

Лабораторная работа 2 (4 часа)

Конструирование программного обеспечения

Фундаментальные типы данных

Цель работы: ознакомиться с фундаментальными типами данных языка программирования C++ и их представлением в памяти компьютера.

1. Используйте при выполнении лабораторной работы материал лекции 2.
2. Значения **X**, **Y**, **Z**, **S**, используемые далее в тексте заданий, определяются по формулам: **X= 9+n**, **Y= 10+n**, **Z= 11+n**, **S= 1.0+n**, где **n** – порядковый номер студента по списку.
3. Создайте проект **SE_Lab02** (консольное приложение C++, архитектура X86). Все дальнейшие задания выполняйте в рамках этого проекта в конфигурации Debug в пошаговом режиме.
Каждая строка исходного кода **должна** сопровождаться комментариями с результатами выполнения заданий, полученными **вручную**.
4. Объявите 2 переменные типа **bool** и инициализируйте значениями **false** и **true**. В режиме отладки определите адреса памяти, в которых расположены значения этих переменных. Продемонстрируйте содержимое памяти.
5. Объявите переменную типа **char** и инициализируйте ее символом латинского алфавита (второй символ вашей фамилии на английском языке). С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти, назовите кодировку.
6. Объявите переменную типа **char** и инициализируйте ее символом русского алфавита (второй символ вашей фамилии на русском языке). С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти, назовите кодировку.
7. Объявите переменную типа **wchar_t** и инициализируйте ее символом латинского алфавита (первый символ вашей фамилии на английском языке). С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти, назовите кодировку.
8. Объявите переменную типа **wchar_t** и инициализируйте ее символом русского алфавита (первый символ вашей фамилии на русском языке). С помощью отладчика определите адрес памяти, в которой расположено

значение этой переменной. Продемонстрируйте содержимое памяти, назовите кодировку.

9. Объявите две переменные типа **short**. Инициализируйте их значениями **X** и **-X**.
 - а. *ручным способом* вычислите шестнадцатеричные значения **X** и **-X**.
 - б. с помощью отладчика определите адрес памяти, в которой расположены значения этих переменных. Продемонстрируйте содержимое памяти. Сравните результаты.
10. Запишите шестнадцатеричное значение самого большого числа типа **short**. Запишите шестнадцатеричное значение самого малого числа типа **short**. Присвойте эти значения двум переменным типа **short** и с помощью отладчика определите соответствующие десятичные значения.
11. Запишите шестнадцатеричное значение самого большого числа типа **unsigned short** (определите *вручную*). Запишите шестнадцатеричное значение самого малого числа типа **unsigned short** (определите *вручную*). Присвойте эти значения переменным типа **unsigned short** и с помощью отладчика определите соответствующие десятичные значения.
12. Объявите две переменные типа **int**. Инициализируйте их значениями **Y** и **-Y**. *Ручным способом* вычислите шестнадцатеричные значения **Y** и **-Y**. С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти и сравните с результатом, полученным вручную.
13. Запишите шестнадцатеричное значение самого большого числа типа **int**. Запишите шестнадцатеричное значение самого малого числа типа **int**. Присвойте эти значения переменным типа **int** и с помощью отладчика определите соответствующие десятичные значения.
14. Запишите шестнадцатеричное значение самого большого числа типа **unsigned int**. Запишите шестнадцатеричное значение самого малого числа типа **unsigned int**. Присвойте эти значения переменным типа **unsigned int** и с помощью отладчика определите соответствующие десятичные значения.
15. Объявите две переменные типа **long**. Инициализируйте их значениями **Z** и **-Z**. *Ручным способом* вычислите шестнадцатеричные значения **Z** и **-Z**. С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти.
16. Запишите шестнадцатеричное значение самого большого числа типа **long**. Запишите шестнадцатеричное значение самого малого числа типа

- long**. Присвойте эти значения переменным типа **long** и с помощью отладчика определите соответствующие десятичные значения.
17. Запишите шестнадцатеричные значение самого большого числа типа **unsigned long**. Запишите шестнадцатеричные значение самого малого числа типа **unsigned long**. Присвойте эти значения переменным типа **unsigned long** и с помощью отладчика определите соответствующие десятичные значения
 18. Объявите две переменные типа **float**. Инициализируйте их значениями **S** и **-S**. *Ручным способом* вычислите шестнадцатеричные значения **S** и **-S** (с точностью до трех знаков после точки). С помощью отладчика определите адрес памяти, в которой расположено значение этой переменной. Продемонстрируйте содержимое памяти и сравните с результатом, полученным вручную.
 19. Выполните операции, результатами которых будут значения: **1.#INF**, **-1.#INF**, **-1.#IND**.
 20. Объявите и проинициализируйте указатели на типы **char**, **wchar_t**, **short**, **int**, **float** и **double**. Просмотрите значения этих переменных с помощью отладчика и выпишите шестнадцатеричные представления значений. Увеличьте значения указателей на 3. Снова просмотрите эти значения, объясните полученный результат.
 21. Объявите и продемонстрируйте применение указателей на функции. Определите с помощью отладчика значения указателей. Дизассемблируйте код программы, определите, что находится в памяти по адресу равному значению этих указателей.

Ответьте на следующие вопросы:

- поясните понятие «фундаментальные типы данных»;
- перечислите все фундаментальные типы данных C++;
- с помощью какой встроенной функции можно определить размер данных любого типа?
- сколько отводится памяти для переменной типа **bool**?
- какие значения хранятся в памяти переменных типа **bool**?
- сколько отводится памяти для переменного типа **char**?
- сколько отводится памяти для переменного типа **wchar_t**?
- сколько отводится памяти для переменных типа **short**, **int**, **long**?
- сколько отводится памяти для переменной типа **unsigned short**, **unsigned int**, **unsigned long**?
- сколько отводится памяти для переменного типа **float**, **double**?

- все числа, записанные в форме десятиной дроби можно точно представить в форме **float** и **double**? Если нет, то почему?
- поясните значение **1.#INF** и **1.#IND** типов данных **float** и **double**;
- сколько отводится памяти для указателя?
- поясните назначение типа ***void**;
- сколько отводится памяти для указателя на функцию?
- сколько отводится памяти для ссылки?
- как определить знак числа типов **short**, **int** и **long** по их шестнадцатеричному представлению?
- как определить знак числа типов **float** и **double** по их шестнадцатеричному представлению?
- какой порядок байт используется для представления целочисленных данных?