البيانات في ازواج (مفتاح و قيمة) في نظامنا سنخزن شروط القبول حيث المفتاح
                                                                       هو اسم الشرط والقيمة في التفاصيل.
                                                                           الوصول الى العناصر في map:
```

هناك طريقتين للوصول , ۲. الوصول باستخدام المعامل () at. ١. الوصول باستخدام معامل []

ال map في ++c يحفظ جميع العناصر مرتبة حسب المفتاح (بترتيب تصاعدي افتراضيا) مما يجعل البحث والوصول الى العناصر سريعا

مميزات loop في map: ١. الإيجاز/ تقليل عدد الكود المطلوب ٢. الأمان/ تجنب الأخطاء الشائعة في استخدام الكود.

استخدام auto في ال map:

تستخدم لاستنتاج نوع عناصر الmap<string, تلقائياً في حالة <map

وبدون ال auto يمكن كتابة الكود يدوياً (...auto auto يمكن كتابة الكود يدوياً (...for (const pair <string, const string

الفائدة من auto

يختصر الكود ويجعله اكثر قراءة وأيضا يتجنب الأخطاء في كتابة النوع يدوياً

٨ . دالة لعملية التسجيل

```
في هذه الدالة نقوم بتسجيل الطالب في الجامعه من خلال الاتي
void registrationProcess() {
    cout << "Registration Process\n";</pre>
    if (colleges.empty()) {
                                                           ١. توجيه الطالب لإدخال البيانات الشخصية
                                                                       في هذا الكود يتحقق في البداية من
time.\n";
                                             وجود كليات متاحه إذا لم يكن هناك كليات :"...to main menu
        cin.ignore();
                                                                      متاحه يطبع "لا يوجد كليات متاحه"
        cin.get();
                                                                ثم يقوم بإدخال اسم الطالب ورقمه الجامعي
        return;
    Student newStudent;
    cout << "Full Name:</pre>
    getline(cin, newStudent.name);
    cout << "University ID: ";</pre>
    getline(cin, newStudent.id);
                                                                       ٢. اختيار الطالب للكلية
cout << "\nAvailable Colleges:\n";</pre>
                                                             استخدمنا دوارة لاختيار كلية محدد في النظام
   for (int i = 0; i < colleges.size(); i++) {</pre>
        cout << i+1 << ". " << colleges[i].name << "\n";</pre>
                                                                                   عرفنا متغير للاختيار
                                                                    وعملنا شرط بنص على ان المستخدم
    int collegeChoice;
                                                                      إذا ادخل رقم اقل من ١ او رقم اكبر
    cout << "Select college (number): ";</pre>
    cin >> collegeChoice;
    if (collegeChoice < 1 || collegeChoice > colleges.size()) {
                                                                            يطبع " الاختيار غير متوفر"
        cout << "Invalid selection!\n";</pre>
        cout << "\nPress any key to return to main menu...";</pre>
        cin.ignore();
                                                             اذا لم يتحقق الشرط يقوم النظام بإضافة الكلية
        cin.get();
        return;
                                                                               المختارة الى سجل الطالب
    newStudent.college = colleges[collegeChoice-1].name;
cout << "\nSpecializations available in " <<</pre>
newStudent.college << ":\n";</pre>
                                                                  ٣. اختيار الطالب للتخصص
    for (int i = 0; i < colleges[collegeChoice-</pre>
1].specializations.size(); i++) {
      cout << i+1 << ". " << colleges[collegeChoice-</pre>
1].specializations[i].name << "\n";</pre>
                                                              استخدمنا دوارة لاختيار كلية محدد في النظام
                                                                                   عرفنا متغير للاختيار
    int specChoice;
   cout << "Select specialization (number): ";</pre>
                                                                     وعملنا شرط ينص على ان المستخدم
    cin >> specChoice;
                                                                       إذا ادخل رقم اقل من ١ او رقم اكبر
   if (specChoice < 1 || specChoice >
colleges[collegeChoice-1].specializations.size()) {
                                                              من حجم قاعدة البيانات الخاصة بالتخصصات
        cout << "Invalid selection!\n";</pre>
                                                                            يطبع " الاختيار غير متوفر"
        cout << "\nPress any key to return to main menu...";</pre>
        cin.ignore();
       cin.get();
                                                         اذا لم يتحقق الشرط يقوم النظام بإضافة التخصص
                                                                               المختارة الى سجل الطالب
   newStudent.specialization = colleges[collegeChoice-
1].specializations[specChoice-1].name;
```

- الصفحة ٧

```
char parallelChoice;
                                                               ٤. اختيار الالتحاق بالنظام الموازي
    cout << "Would you like to join the Parallel Program?</pre>
(y/n): ";
   cin >> parallelChoice;
                                                                 يختار المستخدم إحدى الخيارين
    newStudent.parallelProgram = (parallelChoice == 'y' | |
                                                                                              (y,n)
parallelChoice == 'Y');
    registeredStudents.push_back(newStudent);
                                                      ومن ثم يقوم النظام بإضافة بيانات الطالب
                                                      الجديد الى قاعدة البيانات الخاصة بالطالب
                                                                     باستخدام دالة push back
cout << "\nRegistration successful!\n";</pre>
    cout << "Your information:\n";</pre>
                                                                عرض بيانات الطالب المسجل
    cout << "Name: " << newStudent.name << "\n";</pre>
                                                                           ۔ اسم الطالب
    cout << "University ID: " << newStudent.id << "\n";</pre>
    cout << "College: " << newStudent.college << "\n";</pre>
                                                                         - الرقم الجامعي
    cout << "Specialization: " <<</pre>
newStudent.specialization << "\n";</pre>

    الكلبة المختارة

    cout << "Parallel Program: " <<</pre>
(newStudent.parallelProgram ? "Yes" : "No") <<</pre>
                                                                      - التخصص المختار
    cout << "\nPress any key to return to main menu...";</pre>
    cin.ignore();
    cin.get();
                      ٩ ـ دالة عرض معلومات النظام المواي
 void showParallelProgramInfo() {
                                                                                     في هذه الدالة نقوم
     cout << "Parallel Program\n";</pre>
                                                                            بشرح ميزات النظام الموازى
      cout << "What is the Parallel Program?\n";</pre>
     cout << "The Parallel Program is an academic program</pre>
  university education opportunities\n";
                                                                                              والتكاليف
     cout << "for students who didn't get admission</pre>
  with higher tuition fees.\n\n";
                                                                                        وشروط الالتحاق
      cout << "Program Features:\n";</pre>
     cout << "- Same curriculum as the regular program\n";</pre>
     cout << "- Same faculty members\n";</pre>
     cout << "- Same awarded degrees\n\n";</pre>
      cout << "Tuition Fees:\n";</pre>
     cout << "- Fees vary by college and specialization\n";</pre>
     cout << "- Average fees between 20,000 to 40,000 SAR annually\n\n";</pre>
      cout << "Admission Requirements:\n";</pre>
     cout << "- High school diploma\n";</pre>
      cout << "- Meeting college requirements (typically 5-10% lower than</pre>
  regular program)\n\n";
     cout << "Note: Transfer from parallel to regular program is possible</pre>
  according to university policies.\n";
```

الصفحة ٨

cout << "\nPress any key to return to main menu...";</pre>

١٠. دالة عرض الطلاب المسجلين في الجامعة

```
void showRegisteredStudents(`
    cout << "Registered Students\n\n";</pre>
                                                        في هذه الدالة نقوم بعرض الطلاب الذي تم تسجيلهم
    if (registeredStudents.empty()) {
        في النظام وقبل ما يتم العرض نتحقق هل يوجد "cout << "No students have registered yet.\n";
        cout << "Total Registered Students: " <<</pre>
                                                                                            طلاب مسجلین ام لا
registeredStudents.size() << "\n\n";
                                                                                فى حالة وجود طلاب مسجلين
        for (const Student& student : registeredStudents) {
             cout << "Student Name: " << student.name << "\n";
cout << "University ID: " << student.id << "\n";
cout << "College: " << student.college << "\n";</pre>
             نقوم بعرض بياناتهم المخزنة في قاعدة أز"n"; و student.college << "\n"; نقوم بعرض بياناتهم المخزنة في قاعدة أزادة و cout << "Specialization: " << student.specialization <<
\n";
                                                 البيانات الخاصة بالطلاب المسجليوstudent.parallelProgram)
 "Yes" : "No") << "\n";
                                                                                          مع تفاصيل كل طالب
    cout << "\nPress any key to return to main menu..</pre>
    cin.ignore();
    cin.get();
                                                                           }
 .nt main() {
    initializeData();
                                           11. الدالة الرئيسية
     int choice;
         showMainMenu();
         cin >> choice;
         cin.ignore(numeric_limits<streamsize>::max(), '\n');
         switch (choice) {
             case 1:
                                                    في هذه الدالة عملنا تهيئة للبيانات الموجودة في النظام
                  showSpecializations();
                  break;
                                                     وقمنا بعمل حلقة تكرار لعرض القائمة الرئيسية وتنفيذ
             case 2:
                  showAdmissionRequirements();
                  break;
                                                                            الخيارات حتى يتم اختيار الخروج
             case 3:
                  registrationProcess();
                  break;
                                                              وداخل حلقة التكرار استخدمنا switch للتحكم
             case 4:
                  showParallelProgramInfo();
                  break;
                                                              في نوع الاختيار التي تم اختياره من المستخدم
             case 5:
                  showRegisteredStudents();
                  break;
             case 6:
                  cout << "Thank you for using the University</pre>
Admission System. Goodbye!\n";
                  break;
             default:
                  cout
                            'Invalid choice, please try
again.\n";
      while (choice != 6);
    return 0;
                                                    الصفحة ٩
```