**Practical 10 – refer to Topic 19**

**Part A(Understanding Concepts)**

1. Write statements to declare a structure type that represents a point with x- and y-coordinate values. Write two versions, one using keyword *struct* and a tag name POINT and another using keywords *typedef struct* and a type name POINT\_TYPE.

Struct POINT

{

Int x\_coordinate;

Int y\_coordinate;

};

typedef struct

{

Int x\_coordinate;

Int y\_coordinate;

} POINT\_TYPE;

1. Referring to the structure types POINT in question 1, write statement to declare a variable to represent a point with coordinates (3,5).

struct POINT pt;

pt.x\_coordinate = 3;

pt.y\_coordinate = 5;

struct POINT pt = { 3, 5};

1. Referring to the structure type POINT\_TYPE in question 1, write statement to declare a variable to represent a point, ask the user to input the values for the x-and y-coordinates, and then display the values.

POINT\_TYPE pt;

cout << “ Enter x-coordinate: “;

cin >> pt.x\_coordinate;

cout << “Enter y\_coordinate: “;

cin >> pt.y\_coordinate;

cout << “The point is (“pt.x\_coordinate << “, “ << pt.y\_coordinate <<”)\n”

1. Referring to structure type POINT\_TYPE in question 1 and the declaration below, write statements to create a variable to represent a line whose end points have coordinates (3,5) and (-1,10).

typedef struct

{

POINT\_TYPE endpoint1;

POINT\_TYPE endpoint2;

}LINE\_TYPE;

LINE\_TYPE line;

Line.endpoint1.x\_coordinate = 3;  
Line.endpoint1.y\_coordinate = 5;

Line.endpoint2.x\_coordinate = -1;  
Line.endpoint2.y\_coordinate = 10;

1. Referring to the structure type POINT\_TYPE in question 1, write a statement to declare an array of structures of size 10 to store 10 points.

POINT\_TYPE points[10];

1. Write statements to:
2. declare a structure type using keywords *typedef struct* with the type name ACCOUNT\_TYPE that represents a bank account with 3 members: name (string), number (integer), and balance (double).

{

char name[30];

int number;

double balance;

}ACCOUNT\_TYPE;

1. create 2 variables to represent 2 accounts with values as shown below. Note: use function strcpy to assign the name values.

|  |  |  |
| --- | --- | --- |
|  | Account 1 | Account 2 |
| Name | Alice Tan | Alex Chong |
| Account Number | 123456789 | 987654321 |
| Balance | 2,345.60 | 3,678.95 |

ACCOUNT\_TYPE account1, account2l

strcpy(account1.name, “Alice Tan”);

account1.number = 123456789;

account1.balance = 2345.60;

strcpy(account2.name, “Alex Chong”);

account2.number = 987654321;

account2.balance = 3678.95;

**Part B (Programming Exercises)**

1. Given the declaration below, write and test a function that has 2 parameters: a parameter of structure type FRACTION (given below) representing a fraction and an integer parameter. The function multiplies the numerator of the fraction by the integer and returns the resulting fraction through its return statement. For example, if the fraction is 2/15 and the integer is 7, then the resulting fraction is 14/15.

typedef struct

{

int numerator;

int denominator;

} FRACTION;

#include <iostream>

using namespace std;

typedef struct

{

int numerator;

int denominator;

} FRACTION;

FRACTION multiply(FRACTION f1, int num);

int main(void)

{

FRACTION f1, f2;

int num;

cout << "Eter the first fraction numerator: ";

cin >> f1.numerator;

cout << "Enter the first fraction denominator: ";

cin >> f1.denominator;

cout << "Enter integer to multiply: ";

cin >> num;

f2 = multiply(f1, num);

cout << "Result is" << f2.numerator << "/" << f2.denominator << endl;

return 0;

}

FRACTION multiply(FRACTION f1, int num)

{

FRACTION f2;

f2.numerator = f1.numerator \* num;

f2.denominator = f1.denominator;

return f2;

}

**Part C (Self-Review / Revision)**

1. What is the purpose of typedef?

the typedef creates a new type by giving a name to a data type.

1. What is a structure?

a structure is a collection of elements of possibly different data types

1. What are 2 ways to declare a structure type?

using a keyword struct or keywords typedefstruct

1. How do you access a member of a structure type variable?

Using a variable-name.member-name ( where the dot is the selection operator)

**Part D (Practice Exercises)**

Write a C/C++ program that uses an Array to store structures and, preferably, uses a menu with the following options:

|  |  |  |
| --- | --- | --- |
| Read Grade | - | asks a user for one student’s name, two grades, and calculates the average of two grades to store into the Array. |
| List | - | prints out all the student names and grades stored in the Array onto the screen. |
| Quit | - | stops the program. |

Note: You may assume that Array size will not exceed 30, all strings will not exceed 30 characters in length, and grades are from 0 to 100. The following structure is adequate:

typedef struct

{

char name[31];

int grade1;

int grade2;

double average;

} STUDENTREC;

#include <iostream>

#include <iomanip>

#include <cstdlib>

#include <cstring>

#define SIZE 30

using namespace std;

typedef struct

{

char name[31];

int grade1;

int grade2;

double average;

} STUDENTREC;

int read\_grade(STUDENTREC stud\_arr[], int index);

void list(const STUDENTREC stud\_arr[], int size);

int main(void)

{

int action, index = 0;

bool quit = false;

STUDENTREC students[SIZE];

while(!quit)

{

cout <<"\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

cout <<"\n\t\t\*\*\tStudent Records Menu\t\t\*\*";

cout <<"\n\t\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

cout <<"\t\t<1> Read Grade.\n";

cout <<"\t\t<2> List.\n";

cout <<"\t\t<3> Quit.\n\n";

cout <<"\tPlease enter your option --> ";

cin >> action;

cout <<"\n\n";

if(action == 1)

index = read\_grade(students, index);

else if(action == 2)

list(students, index);

else if(action == 3)

quit = true;

else

cout <<"\tWrong selection.\n";

}

}

int read\_grade(STUDENTREC stud\_arr[], int index)

{

if(index >= SIZE)

cout << "Array is full.\n";

else

{

cout <<"\tPlease enter the student name : ";

fflush(stdin);

cin.get(stud\_arr[index].name, 31);

cout <<"\tPlease enter the student grade 1: ";

cin >> stud\_arr[index].grade1;

cout <<"\tPlease enter the student grade 2: ";

cin >> stud\_arr[index].grade2;

stud\_arr[index].average = (double)(stud\_arr[index].grade1 + stud\_arr[index].grade2) / 2.0;

cout <<"\tThe average of the student grades is "

<< fixed << setprecision(2) << stud\_arr[index].average << endl;

index++;

}

return index;

}

void list(const STUDENTREC stud\_arr[], int size)

{

if(size <= 0)

{

cout << "\tNo record.\n";

return;

}

cout <<"Name\t\t\tGrade1\t\tGrade2\t\tAverage\n";

for(int i = 0; i < size; i++)

{

if (strlen(stud\_arr[i].name) < 15)

cout << stud\_arr[i].name << "\t\t"

<< setw(3) << stud\_arr[i].grade1 << "\t\t"

<< setw(3) << stud\_arr[i].grade2 << "\t\t"

<< setw(6) << fixed << setprecision(2)

<< stud\_arr[i].average << endl;

else

cout << stud\_arr[i].name << "\t"

<< setw(3) << stud\_arr[i].grade1 << "\t\t"

<< setw(3) << stud\_arr[i].grade2 << "\t\t"

<< setw(6) << fixed << setprecision(2)

<< stud\_arr[i].average << endl;

}

}