Белорусский государственный технологически университет

Факультет информационных технологий

Кафедра программной инженерии

Лабораторная работа 6

По дисциплине «Основы программной инженерии»

На тему «Модульное программирование»

Выполнил:

Студент 1 курса 8 группы

Стахейко Антон Алексеевич

Преподаватель: Ольга Александровна

3.3) Входные данные:

- Пользователь должен выбрать один из вариантов использования (1, 2, 3 или 4) путем ввода соответствующего числа.

- Для каждого выбранного варианта, пользователь должен ввести количество символов и сами символы, которые будут использоваться для выполнения соответствующих операций.

Выходные данные:

- Вывод результатов каждой операции в зависимости от выбранного варианта использования.

- В случае ошибки (некорректный ввод символов или выбор некорректного варианта использования), выводится сообщение об ошибке.

5)

Состав кода:

- Код написан на языке программирования C++.

- Необходимые библиотеки: iostream, Windows.h.

- Созданы функции для каждого из четырех вариантов использования: delta\_eng (), delta\_rus (), code\_of\_sumbol(), и вспомогательные функции:

checking(), data\_input(), main().

- В модулях delta\_eng (), delta\_rus (), code\_of\_sumbol(), реализованы операции с символами, в зависимости от выбранного варианта.

- В модуле checking() проходит проверка входных данных на корректность.

- В модуле data\_imput() происходит запись всех данных, для последующей работы с ними.

- В модуле main() осуществляется вход в программу и настройка для работы с Русским языком так же происходит построение диалогового окна с пользователем, и выбор исполняемой функции.

- Для работы с русскими символами установлена кодировка Windows-1251 с помощью функций SetConsoleCP() и SetConsoleOutputCP().

Модуль delta\_eng (), предназначен для работы с символами латинского алфавита.

Входными данными является: один из символов, введенный пользователем.

Выходные данные - разница кодов между прописной и строчной буквой латинского алфавита.

Aлгоритм:

1)проверка передаваемых данных на корректность функцией checking(), если проверка успешна, перейти к пункту 2,иначе прейти к пункту 8.

2)если код символа лежит в диапазоне от 65 до 90 (включая),то прейти к пункту 3, иначе перейти к пункту 4.

3)присвоить значение переменным по формулам:

Big\_ch = (int)symbol,

small\_ch = (int)tolower(symbol), и перейти к пункту 5.

4) присвоить значение переменным по формулам:

Big\_ch = (int)toupper(symbol),

small\_ch = (int)symbol, и перейти к пункту 6.

5)Вывод:

“Верхний регистр - ” symbol “ = ” big\_ch,

“Нижний регистр - ”(char)tolower(symbol) “ = ” small\_ch,

И перейти к пункту 7.

6)Вывод:

“Верхний регистр -” (char)toupper(symbol), “ = ” big\_ch,

“Нижний регистр -” symbol “ = ” small\_ch,

И перейти к пункту 7.

7)Вернуть значение равное: small\_ch-big\_ch.

8)Вернуть значение равное: 0.

Модуль delta\_rus (), предназначен для работы с символами русского алфавита.

Входными данными является: один из символов, введенный пользователем.

Выходные данные - разница кодов между прописной и строчной буквой русского алфавита.

Aлгоритм:

1)проверка передаваемых данных на корректность функцией checking(), если проверка успешна, перейти к пункту 2,иначе прейти к пункту 8.

2)если код символа лежит в диапазоне от 192 до 223 (включая),то прейти к пункту 3, иначе перейти к пункту 4.

3)присвоить значение переменным по формулам:

Big\_ch = (int)symbol,

small\_ch = (int)tolower(symbol), и перейти к пункту 5.

4) присвоить значение переменным по формулам:

Big\_ch = (int)toupper(symbol),

small\_ch = (int)symbol, и перейти к пункту 6.

5)Вывод:

“Верхний регистр - ” symbol “ = ” big\_ch,

“Нижний регистр - ”(char)tolower(symbol) “ = ” small\_ch,

И перейти к пункту 7.

6)Вывод:

“Верхний регистр -” (char)toupper(symbol), “ = ” big\_ch,

“Нижний регистр -” symbol “ = ” small\_ch,

И перейти к пункту 7.

7)Вернуть значение равное: small\_ch-big\_ch.

8)Вернуть значение равное: 0.

Модуль code\_of\_sumbol(), предназначен для работы с цифрами.

Входными данными является: один из символов, введенный пользователем.

Выходные данные – код введенного символа.

Aлгоритм:

1. проверка передаваемых данных на корректность функцией checking(), если проверка успешна, перейти к пункту 2,иначе прейти к пункту 3.
2. Вернуть значение равное: (int)symbol.
3. Вернуть значение равное: 0.

Модуль checking (), предназначен для проверки введённых данных на корректность.

Входными данными является: один из символов, введенный пользователем, и id выполняемой функции.

Выходные данные – логическое значение.

Aлгоритм функции checking\_symbol():

1. Если id==’1’,то прейти к пункту 2, иначе перейти к пункту 5.
2. Если кода символа(symbol) лежит в диапазоне от 65 до 90 (включая),

или от 97 до 122 (включая) то перейти к пункту 3, иначе перейти к пункту 4.

1. Вернуть значение: true.
2. Вернуть значение: false.
3. Если id==’2’,то прейти к пункту 6, иначе перейти к пункту 9.
4. Если кода символа(symbol) лежит в диапазоне от 192 до 255(включая), то перейти к пункту 7, иначе перейти к пункту 8.
5. Вернуть значение: true.
6. Вернуть значение: false.
7. Если id==’3’,то прейти к пункту 10, иначе перейти к пункту 13.
8. Если кода символа(symbol) лежит в диапазоне от 48 до 57(включая), то перейти к пункту 11, иначе перейти к пункту 12.
9. Вернуть значение: true.
10. Вернуть значение: false.
11. Вывод: “Данные введены не корректно”, и перейти к пункту 14.
12. Вернуть значение: false.

Модуль main(), предназначен для работы с построения диалогового окна с пользователем и вызова функций.

Входными данными является: цифра выполняемого действия (1,2,3,4).

Aлгоритм:

1. 1)настройка на корректность отображения русских символов

SetConsoleCP(1251)

SetConsoleOutputCP(1251)

1. Выполнять пока(is\_active) начало цикла
2. Вывод:

“Выберите выполняемое действие”

“1- разница значении кодов ASCII в прописном и строчном формате”

“2 - разница значении кодов Windows-1251 в прописном и строчном формате Русского алфавита”

“3-вывести код символа, соответствующего введенной цифре”

“4-завершение работы”,

и перейти к пункту 4.

1. Ввод: id.
2. Если id==’1’ то перейти к пункту 6, иначе перейти к пункту 10.
3. Создание динамического массива: char\* symbols =data\_input(&count), и перейти к пункту 7.
4. Выполнять шаг 8, пока i<count, с шагом в один,a затем перейти к пункту 9:
5. Вывод: “разница =” delta\_eng(symbols[i]),перейти к пункту.
6. Очистка динамической памяти:

delete symbols.

Symbols= null.

Перейти к пункту 23.

1. Если id==’2’ то перейт к пункут 11, иаче перейти к пункту 15.
2. Создание динамического массива: char\* symbols =data\_input(&count), и перейти к пункту 11.
3. Выполнять шаг 13, пока i<count, с шагом в один,a затем перейти к пункту 14:
4. Вывод: “разница =” delta\_rus(symbols[i]),перейти к пункту.
5. Очистка динамической памяти:

delete symbols.

Symbols= null.

Перейти к пункту 23.

1. Если id==’3’ то перейти к пункту 16, иначе перейти к пункту 20.
2. Создание динамического массива char\* symbols =data\_input(&count), и перейти к пункту 17.
3. Выполнять шаг 18, пока i<count, с шагом в один,a затем перейти к пункту 19:
4. Вывод: “разница =” code\_of\_symbol(symbols[i]),перейти к пункту.
5. Очистка динамической памяти:

delete symbols.

Symbols= null.

Перейти к пункту 23.

1. Если id==’4’, то перейти к пункту 21, иначе перейти к пункту 22.
2. Is\_active=false.
3. Вывод: “такого варианта не существует”, перейти к пункту 23.
4. Конец цикла
5. Вернуть значение 0

Модуль data\_inputl(), предназначен записи введенной пользователем информации.

Входными данными является: ссылка на счетчик количества элементов, сами символы.

Выходные данные – ссылка на область памяти, где лежит массив с данными.

Aлгоритм:

1. Вывод: ”Количество символов: ”
2. Ввод \*amount\_of\_char.
3. Выделение динамической памяти char\* symbols = new char[\*amount\_of\_char]
4. Вывод: “Введите символы”
5. Выполнять пункт 6 пока i=0, i<\*amount\_of\_char, затем перейти к пункту 6.
6. Вернуть ссылку на созданный массив: symbols

6)

Главный модуль

Main()

Модуль data\_input()

Модуль delta\_rus()

Модуль code\_of\_symbol()

Модуль delta\_eng()

Модуль checking()

Псевдокод модуля delta\_eng():

Подключение заголовка

НАЧАЛО

ЕСЛИ(checking\_symbol(symol, 1)ТО

ЕСЛИ((int)symbol>=65 И (int)symbol <=90) ТО

Big\_ch = (int)symbol

Small\_ch =(int)tolower(symbol)

ВЫВОД(“Верхний регистр - ” symbol “ = ” big\_ch)

ВЫВОД(“Нижний регистр - ” (char)tolower(symbol) “ = ” small\_ch)

ИНАЧЕ

Big\_ch = (int)toupper(symbol)

Small\_ch =(int)symbol

ВЫВОД(“Верхний регистр - ” (char)toupper(symbol) “ = ” big\_ch)

ВЫВОД(“Нижний регистр - ”symbol“ = ” small\_ch)

ВЕРНУТЬ(small\_ch – big\_ch)

ИНАЧЕ

ВЕРНУТЬ(0)

КОНЕЦ

Псевдокод модуля delta\_rus():

Подключение заголовка

НАЧАЛО

ЕСЛИ(checking\_symbol(symol, 2)ТО

ЕСЛИ((int)symbol>=192 И (int)symbol <=223) ТО

Big\_ch = (int)symbol

Small\_ch =(int)tolower(symbol)

ВЫВОД(“Верхний регистр - ” symbol “ = ” big\_ch)

ВЫВОД(“Нижний регистр - ” (char)tolower(symbol) “ = ” small\_ch)

ИНАЧЕ

Big\_ch = (int)toupper(symbol)

Small\_ch =(int)symbol

ВЫВОД(“Верхний регистр - ” (char)toupper(symbol) “ = ” big\_ch)

ВЫВОД(“Нижний регистр - ”symbol“ = ” small\_ch)

ВЕРНУТЬ(small\_ch – big\_ch)

ИНАЧЕ

ВЕРНУТЬ(0)

КОНЕЦ

Псевдокод модуля code\_of\_symbol():

Подключение заголовка

НАЧАЛО

ЕСЛИ(checking\_symbol(symbol,3)) ТО

ВЕРНУТЬ((int)symbol)

ИНАЧЕ

ВЕРНУТЬ(0)

КОНЕЦ

Псевдокод модуля checking\_symbol():

Подключение заголовка

НАЧАЛО

ЕСЛИ(id = 1) ТО

ЕСЛИ(((int)symbol>=65 И (int)symbol<=90) ИЛИ ((int)symbol>=65 И (int)symbol<=90))ТО

ВЕРНУТЬ(true)

ИНАЧЕ

ВЫВОД(“Данные введены не коректно”)

ВЕРНУТЬ(false)

ИНАЧЕ ЕСЛИ(id=2)ТО

ЕСЛИ((int)symbol>=192 И (int)symbol<=255)ТО

ВЕРНУТЬ(true)

ИНАЧЕ

ВЫВОД(“Данные введены не коректно”)

ВЕРНУТЬ(false)

ИНАЧЕ ЕСЛИ(id=3)ТО

ЕСЛИ((int)symbol>=48 И (int)symbol<=57)ТО

ВЕРНУТЬ(true)

ИНАЧЕ

ВЫВОД(“Данные введены не коректно”)

ВЕРНУТЬ(false)

ИНАЧЕ

ВЫВОД(“Данные введены не коректно”)

ВЕРНУТЬ(false)

КОНЕЦ

Псевдокод модуля data\_input():

Подключение заголовка

НАЧАЛО

ВЫВОД(“Количество символов: ”)

ВВОД(\*amount\_of\_char)

Char\* symbols = ВЫДЕЛИТЬ ОБЛАСТЬ ПАМЯТИ((\*amount\_of\_char)

ВЫВОД(“Введите символы”)

ВЫПОЛНЯТЬ ПОКА(int i =0, i<\*amount\_of\_char, i++) НАЧАЛО ЦИКЛА

ВВОДА(symbols[i])

КОНЕЦ ЦИКЛА

ВЕРНУТЬ(symbols)

КОНЕЦ

Псевдокод модуля main():

Подключение заголовка

НАЧАЛО

НАСТРОЙКА КОДИРОВКИ НА ВЫВОД(1251)

НАСТРОЙКА КОДИРОВКИ НА ВВОД(1251)

ВЫПОЛНЯТЬ ПОКА(is\_active)НАЧАЛО ЦИКЛА

ВЫВОД(“Выберите выполняемое действие”)

ВЫВОД(“1-разница значении кодов ASCII

2-разница значений кодов Windows-1251

3-вывести код символа

4-завершение работы”)

ВВОДА(id)

ЕСЛИ(id=1)ТО

Char\* symbols =data\_input(&count)

ВЫПОЛНЯТЬ ПОКА(int i =0, i<count, i++) НАЧАЛО ЦИКЛА

ВЫВОД(“Разница = ”delta\_eng(symbol[i]))

КОНЕЦ ЦИКЛА

УДАЛИТЬ symbols

Symbols=NULL

ИНАЧЕ ЕСЛИ(id=2) ТО

Char\* symbols =data\_input(&count)

ВЫПОЛНЯТЬ ПОКА(int i =0, i<count, i++) НАЧАЛО ЦИКЛА

ВЫВОД(“Разница = ”delta\_rus(symbol[i]))

КОНЕЦ ЦИКЛА

УДАЛИТЬ symbols

Symbols=NULL

ИНАЧЕ ЕСЛИ(id=3) ТО

Char\* symbols =data\_input(&count)

ВЫПОЛНЯТЬ ПОКА(int i =0, i<count, i++) НАЧАЛО ЦИКЛА

ВЫВОД(“код символа” symbols[i] ”= ”delta\_rus(symbol[i]))

КОНЕЦ ЦИКЛА

УДАЛИТЬ symbols

Symbols=NULL

ИНАЧЕ ЕСЛИ(id=3) ТО

ПРЕРВАТЬ

ИНАЧЕ

ВЫВОД(“Такого варианта не существует”)

КОНЕЦ ЦИКЛА

ВЕРНУТЬ(0)

КОНЕЦ