**TITTLE:-STRUCTURE AND ARRAY**

**OBJECTIVE:-To utilize structure and array for DSA**

**THEORY**:-

• DSA : A data structure Is a named location that is used to store and organize data. And, an algorithm is a collection of steps to solve a particular problem. Learning data structures and algorithms allow us to write optimized and efficient computer programs.

• Structure : It is a user-defined data type in C language which allows us to combine data of different types together. It helps to construct a complex data type . Structure of structure is called nesting structure which can a build a powerful data structure.

• ARRAY : An array is a collection of similar data items stored at contiguous memory locations and by using which elements can be accessed. They are used to store collection of primitive data types such as int , float, double, char, etc of any particular type.

**Problem**: Component development for the result evaluation system.

Description: Nepal Technical University teaches only three subjects in their special vocational program. The following 2D table describes the result of the internal examination of 5 students.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CRN | Name | Programming | Maths | Physics |
| 301 | Hari Kunwar | 45 | 60 | 36 |
| 302 | Manita Thapa | 52 | 15 | 65 |
| 303 | Puskar Shah | 78 | 85 | 79 |
| 304 | Usha Karki | 48 | 45 | 45 |
| 305 | Bikash Rajat | 92 | 95 | 88 |

1. Represent the students and their associated marks using array of structures(class).

2. Assume, the pass percentage is 45%, write a function to count the number of students failing in each subjects.

3.Given the following result categories (Division)

a. Less than 45%: Fail

b. Above 45%: Pass

c. Above 50%: Second Division

d. Above 75%: First Division

e. Above 90%: Distinction.

The software has a feature of SMS services to their clients. To integrate with SMS services, write a function that takes an input of CRN of the student and returns the result category (division) of the student.

4. The software has a feature of report generation which contains the results of all the students. To achieve that write a function to display the results in the following format.

|  |  |  |
| --- | --- | --- |
| CRN | Name | Division |
| 301 | Hari Kunwar | ???? |
| 302 | Manita Thapa | ???? |
| 303 | Puskar Shah | ???? |
| 304 | Usha Karki | ???? |
| 305 | Bikash Rajat | ???? |

**Implementation:**

Source Code:

#include<conio.h>

#include<stdlib.h>

#include<string.h>

#include<iostream>

using namespace std;

int i;

struct Subject{

char name[30];

int marks;

};

struct Student{

int crn;

char name[30];

Subject subject[3];

};

Student \* initialize(){

static Student student[5];

student[0].crn=301; /\*first student data initialization \*/

strcpy(student[0].name,"Hari Kunwar");

strcpy(student[0].subject[0].name, "Programming");

student[0].subject[0].marks=45;

strcpy(student[0].subject[1].name, "Maths");

student[0].subject[1].marks=60;

strcpy(student[0].subject[2].name, "Physics");

student[0].subject[2].marks=36;

student[1].crn=302; /\*second student data initialization \*/

strcpy(student[1].name,"Manita Thapa");

strcpy(student[1].subject[0].name, "Programming");

student[1].subject[0].marks=52;

strcpy(student[1].subject[1].name, "Maths");

student[1].subject[1].marks=15;

strcpy(student[1].subject[2].name, "Physics");

student[1].subject[2].marks=65;

student[2].crn=303; /\*third student data initialization \*/

strcpy(student[2].name,"Puskar Shah");

strcpy(student[2].subject[0].name, "Programming");

student[2].subject[0].marks=78;

strcpy(student[2].subject[1].name, "Maths");

student[2].subject[1].marks=85;

strcpy(student[2].subject[2].name, "Physics");

student[2].subject[2].marks=79;

student[3].crn=304; /\*fourth student data initialization \*/

strcpy(student[3].name,"Usha Karki");

strcpy(student[3].subject[0].name, "Programming");

student[3].subject[0].marks=48;

strcpy(student[3].subject[1].name, "Maths");

student[3].subject[1].marks=45;

strcpy(student[3].subject[2].name, "Physics");

student[3].subject[2].marks=45;

student[4].crn=305; /\*fifth student data initialization \*/

strcpy(student[4].name,"Bikash Rajat");

strcpy(student[4].subject[0].name, "Programming");

student[4].subject[0].marks=92;

strcpy(student[4].subject[1].name, "Maths");

student[4].subject[1].marks=95;

strcpy(student[4].subject[2].name, "Physics");

student[4].subject[2].marks=88;

return student;

}

float calculatePercentage(Student \*individual){/\*%calculation \*/

int total;

total=individual->subject[0].marks+individual->subject[1].marks+individual->subject[2].marks;

return total/3;

}

void PassStudentCount(Student \* student){

int count=0;

for (i=0; i<5; i++){

if (calculatePercentage(student+i)>=45){

count++;

}

}

cout<<"Pass student= "<<count<<endl;

}

string calculateDivision(Student \* student){

if(calculatePercentage(student)>=90){

return "Distinction";

}else if(calculatePercentage(student)>=75){

return "First Division";

}else if(calculatePercentage(student)>=50){

return "Second Division";

}else if(calculatePercentage(student)>=45){

return "Pass";

}else{

return "Fail";

}

}

void IndividualResult(Student \*student){

int input;

cout<<"CRN number: ";

cin>>input;

for(i=0; i<5;i++){

if((student+i)->crn==input){

cout<<"Division is: "<<calculateDivision(student+i)<<endl;

}

}

}

void displayReport(Student \*student){

cout<<"CRN "<<" Name "<<" Division "<<endl;

for(i=0;i<5;i++){

cout<<(student+i)->crn<<" "<<(student+i)->name<<" "<<calculateDivision(student+i)<<endl;

}

}

int main(){

Student \*allStudents;

allStudents=initialize();

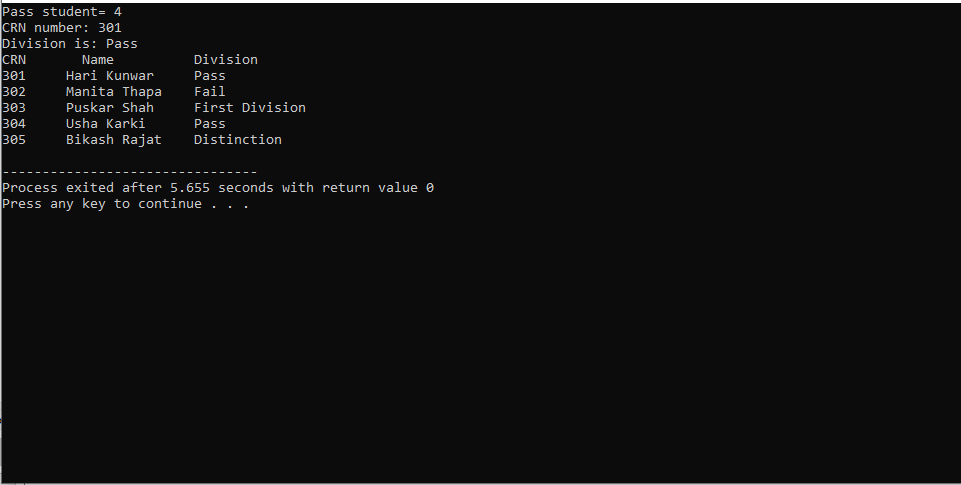
PassStudentCount(allStudents);

IndividualResult(allStudents);

displayReport(allStudents);

return 0;

}



**Assumptions** : The source code was started by declaring the structure of student. As another structure for name and marks of subject was required, it was created in the name "Subject". Then the function to initialize the structure was created which returns the pointer to structure . The purpose of "hasPassedAllSubject()" is to see if a student has passed each and every subject or not. The function returns 1 if student has passed all subject, else returns 0. Then function is used for result calculation of student. Then we define a function to calculate no of student failed for every subject. The function to check the result of individual student was defined and all required function are called from main() and user defined function .

**CONCLUSION :** Hence , We became familiar with the process of solving Real World Problem by using structure.