3. Основы C++. Определение времени жизни и области видимости переменных. Пространство имен.

*Определение времени жизни и области видимости переменных*

Время жизни переменной определяется следующими правилами:

• Переменные, объявленные на внешнем уровне, всегда имеют глобальное время жизни

• Переменные, объявленные на внутреннем уровне, имеют локальное время жизни. При помощи static возможно обеспечить глобальное время жизни переменной

Видимость переменной в программе определяется по правилам:

• Переменные, объявленные или определенные на внешнем уровне, видимы от точки объявления или определения до конца исходного файла

• Переменные, объявленные или определенные внутри блока, видимы только внутри этого блока

*Модификаторы памяти*

• auto — автоматическая переменная. Память выделяется в стеке и при необходимости инициализируется каждый раз при выполнении оператора, содержащего ее определение. Освобождение памяти — при выходе из блока

• extern — переменная определяется в другом месте программы

• static — статическая переменная. Время жизни — постоянное. Инициализируется один раз при первом выполнении оператора, содержащего определение переменной. В зависимости от расположения оператора описания статические переменный могут быть глобальными и локальными

• register — аналогично auto, но память выделяется по возможности в регистрах процессора.

*Пространство имен*

Использование пространство имен позволяет разрешать конфликты с именами в программе в случае различных сбоев. Для того чтобы определить пространство имен необходимо использовать namespace.

ПРИМЕР(необязательно!!)

*namespace Jack*

*{*

*void fetch();*

*int pal;*

*}*

*namespace Jill*

*{*

*double fetch;*

*int pal;*

*}*

*Jack::pal = 12;*

*Пространство имен можно объявлять и использовать. Для этого есть директива using.*

*Namespace Jill*

*{*

*double fetch;*

*};*

*char fetch;*

*int main() {*

*using Jill::fetch;*

*dobule fetch;*

*cin >> fetch;*

*cin >> ::fetch;*

*}*

*Для перевода всех переменных и функций из пространства имен в глобальную область используется using namespace*

*std::cout << «Hello» << endl;*

*using namespace std;*

*cout << «Hello» << endl;*