

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Казанский национальный исследовательский технический университет
им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации
Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №2
по дисциплине

Основы алгоритмизации и программирования

Тема: «Программирование алгоритмов разветвляющейся структуры»

Работу выполнил
Студент гр.4235
Желваков А.С.

Принял
Преподаватель Шмидт И.Р.

Казань 2023

ВАРИАНТ 9

1) Цель работы

Приобрести умения и практические навыки для программирования алгоритмов разветвляющейся структуры.

2) Задание на лабораторную работу –

- Даны действительные числа a , b , c . удвоить эти числа, если $a \geq b \geq c$, и заменить их абсолютными значениями (значение по модулю), если это не так.
- Разработать простой тест: на экране по очереди появляются вопросы (вопросы выбираются студентом), с вариантами ответов. В конце работы программа выдает количество заработанных баллов по результатам ответа. Минимальное количество вопросов – пять.

3) Результат выполнения работы

- Задание 1.

Код представлен в листинге 1.1

1. Декларируем переменные a , b , c с типом `int`.
2. Просим юзера ввести каждую переменную.
3. Проверяем условие, что $a \geq b \geq c$. Если оно истинно - умножаем переменные на два, если нет - приводим к абсолютному числу.
4. Выводим результат.

- Задание 2.

Код представлен в листинге 1.2

1. Прописываем класс `User`. Указываем инициализируемые переменные и пишем методы.
2. Прописываем класс `Question`. Указываем инициализируемые переменные и пишем методы.
3. В основном теле программы декларируем все необходимые переменные.
4. Запрашиваем имя, создаем объект класса `User`.
5. Прописываем все вопросы. После каждого вопроса запрашиваем ответ и проверяем его. Если он правильный - начисляем юзеру 5 баллов.

6. Проверяем, получил ли юзер более 20 баллов. Если получил - поздравляем с победой. Если нет - показываем ему это.

4) Ответы на контрольные вопросы.

Вопрос: Что такое тип данных?

Ответ: Тип данных - это спецификация, определяющая, какие виды данных могут быть хранены и обработаны в программе. Типы данных для хранения числовых данных включают целочисленные типы (например, int, long), типы с плавающей запятой (например, float, double) и другие.

Вопрос: Какие условные операторы используются при реализации алгоритмов разветвляющейся структуры?

Ответ: Условные операторы, используемые при реализации алгоритмов разветвляющейся структуры, включают в себя if, else if и else. Они позволяют выполнять различные блоки кода в зависимости от выполнения условий.

Вопрос: Какие операторы используются при создании сложных логических выражений?

Ответ: Для создания сложных логических выражений используются логические операторы, такие как логическое И (&&), логическое ИЛИ (||) и логическое НЕ (!). Эти операторы позволяют комбинировать более простые логические условия для получения более сложных выражений.

5) Листинг программы.

1.1

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int a, b, c;
```

```

cout << "Enter the number a: ";
cin >> a;

cout << "Enter the number b: ";
cin >> b;

cout << "Enter the number c: ";
cin >> c;

if (a >= b && b >= c) {
    a *= 2;
    b *= 2;
    c *= 2;
} else {
    a = abs(a);
    b = abs(b);
    c = abs(c);
}

cout << "a = " << a << " " << "b = " << b << " " << "c = " << c;

return 0;
}

```

1.2

```

#include <iostream>
#include <limits>
#include <map>

```

```
using namespace std;
```

```
class User {
```

```
public:
```

```
    User(string name) {  
        self_name = name;  
        scores = 0;  
    }
```

```
    int getScores() {  
        return scores;  
    };
```

```
    void upScores(int score) {  
        scores += score;  
    };
```

```
private:
```

```
    string self_name;  
    int scores;  
};
```

```
class Question {
```

```
public:
```

```
    Question(  
        string title,  
        std::map<int, string> answers,  
        int correct_answer
```

```

    ) {
    self_title = title;
    self_answers = answers;
    self_correct_answer = answers[correct_answer];
}

```

```

void displayData() {
    cout << "Question: " << self_title << endl;
    cout << "Answers:" << endl;
    for (int i = 1; i <= self_answers.size(); i++) {
        cout << i << " - " << self_answers[i] << endl;
    };
};

```

```

bool checkAnswer(int answer) {
    bool result = (self_answers[answer] == self_correct_answer);

    if (result) {
        cout << "Correct!" << endl << endl;
    } else {
        cout << "Incorrect. Correct answer is " << self_correct_answer << endl <<
endl;
    };
    return result;
};

```

```

private:
    string self_title;
    std::map<int, string> self_answers;
    string self_correct_answer;

```

```
};
```

```
int getPositiveIntegerInput(std::map<int, string> answers) {
    int input;
    while (true) {
        cout << "Answer number...: ";
        cin >> input;

        if (answers.count(input) == 0 || cin.fail()) {
            cin.clear();
            cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
            cout << "There is no such answer" << endl << endl;
        } else {
            break;
        }
    };
    return input;
}
```

```
int main() {
    string name;
    int answer1, answer2, answer3, answer4, answer5;
    int score_constant = 5;

    cout << "What's your name? : ";
    cin >> name;

    User user(name);
```

```
cout << "Hello, " << name << endl << "Your initial scores are as follows " <<
user.getScores() << endl;
```

```
cout << "Each question gives 5 points. If you score 20, you win" << endl <<
endl;
```

```
std::map<int, string> answers1 = {{1, "2 floors"}, {2, "3 floors"}, {3, "4
floors"}};
```

```
Question question1("How many floors are there in the 7th KAI building?",
answers1, 3);
```

```
question1.displayData();
```

```
answer1 = getPositiveIntegerInput(answers1);
```

```
if (question1.checkAnswer(answer1)) {
```

```
    user.upScores(score_constant);
```

```
};
```

```
std::map<int, string> answers2 = {{1, "Berlin"}, {2, "London"}, {3, "Paris"}};
```

```
Question question2("What is the capital of France?", answers2, 3);
```

```
question2.displayData();
```

```
answer2 = getPositiveIntegerInput(answers2);
```

```
if (question2.checkAnswer(answer2)) {
```

```
    user.upScores(score_constant);
```

```
};
```

```
std::map<int, string> answers3 = {{1, "Venus"}, {2, "Mars"}, {3, "Jupiter"}};
```

```
Question question3("Which planet is known as the 'Red Planet'?", answers3, 2);
```

```
question3.displayData();
```



```
answer3 = getPositiveIntegerInput(answers3);
```

```
if (question3.checkAnswer(answer3)) {
    user.upScores(score_constant);
};
```

```
std::map<int, string> answers4 = {{1, "Charles Dickens"}, {2, "William
Shakespeare"}, {3, "Jane Austen"}};
```

```
Question question4("Who wrote the play 'Romeo and Juliet'?", answers4, 2);
question4.displayData();
```

```
answer4 = getPositiveIntegerInput(answers4);
```

```
if (question4.checkAnswer(answer4)) {
    user.upScores(score_constant);
};
```

```
std::map<int, string> answers5 = {{1, "African Elephant"}, {2, "Blue Whale"},
{3, "Giraffe"}};
```

```
Question question5("What is the largest mammal in the world?", answers5, 2);
question5.displayData();
```

```
answer5 = getPositiveIntegerInput(answers5);
```

```
if (question5.checkAnswer(answer5)) {
    user.upScores(score_constant);
};
```

```
if (user.getScores() >= 20) {
```

```

    cout << "Congratulations, " << name << "! You won the game! " << "You
scored " << user.getScores() << " point!" << endl;
} else {
    cout << "You couldn't score the required points" << endl;
};

return 0;
}

```

ПРИЛОЖЕНИЕ 2. ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ ОТЧЕТА

| Объект | Характеристики |
|-----------------|---|
| Лист документа | 1) Ориентация – книжная. 2) Поля документа левое – 3 см; правое – 1 см; верх – 1 см; низ – 1 см; 3) Нумерация страниц – внизу, по центру, особый колонтитул для первой страницы.. |
| Абзац | 1) Междустрочный - 1,5 (полуторный) 2) Отступ первой строки – 1,25 |
| Текст документа | 1) Шрифт - Times New Roman 2) Размер – 14 3) Выравнивание – по ширине |
| Текст заголовка | 1) Шрифт - Times New Roman |

| | |
|-----------------|--|
| | <p>2) Размер – 16</p> <p>3) Начертание – полужирный</p> <p>4) Размещение – по центру</p> |
| Рисунки | <p>1) Размещение – по центру</p> <p>2) Все рисунки нумеруются по порядку.</p> <p>3) Перед рисунком в тексте на него должна быть ссылка (как в методичке)</p> |
| Подпись рисунка | <p>1) Размещение – по центру</p> <p>2) Шрифт - Times New Roman</p> <p>3) Размер -12 Сначала пишется фраза «Рисунок 4», а через тире с заглавной буквы название рисунка (как в методичке)</p> |