

**LAB 2**

Submitted by:

Name: Haseeb Ullah

ID No: F20232661009

Section: V12

Submitted to:

M. OWAIS KHAN

**STRUCTURE**

Date : 03/11/2023

C-II Block C 2 Phase 1 Johar Town, Lahore, Punjab 54770.

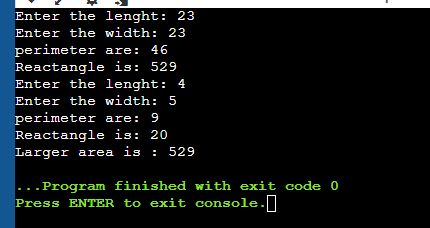
Lab Task 1

"Write a C++ program that defines a structure called 'rectangle' with members for 'length' and 'width'. The program should then prompt the user to enter the dimensions of two rectangles and store them in variables of type 'rectangle'. Finally, display the area and perimeter of each rectangle, and determine which rectangle has the larger area."

**Source Code:**

|  |
| --- |
| #include <iostream>  using namespace std;  struct rectangle  {  double lenght, width;  };  double  rectangles (double a, double b)  {  double rect = a \* b;  return rect;  }  int  main ()  {  rectangle rect[2];  double a[2];  for (int i = 0; i < 2; i++)  {  cout << "Enter the lenght: ";  cin >> rect[i].lenght;  cout << "Enter the width: ";  cin >> rect[i].width;  a[i] = rectangles (rect[i].lenght, rect[i].width);  cout << "perimeter are: " << rect[i].lenght + rect[i].width << endl;  cout << "Reactangle is: " << a[i]<<endl;  }  // for(int i = 0; i <2)  if (a[0] > a[1])  {  cout << "Larger area is : " << a[0];  }  else  {  cout << "Larger area is : " << a[1];  }  return 0;  } |

**Output:**

****

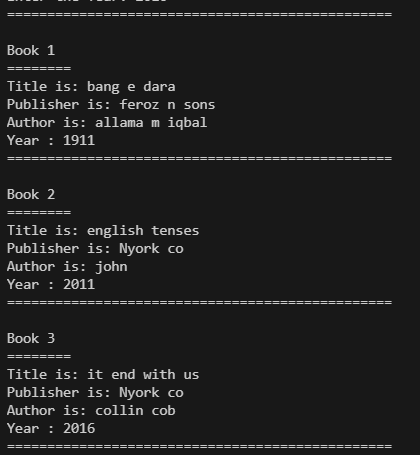
Lab Task 2:

"Define a C++ structure called 'book' with members for 'title', 'author', 'publisher', and 'year of publication'. Then, write a program that prompts the user to enter data for three books and stores it in an array of 'book' structures. Finally, display the data for all three books in a formatted way that includes all four members, and sort the books in alphabetical order by title."

**Source code:**

|  |
| --- |
| #include <iostream>  using namespace std;  int main()  {  struct book  {  string title, publisher, author, year;  };  cout << "Enter total book you want to store : ";  int size;  cin >> size;  struct book book[size];  for (int i = 0; i < size; i++)  {  cin.ignore();  cout << "Book " << i + 1 << endl;  cout << "Enter the Title of the book: ";  getline(cin, book[i].title);  cout << "Enter the Publisher name: ";  getline(cin, book[i].publisher);  cout << "Enter the Author of the book: ";  getline(cin, book[i].author);  cout << "Enter the Year: ";  cin >> book[i].year;  }  cout << "================================================" << endl;  for (int i = 0; i < size; i++)  {  cout << "\nBook " << i + 1 << endl;  cout << "========" << endl;  cout << "Title is: " << book[i].title << endl;  cout << "Publisher is: " << book[i].publisher << endl;  cout << "Author is: " << book[i].author << endl;  cout << "Year : " << book[i].year << endl;  cout << "================================================" << endl;  }  for (int i = 0; i < size; i++)  {  if (book[i].title[0] >= 'A' && book[i].title[0] <= 'Z' || book[i].title[0] >= 'a' && book[i].title[0] <= 'z')  {  struct book temp = book[i];  book[i] = book[i + 1];  book[i + 1] = temp;  }  }  cout << "||||||||||||||||||||||||||||||||||||||||||||||||" << endl;  cout << "================================================" << endl;  for (int i = 0; i < size; i++)  {  cout << "Book " << i + 1 << endl;  cout << "========" << endl;  cout << "Title is: " << book[i].title << endl;  cout << "Publisher is: " << book[i].publisher << endl;  cout << "Author is: " << book[i].author << endl;  cout << "Year : " << book[i].year<<endl;  cout << "================================================" << endl;  }  return 0;  } |

**Output:**

****

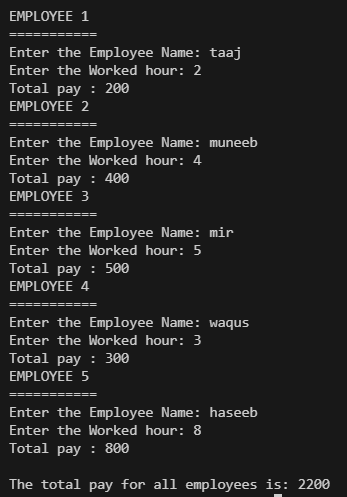
Lab Task 3:

"Write a C++ program that defines a structure called 'employee' with members for 'name', 'ID number', 'hourly wage', and 'hours worked'. The program should then prompt the user to enter data for five employees and store it in an array of 'employee' structures. Finally, display the total pay for each employee (hourly wage \* hours worked) and the total pay for all employees combined."

**Source code:**

|  |
| --- |
| #include <iostream>  using namespace std;  struct employee  {  string name;  static double hourly\_wage;  int id\_number,  worked\_hour;  };  double employee ::hourly\_wage = 100;  double totalpay(double hours, double wage)  {  return hours \* wage;  }  int main()  {  employee employees[5];  double total = 0;  // "Write a C++ program that defines a structure called 'employee' with members for 'name', 'ID number', 'hourly wage', and 'hours worked'.  // The program should then prompt the user to enter data for five employees and store it in an array of 'employee' structures.  // Finally, display the total pay for each employee (hourly wage \* hours worked).  // The total pay for all employees combined."  for (int i = 0; i < 5; i++)  {  cin.ignore();  cout << "EMPLOYEE "<< i + 1 <<endl;  cout << "==========="<<endl;  cout << "Enter the Employee Name: ";  getline(cin, employees[i].name);  cout << "Enter the Worked hour: ";  cin >> employees[i].worked\_hour;  cout << "Total pay : " << totalpay(employees[i].worked\_hour, employee::hourly\_wage) << endl;  total += totalpay(employees[i].worked\_hour, employee::hourly\_wage);  }  cout << "\nThe total pay for all employees is: " << total << endl;  return 0;  } |

**Output:**

****

Lab Task 4:

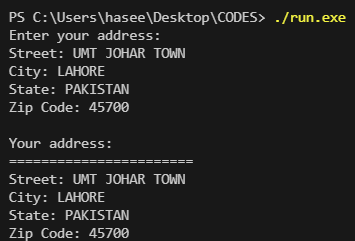
"Define a C++ structure called 'address' with members for 'street', 'city', 'state', and 'zip code'. Then, write a program that prompts the user to enter their address and stores it in a variable of type 'address'. Finally, display the address in a formatted way that includes all

four members."

**Source Code:**

|  |
| --- |
| // "Define a C++ structure called 'address' with members for 'street', 'city', 'state', and 'zip code'.  // Then, write a program that prompts the user to enter their address and stores it in a variable of type 'address'.  // Finally, display the address in a formatted way that includes all  // four members."  #include <iostream>  using namespace std;  struct address {  string street;  string city;  string state;  int zip\_code;  };  int main() {  address address;  cout << "Enter your address:" << endl;  cout << "Street: ";  getline(cin, address.street);  cout << "City: ";  getline(cin, address.city);  cout << "State: ";  getline(cin, address.state);  cout << "Zip Code: ";  cin >> address.zip\_code;  cout << "\nYour address:" << endl;  cout << "======================="<<endl;  cout << "Street: " << address.street << endl;  cout << "City: " << address.city << endl;  cout << "State: " << address.state << endl;  cout << "Zip Code: " << address.zip\_code << endl;  return 0;  } |

**Output:**

****

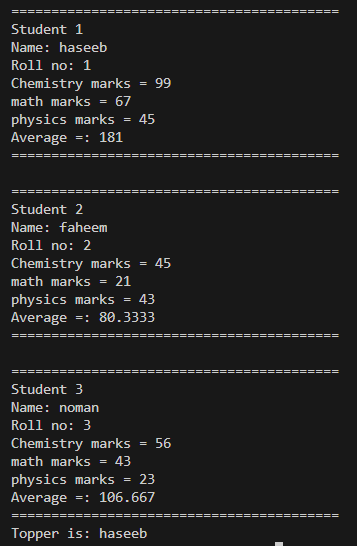
Lab Task 5:

"Write a C++ program that defines a structure named 'student' with the following members: 'name', 'roll number', 'marks in physics', 'marks in chemistry', and 'marks in mathematics'. Prompt the user to enter the data for three students and store it in an array of 'student' structures. Then, display the data for all three students in tabular format, including their total marks and average marks. Finally, display the name of the student with the highest marks."

**Source Code:**

|  |
| --- |
| #include <iostream>  using namespace std;  struct marks{  double roll\_no;  char name[10];  double chem,phy,math;  int obtainmarks;  };  void highest(marks s[], int size)  {  double maximum = 0;  string topper;  for (int i = 0; i < size; ++i) {  if (s[i].obtainmarks > maximum) {  maximum = s[i].obtainmarks;  topper = s[i].name;  }  }  cout << "Topper is: "<< topper<<endl;  }  double average\_marks(double a,double b,double c)  {  double average = a + b + c / 3 ;  return average;  }  int main()  {  marks s[3];  double total = 300;  for (int i = 0; i < 3; i++)  {  s[i].roll\_no = i + 1;  cout << "Student "<< i + 1 <<endl;  cout << "Enter Name: ";  cin >> s[i].name;    cout << "Enter the chemistry: ";  cin >>s[i].chem ;  cout << "Enter the math: ";  cin >>s[i].math ;  cout << "Enter the physics: ";  cin >>s[i].phy ;    s[i].obtainmarks = s[i].chem + s[i].math +s[i].phy;  }      for (int i = 0; i < 3; i++)  {  cout << "\n========================================= " <<endl;  cout << "Student "<< i + 1 <<endl;  cout << "Name: "<< s[i].name <<endl;  cout << "Roll no: " << s[i].roll\_no<<endl;  cout << "Chemistry marks = "<< s[i].chem<<endl;  cout << "math marks = "<< s[i].math<<endl;  cout << "physics marks = "<< s[i].phy<<endl;  cout << "Average =: " << average\_marks(s[i].chem, s[i].math, s[i].phy)<<endl;  cout << "========================================= " <<endl;      }  highest (s,3);  return 0;  } |

**Output:**

****

Lab Task 6:

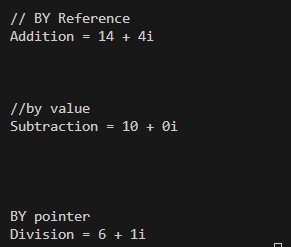
"Write a C++ program (one or more) that defines a structure on your interested programming example and demonstrate the following in that program.

* • Use array in structure
* • Use the concept of Pointers with structure
* • Pass the structure variable to function and demonstrate the call by value and call by
* • function with structure.

**Source Code:**

|  |
| --- |
| #include <iostream>  using namespace std;  struct complex  {      int real, imaginary;  } c1, c2;  // BY Reference  struct complex add(struct complex &a, struct complex &b)  {      struct complex addition;      addition.real = a.real + b.real;      addition.imaginary = a.imaginary + b.imaginary;      return addition;  }  // by value  struct complex sub(struct complex a, struct complex b)  {      struct complex subtraction;      subtraction.real = a.real - b.real;      subtraction.imaginary = a.imaginary - b.imaginary;      return subtraction;  }  void divid(complex \*a, complex \*b)  {      struct complex division;      division.real = a->real / b->real;      division.imaginary = a->imaginary / b->imaginary;      cout << "Division = " << division.real << " + " << division.imaginary << "i" << endl;  }  int main()  {      // Write a program to      // add,      // subtract complex numbers using structures to function.      struct complex c1 = {12, 2};      struct complex c2 = {2, 2};      struct complex additoin = add(c1, c2);      struct complex subrtaction = sub(c1, c2);      cout << "\n\n\n// BY Reference\n";      cout << "Addition = " << additoin.real << " + " << additoin.imaginary << "i" << endl;      cout << "\n\n\n//by value\n";      cout << "Subtraction = " << subrtaction.real << " + " << subrtaction.imaginary << "i" << endl;      cout << "\n\n\n\nBY pointer \n";      // we can also use sperate pointer to store the address of c1 and c2.      divid(&c1, &c2);      return 0;  } |

**Output:**

****