Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина Методы трансляции

ОТЧЕТ

к лабораторной работе № 1

на тему

**ОПРЕДЕЛЕНИЕ МОДЕЛИ ЯЗЫКА. ВЫБОР   
ИНСТРУМЕНТАЛЬНОЙ ЯЗЫКОВОЙ СРЕДЫ**

Выполнил             В. П. Бычко

Проверил                Н. Ю. Гриценко

Минск 2024

СОДЕРЖАНИЕ

[1 Постановка задачи 3](#_Toc160205801)

[2 Определение модели языка программирования С 4](#_Toc160205802)

[2.1 Типы данных в языке программирования C 4](#_Toc160205803)

[2.2 Операторы циклов и условные операторы 4](#_Toc160205804)

[2.3 Структуры 5](#_Toc160205805)

[3 Определение инструментальной языковой среды 7](#_Toc160205806)

[Выводы 8](#_Toc160205807)

[Приложение А (обязательное) Пример реализации программ на языке программирования C 9](#_Toc160205808)

## 1 ПОСТАНОВКА ЗАДАЧИ

Цель выполнения данной лабораторной работы – определить подмножество выбранного языка программирования, предоставить тексты двух или трех программ, включающих все элементы этого подмножества, а также определить инструментальную языковую среду, которая включает в себя язык программирования с указанием версии, на котором ведется разработка, операционная система, в которой выполняется разработка, и компьютер.

## 2 ОПРЕДЕЛЕНИЕ МОДЕЛИ ЯЗЫКА ПРОГРАММИРОВАНИЯ C

## 2.1 Типы данных в языке программирования C

В языке C имеются следующие типы данных:

1 char представляет один символ. Занимает в памяти 1 байт (8 бит). Может хранить любое значение из диапазона от -128 до 127.

2 int представляет целое число. В зависимости от архитектуры процессора может занимать 2 байта (16 бит) или 4 байта (32 бита). Если брать основные платформы - 64-разрядные Windows, Linux (вместе с Android) и MacOS, то размер int составляет 4 байта. Диапазон предельных значений соответственно также может варьироваться от –32768 до 32767 (при 2 байтах) или от −2 147 483 648 до 2 147 483 647 (при 4 байтах) и выше.

3 short представляет целое число в диапазоне от –32768 до 32767. Занимает в памяти 2 байта (16 бит). Имеет псевдонимы short int, signed short и signed short int.

4 long представляет целое число и занимает в памяти 4 байта (32 бита) или 8 байт (64 бита).

5 float представляет вещественное число одинарной точности с плавающей точкой в диапазоне +/- 3.4E-38 до 3.4E+38. В памяти занимает 4 байта (32 бита).

6 double представляет вещественное число двойной точности с плавающей точкой. В памяти занимает 8 байт (64 бита).

7 void представляет отсутствие типа.

## 2.2 Операторы циклов и условные операторы

Операторы циклов:

– Оператор цикла for. For в С используется для повторения блока кода определенное количество раз или до выполнения заданного условия. Инициализация выполняется перед началом цикла, условие проверяется перед каждой итерацией, пост-операция выполняется после каждой итерации;

– while проверяет некоторое условие, и если оно возвращает true, то выполняет блок кода;

– do -while выполняет один раз цикл, и если некоторое условие возвращает true, то продолжает выполнение цикла.

Условные операторы:

– Оператор if. Оператор ifиспользуется для выполнения блока кода, если заданное условие истинно. Синтаксис показан на рисунке 2.1

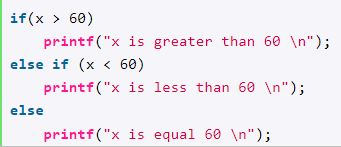


Рисунок 2.1 – Синтаксис оператора if

– switch/case позволяет обработать сразу несколько условий. После ключевого слова switch идет сравниваемое выражение. Это может быть переменная или константа. Значение этого выражения последовательно сравнивается со значениями, помещенными после оператора сase. И если совпадение будет найдено, то будет выполняться определенный блок сase. Синтаксис показан на рисунке 2.2.

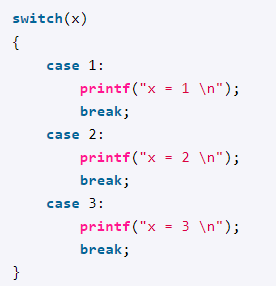


Рисунок 2.2 – Синтаксис оператора switch

## 2.3 Структуры

Теперь рассмотрим все структуры, которые присутствуют в языке C. Массивы в C представляют собой структуру данных, которая позволяет хранить набор элементов одного типа. Размер массива должен быть известен на этапе компиляции и не может изменяться во время выполнения программы.

Элементы массива нумеруются с помощью индексов, начиная с 0. Элементы массива хранятся последовательно в памяти компьютера, что обеспечивает быстрый доступ к ним.

Структуры в C позволяют объединять несколько переменных разных типов данных в одну составную переменную. Они могут содержать как простые типы данных, так и другие структуры, а также массивы. Структуры полезны для организации данных и создания пользовательских типов.

Структуры в C могут быть использованы для представления сложных данных, таких как информация о человеке (имя, возраст, адрес), автомобиле (марка, модель, год выпуска) и многое другое.

# 3 ОПРЕДЕЛЕНИЕ ИНСТРУМЕНТАЛЬНОЙ ЯЗЫКОВОЙ СРЕДЫ

В качестве языковой среды выбран язык программирования C# 12. Разработка основана на работе с операционной системой Windows на PC*.*

Инструментальная среда C# 12 может включать в себя мощную интегрированную среду разработки, которая предоставляет различные функции, такие как подсветка синтаксиса, автодополнение кода, интегрированный отладчик, возможность быстрого перехода к определению функций и классов, а также инструменты для рефакторинга кода. Некоторые популярные IDE для C# включают в себя Visual Studio, Project Rider.

C# является объектно-ориентированным и в этом плане много перенял у Java и С++. Например, C# поддерживает полиморфизм, наследование, перегрузку операторов, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений. И C# продолжает активно развиваться, и с каждой новой версией появляется все больше интересных функциональностей.

В данной главе была рассмотрена основная языковая среда, а также была выбрана платформа, на которой будет проводиться анализ.

## ВЫВОДЫ

В ходе работы было определено подмножества языка программирования (типы констант, переменных, операторов и функций). В подмножество языка были включены все типы. Были учтены все структуры, представленные в рассматриваемом языке. Операторы цикла (do...while, for, while). Условные операторы (if...else, case). Определена инструментальная языковая среда, т.е. язык программирования и операционная система для разработки. В ходе лабораторной работе дается полное определение подмножества языка программирования, тексты 2-3-х программ, включающих все элементы этого подмножества. Приводится подробное описание инструментальной языковой среды.

## ПРИЛОЖЕНИЕ А (обязательное) Пример реализации программ на языке программирования C

Листинг 1 – Пример программы на языке C

#include <stdio.h>

void printMessage() {

printf("Example of using different data types and arrays in C\n");

}

int main() {

char charVar = 'A';

int intArray[5] = {1, 2, 3, 4, 5};

short shortArray[3] = {10, 20, 30};

long longVar = 1234567890L;

float floatArray[4] = {1.1f, 2.2f, 3.3f, 4.4f};

double doubleArray[2] = {5.5, 6.6};

printMessage();

printf("charVar: %c\n", charVar);

printf("intArray: ");

for (int i = 0; i < 5; i++) {

printf("%d ", intArray[i]);

}

printf("\n");

printf("shortArray: ");

for (int i = 0; i < 3; i++) {

printf("%d ", shortArray[i]);

}

printf("\n");

printf("longVar: %ld\n", longVar);

printf("floatArray: ");

for (int i = 0; i < 4; i++) {

printf("%f ", floatArray[i]);

}

printf("\n");

printf("doubleArray: ");

for (int i = 0; i < 2; i++) {

printf("%lf ", doubleArray[i]);

}

printf("\n");

return 0;

}

Листинг 2 – Пример программы на языке C

#include <stdio.h>

int main() {

int whileCounter = 0;

while (whileCounter < 5) {

printf("while loop: %d\n", whileCounter);

whileCounter++;

}

for (int forCounter = 0; forCounter < 5; forCounter++) {

printf("For loop: %d\n", forCounter);

}

int doWhileCounter = 0;

do {

printf("do-while loop: %d\n", doWhileCounter);

doWhileCounter++;

} while (doWhileCounter < 5);

int num = 10;

if (num > 0) {

printf("The number is positive\n");

} else if (num < 0) {

printf("The number is negative\n");

} else {

printf("The number is zero\n");

}

char grade = 'B';

switch (grade) {

case 'A':

printf("Excellent!\n");

break;

case 'B':

printf("Good!\n");

break;

case 'C':

printf("Satisfactory\n");

break;

default:

printf("Unknown grade\n");

}

for (int i = 0; i < 10; i++) {

if (i == 5) {

break; // Exit the loop when i is 5

}

if (i % 2 == 0) {

continue; // Skip even values of i

}

printf("Current value of i: %d\n", i);

}

return 0;

}