**ГЛАВА 3**

1. а) Для населения Москвы будет выбран тип данных long int так, как программа может запуститься на системе с 16-битным int, чего явно будет недостаточно.

б) Для стоимости фильма на DVD диске будет выбран тип short.

в) Для буквы будет выбран тип char так, как это символ.

г) Для количества раз буквы, которая чаще всего встречается в книге будет выбран тип short.

1. Переменную типа long следует использовать вместо int в случаях, когда от задачи требуется виделиние памяти для переменной больше 16-ти бит. Ведь int в разных системах может быть и 16 бит и 32, а long не может быть меньше 32-х бит.
2. Для получения 32-х битного числа со знаком, нужно использовать переносимый тип данных int32\_t и включение соответствующего заголовочного файла. Если программа будет компилироваться на системе, где int имеет размер 16 бит, то во время компиляции int32\_t будет заменен на тип данных long, который имеет 32 бита.
3. Тип и назначение констант:

а) '\b' --- тип char;

b) 1066 --- тип int;

в) 99.44 --- тип double (по умолчанию все константы с плавающей точкой сохраняются как double);

г) 0XAA --- тип данных int в шестнадцатиричной форме;

д) 2.0е30 --- тип double.

1. В первой строке пропущена # перед include.

Во второй строке не указан тип данных главной функции main и нету скобок с аргументом функции (void).

В третей строке нужно заменить круглое скобки на фигурные.

В четвёртой строке объявление переменной идет не через запятую, а через точку с запятой.

В шестой строке переменной g присвоено значение только порядка числа, а нужно написать ещё целую часть или дробную.

В седьмой строке происходит умножение переменных rate на g, но переменной rate не присвоили никакого значения.

В восьмой строке нужно добавить оператор возврата return 0.

И в девятой строке заменить круглую скобка на фигурную.

Правильный вариант программы, но с приблизительным значением данных:

#include <stdio.h>

int main(void)

{

float g, h;

float rate, tax;

g = 2.12e21;

rate = 30;

tax = rate \* g;

return 0;

}

1. Константа Тип Спецификатор

а) 12 int %d

б) 0X3 int. %#X

в) 'С'. char %c

г) 2.34Е07. double. %f

д) '\040' char. %c

е) 7.0. double %f

ж) 6L. long %ld

з) 6.0f. float. %f

и) 0x5.b6p12. double. %a

1. Константа Тип Спецификатор

а) 012 int %#o

б) 2.9е05L long double. %le

в) 's'. char %c

г) 100000 long int %ld

д) '\n' char. %c

е) 20.0 double %f

ж) 0x44 int %#x

з) -40 int %d

1. printf(«Шансы попасть в %d были в %ld к 1.\n», imate, shot);

printf(«Счет %f не соответствует уровню %c.\n», log, grade);

1. char ch;

ch = '\r';

ch = ‘13’;

ch = ‘\015’;

ch = ‘\xd';

1. #include <stdio.h>

//эта программа безупречна

int main(void)

{

int cows, legs;

printf("Сколько коровьих ног Вы насчитали?\n");

scanf("%d%", &legs);

cows = legs / 4;

printf("Отсюда следует, что есть %d коров(а,ы).\n", cows);

return 0;

}



а) \n – перевод курсора на новую строку;

б) \\ - написание символа косой черты;

в) \” – написание символа двойных кавычок;

г) \t – табуляция, перевод курсора на заданное количество пунктов вперёд.