Assignment 4

DDL: 23:59, Dec 10, 2023

Note: Late homework assignments will not be accepted, unless you have a valid written excuse (medical, etc.). You must do this assignment alone. No team work or "talking with your friends" will be accepted. No copying from the Internet. Cheating means zero. Here are a few extra instructions: Give meaningful names to your variables so we can easily know what each variable is used for in your program. Put comments in your code (in English!) to explain WHAT your code is doing and also to explain HOW your program is doing it. Make sure all your code is properly indented (formatted). Your code should be beautiful to read.

Part 1: Programming

Write programs in C++ to complete the following tasks:

Task 1

Practice basic declaration of pointer. Please complete the following task:

```
int main()
{
std::cout << "--- Task 1 ---\n";
// start task 1 here
std::cout << "Practice the basic declaration of pointer :\n";
int x = 10, y = 20;
int *z = _________________; // FIXME: the pointer points to the address of
x
std::cout << "x stores the value is " << ____ << "\n";
std::cout << "y stores the value is " << ____ << "\n";
std::cout << "z stores the value is " << ____ << "\n";
std::cout << "The address of x is " << ____ << "\n";
std::cout << "The address of y is " << ____ << "\n";
std::cout << "The address of z is " << ____ << "\n\n";
std::cout << "The address of x is the same as the value z stores:" << ____
<<
"\n\n"; // FIXME: here should be the value z stores
std::cout << "Use * dereference operator to access the value at the
address of
x : " << ____ << "\n"; // FIXME: dereference the pointer z

return 0;
}
```

Note: It is fine if the value of address is not the same as author's. Here is the author's sample output:

```
--- Task 1 ---
Practice the basic declaration of pointer :
```

```
x stores the value is 10
y stores the value is 20
z stores the value is 0x61fe4c
The address of x is 0x61fe4c
The address of y is 0x61fe
The address of z is 0x61fe
The address of x is the same as the value z stores:0x61fe4c
Use * dereference operator to access the value at the address of x : 10
```

Task 2

Practice pointer to structure. Please define a struct named Staff which contains four variables inside of it: an interger named id, an int named age, a double named wage, and a string named name. Create a Staff struct (staff) and assign values to id, age, wage and name respectively. Declare a Staff pointer (pt) and point to the staff and output the staff information through this pointer. Here is the author's sample output:

```
--- Task 2 ---
Practice pointer to structure.
1) define a struct named Staff with 4 variable id, age, wage, name
2) delcare a Staff struct variable named staff
3) delcare a Staff pointer named pt and point to staff
4) Display the staff information through this pointer (pt)
ID: 18
Age: 20
Wage: 108.
Name: Alita
```

Task 3

Sort an integer array ({6, 2, 8, 1, 7, 10, 5}) using pointer in ascending order.

Note: for this task, you can use iostream library only, other libraries are not allowed. Here is the author's sample output:

```
--- Task 3 ---
Sort an integer array using pointer in ascending order.
Original array: 6 2 8 1 7 10 5
After sorted array: 1 2 5 6 7 8 10
```

Task 4

Write a function named printDigit accept an integer to print out each digit in reversely. Note, required to use recursive function. Here is the author's sample output:

```
--- Task 4 ---
Write a function named `printDigit` to print out each digit in reversely.
```

```
Please input an integer: 12358902
20985321
```

Task 5

Write a program that allows user perform arithmetic operations on two numbers. Your program must be menu driven, allowing the user to select the operation (+, -, *, or /) and input the numbers. Furthermore, your program must consist of following functions:

1. Function showMenu: This function shows the options to the user and explains how to enter data.
2. Function add: This function accepts two number as arguments and returns sum.
3. Function subtract: This function accepts two number as arguments and returns their difference.
4. Function mulitiply: This function accepts two number as arguments and returns product.
5. Function divide: This function accepts two number as arguments and returns quotient. Here is the author's sample output:

```
--- Task 5 ---
Write a program that allows user perform arithmetic operations on two
numbers.
MENU
1: Add
2: Subtract
3: Multiply
4: Divide
5: Exit
```

```
Enter your choice : 1
Please input two numbers: 2 4.
Sum: 6.
----------------------------
MENU
1: Add
2: Subtract
3: Multiply
4: Divide
5: Exit
Enter your choice : 4
Please input two numbers: 3 2
Quotiet: 1.
----------------------------
MENU
1: Add
2: Subtract
3: Multiply
4: Divide
5: Exit
Enter your choice : 8
Hey, invalid input!
```

```
----------------------------
MENU
1: Add
2: Subtract
3: Multiply
4: Divide
5: Exit
Enter your choice : 5
Bye!
```

Task 6

Swap two values by function call with diferent parameters' types.

1. passing by value
2. passing by reference
3. passing by pointer

Please complete:

```
/*
* 1) passing by value
* 2) passing by reference
* 3) passing by pointer
*/
```

Here is the author's sample output:

```
--- Task 6 ---
Swap two values by function call with diferent parameters' types.
Before calling swap_pass_by_value(), a is 10, b is 20
Inside body of swap_pass_by_value(), x is 20, y is 10
After calling swap_pass_by_value(), a is 10, b is 20
------------------------------------------------------------
Before calling swap_pass_by_reference(), a is 10, b is 20
Inside body of swap_pass_by_reference(), x is 20, y is 10
After calling swap_pass_by_reference(), a is 20, b is 10
------------------------------------------------------------
Before calling swap_pass_by_pointer(), a is 10, b is 20
Inside body of swap_pass_by_pointer(), x is 20, y is 10
After calling swap_pass_by_pointer(), a is 20, b is 10
------------------------------------------------------------
```

Part-2 Assignment4_Part2_0123456789.docx

Submission

All the tasks you should complete them into **header.h** file. The **main.cpp** file is used to call the functions to check if success or not. And compress these files into a zip file and rename it Assignment4_q0123456789.zip and upload to iSpace before deadline. (replace q0123456789 with your student ID number) Files list in the zip:

```
header.h
main.cpp
Assignment4_Part2_0123456789.docx or pdf or png
```

Deadline will be announced on iSpace.