Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 9

з навчальної дисципліни

“Базові методології та технології програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки та програмного забезпечення Доренський О. П.

ВИКОНАВ

студент академічної групи КБ-

23

Литвин М. В.

ПЕРЕВІРИВ

викладач кафедри кібербезпеки та програмного забезпечення Дрєєва Г.М.

Кропивницький – 2024

**Мета:** полягає у набутті ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів С/С++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

**ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ**

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

**Варіант 4**

**Алгоритм 9.1**

1.    Відображення інформації про розробника.

2.    Введення користувачем суми покупки.

3.    Розрахунок відсотка знижки та суми до сплати з урахуванням цієї знижки згідно з умовами:

o   Якщо сума покупки менше або дорівнює 100, знижка становить 5%.

o   Якщо сума покупки більше 100 і менше або дорівнює 200, знижка становить 7%.

o   Якщо сума покупки більше 200 і менше або дорівнює 300, знижка становить 9%.

o   Якщо сума покупки більше 300 і менше або дорівнює 400, знижка становить 12%.

o   Якщо сума покупки більше 400 і менше або дорівнює 500, знижка становить 15%.

o   Якщо сума покупки більше 500 і менше або дорівнює 1000, знижка становить 20%.

o   Якщо сума покупки більше 1000, знижка становить 75%.

4.    Виведення результату на екран у вигляді відсотка знижки та суми до сплати з урахуванням цієї знижки.

**Лістинг коду програми 9.1 з коментарями**

#include <iostream> // підключення бібліотеки введення та виведення на консоль

#include <iomanip> // підключення бібліотеки для форматованого виведення

void SoftwareDeveloper(){ // функція виводу інформації про розробника

setlocale(LC\_ALL, "ukr"); // встановлення локалізації на українську

std::cout << "\n -------------------------------------\n"

<< "| Lytvyn Maksym, CUNTU, opd@kntu.kr.ua |\n" << "| Литвин Максим, ЦНТУ, opd@kntu.kr.ua |\n" << "\n ------ (c) All Rights Reserved ------\n\n"; // виведення інформації про розробника

}

void calculateDiscount(double purchaseAmount) { // функція розрахунку знижки

double discountRate = 0.0; // змінна для відсотка знижки

double discountAmount = 0.0; // змінна для суми знижки

if (purchaseAmount <= 100) { // перевірка умови для розрахунку знижки

discountRate = 0.05; // встановлення відсотка знижки

} else if (purchaseAmount <= 200) { // перевірка іншої умови для розрахунку знижки

discountRate = 0.07; // встановлення відсотка знижки

} else if (purchaseAmount <= 300) { // перевірка іншої умови для розрахунку знижки

discountRate = 0.09; // встановлення відсотка знижки

} else if (purchaseAmount <= 400) { // перевірка іншої умови для розрахунку знижки

discountRate = 0.12; // встановлення відсотка знижки

} else if (purchaseAmount <= 500) { // перевірка іншої умови для розрахунку знижки

discountRate = 0.15; // встановлення відсотка знижки

} else if (purchaseAmount <= 1000) { // перевірка іншої умови для розрахунку знижки

discountRate = 0.20; // встановлення відсотка знижки

} else { // в разі, якщо жодна з умов не виконалась

discountRate = 0.75; // встановлення відсотка знижки

}

discountAmount = purchaseAmount \* discountRate; // розрахунок суми знижки

double totalAmount = purchaseAmount - discountAmount; // розрахунок суми до сплати з урахуванням знижки

std::cout << "Знижка: " << std::fixed << std::setprecision(2) << (discountRate \* 100) << "% (" << discountAmount << " грн.)" << std::endl; // виведення інформації про знижку

std::cout << "Сума до сплати з урахуванням знижки: " << totalAmount << " грн." << std::endl; // виведення суми до сплати з урахуванням знижки

}

int main() { // точка входу в програму

SoftwareDeveloper(); // виведення інформації про розробника

double purchaseAmount; // змінна для суми покупки

std::cout << "Введіть суму покупки: "; // виведення запиту на введення суми покупки

std::cin >> purchaseAmount; // зчитування суми покупки з консолі

calculateDiscount(purchaseAmount); // виклик функції розрахунку знижки

return 0; // вихід з програми

}

**Алгоритм 9.2**

1. Відображення інформації про розробника.
2. Введення користувачем розміру чоловічої спідні за українською системою.
3. Створення об'єктів класів конвертерів розмірів для міжнародної, чеської, британської, німецької та французької систем розмірів.
4. Виклик методу getSize для кожного конвертера та виведення результатів конвертації розміру в інші системи.

**Лістинг коду програми 9.2 з коментарями**

#include <iostream> // Підключення бібліотеки для введення/виведення

#include <map> // Підключення бібліотеки для використання мап

#include <string> // Підключення бібліотеки для роботи з рядками

void SoftwareDeveloper() { // Функція для виведення інформації про розробника

setlocale(LC\_ALL, "ukr"); // Встановлення локалізації для виводу українською

std::cout << "\n -------------------------------------\n"

<< "| Lytvyn Maksym, CUNTU, opd@kntu.kr.ua |\n" << "| Литвин Максим, ЦНТУ, opd@kntu.kr.ua |\n" << "\n ------ (c) All Rights Reserved ------\n\n"; // Виведення інформації про розробника

}

class SizeConverter { // Базовий клас конвертера розмірів

public:

virtual std::string getSize(int ukraineSize) const = 0; // Віртуальний метод для отримання розміру

};

class InternationalSizeConverter : public SizeConverter { // Клас для конвертації в міжнародний розмір

public:

std::string getSize(int ukraineSize) const override { // Реалізація методу конвертації

if (ukraineSize == 0) return "S"; // Перетворення розміру відповідно до таблиці

if (ukraineSize == 1) return "M";

if (ukraineSize == 2) return "L";

if (ukraineSize == 3) return "XL";

if (ukraineSize == 4) return "XXL";

if (ukraineSize == 5) return "XXXL";

return "Розмір не визначено"; // Повернення рядка, якщо розмір не знайдено

}

};

// Аналогічно реалізовано класи CzechSizeConverter, UKSizeConverter, GermanySizeConverter, FranceSizeConverter

int main() { // Головна функція програми

SoftwareDeveloper(); // Виведення інформації про розробника

std::cout << "Введіть розмір чоловічої спідні за українською системою: "; // Виведення запиту на введення розміру

int ukraineSize; // Змінна для зберігання введеного розміру

std::cin >> ukraineSize; // Зчитування розміру з консолі

// Створення об'єктів класів конвертерів розмірів

InternationalSizeConverter internationalConverter;

CzechSizeConverter czechConverter;

UKSizeConverter ukConverter;

GermanySizeConverter germanyConverter;

FranceSizeConverter franceConverter;

// Виведення результатів конвертації розміру в інші системи

std::cout << "Міжнародний: " << internationalConverter.getSize(ukraineSize) << std::endl;

std::cout << "Чехія, Словаччина: " << czechConverter.getSize(ukraineSize) << std::endl;

std::cout << "Великобританія: " << ukConverter.getSize(ukraineSize) << std::endl;

std::cout << "Німеччина: " << germanyConverter.getSize(ukraineSize) << std::endl;

std::cout << "Франція: " << franceConverter.getSize(ukraineSize) << std::endl;

return 0; // Повернення 0, що означає успішне завершення програми

}

**Алгоритм 9.3**

1. Відображення інформації про розробника.
2. Введення користувачем натурального числа N від 0 до 190550.
3. Перевірка коректності введеного числа.
4. Обчислення кількості нулів або одиниць у бінарному представленні числа.
5. Виведення результату.

**Лістинг коду програми 9.3 з коментарями**

#include <iostream> // Підключення бібліотеки для введення/виведення

void SoftwareDeveloper(){ // Функція для виведення інформації про розробника

setlocale(LC\_ALL, "ukr"); // Встановлення локалізації для виводу українською

std::cout << "\n -------------------------------------\n" // Виведення декоративної рамки

<< "| Lytvyn Maksym, CUNTU, opd@kntu.kr.ua |\n" << "| Литвин Максим, ЦНТУ, opd@kntu.kr.ua |\n" << "\n ------ (c) All Rights Reserved ------\n\n"; // Виведення інформації про розробника

}

int countBits(int n) { // Функція для підрахунку кількості нулів та одиниць у бінарному представленні числа

int countZeroes = 0; // Лічильник нулів

int countOnes = 0; // Лічильник одиниць

while (n) { // Поки число не стане 0

if (n & 1) { // Якщо останній біт числа - одиниця

countOnes++; // Збільшуємо лічильник одиниць

} else { // Якщо останній біт числа - нуль

countZeroes++; // Збільшуємо лічильник нулів

}

n >>= 1; // Зсув числа вправо на один біт (еквівалент ділення на 2)

}

return n ? countZeroes : countOnes; // Повертаємо кількість нулів, якщо число не дорівнює 0, інакше - кількість одиниць

}

int main() { // Головна функція програми

SoftwareDeveloper(); // Виведення інформації про розробника

int N; // Змінна для зберігання введеного числа

std::cout << "Введіть натуральне число N від 0 до 190550: "; // Виведення запиту на введення числа

std::cin >> N; // Зчитування числа з консолі

if (N < 0 || N > 190550) { // Перевірка коректності введеного числа

std::cout << "Некоректне значення N.\n"; // Виведення повідомлення про помилку

return 1; // Повернення 1, що означає помилку

}

int result = countBits(N); // Обчислення кількості нулів або одиниць у бінарному представленні числа

std::cout << "Результат: " << result << "\n"; // Виведення результату

return 0; // Повернення 0, що означає успішне завершення програми

}

**Алгоритм 9.4**

1. Відображення інформації про розробника.
2. Цикл, що виконується доти, доки користувач не вибере вихід.
3. Зчитування символу від користувача.
4. Виклик відповідної функції в залежності від введеного символу.
5. Запит на продовження роботи або вибір вихідної дії.

**Лістинг коду програми 9.4 з коментарями**

#include <iostream> // Підключення бібліотеки для введення/виведення

#include <conio.h> // Підключення бібліотеки для роботи з консоллю

void SoftwareDeveloper(){ // Функція для виведення інформації про розробника

setlocale(LC\_ALL, "ukr"); // Встановлення локалізації для виведення українською

std::cout << "\n -------------------------------------\n"

<< "| Lytvyn Maksym, CUNTU, opd@kntu.kr.ua |\n" << "| Литвин Максим, ЦНТУ, opd@kntu.kr.ua |\n" << "\n ------ (c) All Rights Reserved ------\n\n"; // Виведення інформації про розробника

}

void task\_9\_1() { // Функція для виконання задачі 9.1

std::cout << "Виклик функції задачі 9.1\n";

}

void task\_9\_2() { // Функція для виконання задачі 9.2

std::cout << "Виклик функції задачі 9.2\n";

}

void task\_9\_3() { // Функція для виконання задачі 9.3

std::cout << "Виклик функції задачі 9.3\n";

}

void calculation() { // Функція для виконання обчислень

std::cout << "Виклик функції calculation\n";

}

int main() { // Головна функція програми

SoftwareDeveloper(); // Виклик функції для виведення інформації про розробника

char choice; // Змінна для зберігання вибору користувача

char repeat = 'y'; // Змінна для повторення циклу

do {

std::cout << "Введіть символ (з, !, д, h): "; // Виведення запиту на введення символу

choice = \_getch(); // Отримання символу з консолі

std::cout << choice << std::endl; // Виведення введеного символу

switch (choice) { // Перевірка введеного символу

case 'з': // Виклик функції обчислення

calculation();

break;

case '!': // Виклик функції задачі 9.1

task\_9\_1();

break;

case 'д': // Виклик функції задачі 9.2

task\_9\_2();

break;

case 'h': // Виклик функції задачі 9.3

task\_9\_3();

break;

default: // Виведення звукового сигналу при некоректному введенні

std::cout << "\a";

break;

}

std::cout << "Бажаєте продовжити (у або j для виходу): "; // Запит на продовження виконання

std::cin >> repeat; // Зчитування вибору користувача

} while (repeat != 'у' && repeat != 'j'); // Повторення циклу, поки користувач не вибере вихід

return 0; // Повернення успішного завершення програми

}

**Висновок:** В ході виконання лабораторної роботи я більше попрактикувався з використанням функцій і створення бібліотеки модулів, що дозволить ефективніше вирішувати поставлені задачі для створення програми. Бібліотеки модулів дають змогу винести функції, які використовує програма у окремий файл, таким чином різні люди можуть додавати елементи програми, як функції.

Завдання різного виду складності, що дозволило отримати більше корисних навичок. Було досліджено процес розгалуження, що дозволяє робити правильні дії через значення, які воодить користувач. Сподобалось використання множинного вибору switch case, що дозволяє не використовувати стандартну конструкцію if elseif else, що може бути корисним, коли ми знаємо, які значення має ввести користувач. Крім цього отримав більш чітке уявлення про пам’ять комп’ютера. Розібрати її роботу для вирішення задачі, що дозволить більш економно розподіляти ресурси комп’ютера під час створення програми. Більше детально розглянув байти та біти на чому і грунтувалось одне з завдань, це достатньо сильно розширило мої знання з цієї теми. Також були задіяні різні типи даних для конкретних завдань

Я попрактикувався в створенні файлів статичних бібліотек та заголовкового файлу в який збираються прототипи функцій, які використовуються в основній програмі. Отримані навички стануть корисними у подальшому розумінні роботи програмістів у команді.