## Лабораторная работа 2.

### Основные свойства фундаментальных типов данных.

Выполните задания исходя из вашего варианта.

|  |  |
| --- | --- |
| **Вариант** | **ФИО** |
| 1 | Абрамова Алина Андреевна |
| 2 | Боднар Игорь Андреевич |
| 3 | БОДРОВ ЕВГЕНИЙ ДМИТРИЕВИЧ |
| 4 | БОРИСОВ САМИР АЛЕКСАНДРОВИЧ |
| 5 | Голубев Максим Алексеевич |
| 6 | Горбенко Никита Андреевич |
| 7 | Игнатьев Артемий Андреевич |
| 8 | Кадарметов Дмитрий Николаевич |
| 9 | Ковалёв Егор Николаевич |
| 10 | Комиссаров Антон Павлович |
| 1 | Кузнецов Никита Сергеевич |
| 2 | ЛАПИЦКИЙ АРТЕМ АЛЕКСАНДРОВИЧ |
| 3 | Муллабаев Марсель Тимурович |
| 4 | РАННЕВ КИРИЛЛ АЛЕКСЕЕВИЧ |
| 5 | Смирнов Артём Сергеевич |
| 6 | Сорокожердьев Андрей Владимирович |
| 7 | Талалаев Михаил Викторович |
| 8 | Ампилова Луара Олеговна |
| 9 | Бережной Евгений Сергеевич |
| 10 | Брикунов Илья Алексеевич |
| 1 | Венсковская Анастасия Сергеевна |
| 2 | Горобец Илья Сергеевич |
| 3 | Дзейтова Дарья Дмитриевна |
| 4 | Кожин Юрий Михайлович |
| 5 | Колобова Александра Павловна |
| 6 | Коншина Арина Александровна |
| 7 | Никулин Никита Владимирович |
| 8 | Олькин Захар Игоревич |
| 9 | Ольховский Илья Сергеевич |
| 10 | Орденко Василий Вячеславович |
| 1 | Смирнов Виктор Михайлович |
| 2 | Сухих Всеволод Александрович |
| 3 | Астахов Кирилл Сергеевич |
| 4 | Болмосов Сергей Сергеевич |
| 5 | Гапонова Виктория Олеговна |
| 6 | Голубинская Олеся Олеговна |
| 7 | Дудорова Екатерина Алексеевна |
| 8 | Зайцева Любовь Игоревна |
| 9 | Кет Дмитрий Александрович |
| 10 | Климова Мишель Владимировна |
| 1 | Корнилов Дмитрий Сергеевич |
| 2 | Куклин Матвей Алексеевич |
| 3 | Мешкова Алиса Алексеевна |
| 4 | Поляков Андрей Андреевич |
| 5 | Сверлов Михаил Игоревич |
| 6 | Трофимов Павел Дмитриевич |
| 7 | Трунова Екатерина Ивановна |
| 8 | Тюриков Дмитрий Евгеньевич |
| 9 | Харламов Дмитрий Сергеевич |
| 10 | Шкаликов Матвей Дмитриевич |

1. Напишите программу, вычисляющую размер памяти, отводимой компилятором под типы:

|  |  |
| --- | --- |
| **Вариант** | **Задание** |
| 1 | *int* |
| 2 | *short int* |
| 3 | *long int* |
| 4 | *float* |
| 5 | *double* |
| 6 | *bool* |
| 7 | *char* |
| 8 | *signed char* |
| 9 | *unsigned char* |
| 10 | *signed int* |

Для расчёта размера памяти используйте функцию **sizeof().**

Оператор sizeof возвращает размер в байтах объекта или типа данных. Синтаксис его таков:

*sizeof (type name);*

*sizeof (object);*

*sizeof object;*

Тип bool определён в заголовочном файле stdbool.h

2. Напишите программу, вычисляющую размер памяти, отводимой под символы:

|  |  |
| --- | --- |
| **Вариант** | **Задание** |
| 1 | 1 |
| 2 | А (латинский алфавит) |
| 3 | А (русский алфавит) |
| 4 | \а (звуковой сигнал) |
| 5 | \n (перевод строки) |
| 6 | \v (вертикальная табуляция) |
| 7 | \t (горизонтальная табуляция) |
| 8 | ? (вопросительный знак) |
| 9 | ! (восклицательный знак) |
| 10 | а (латинский алфавит) |

3. Напишите программу, вычисляющую предельные значения (минимальное и максимальное возможные значения) для типов:

|  |  |
| --- | --- |
| **Вариант** | **Задание** |
| 1 | *int* |
| 2 | *short int* |
| 3 | *long* |
| 4 | *float* |
| 5 | *double* |
| 6 | *char* |
| 7 | *signed char* |
| 8 | *unsigned char* |
| 9 | *signed int* |
| 10 | *long double* |

В программе использовать макросы из файлов *limits.h* или *float.h.*

Заголовочный файл *limits.h* определяет константы с ограничениями целочисленного типа данных для конкретной системы и компилятора.

Ниже в таблице отображены все константы:

|  |  |
| --- | --- |
| Константы | Описание |
| CHAR\_BIT | Количество бит в символе типа char. |
| SCHAR\_MIN | Минимальное значение объекта типа signed char. |
| SCHAR\_MAX | Максимальное значение объекта типа signed char. |
| UCHAR\_MAX | Максимальное значение объекта типа unsigned char. |
| CHAR\_MIN | Минимальное значение объекта типа char. |
| CHAR\_MAX | Максимальное значение объекта типа char. |
| MB\_LEN\_MAX | Максимальное количество байтов многобайтового символа, любого языка. |
| SHRT\_MIN | Минимальное значение объекта типа short int. |
| SHRT\_MAX | Максимальное значение объекта типа short int. |
| USHRT\_MAX | Максимальное значение объекта типа unsigned short int. |
| INT\_MIN | Минимальное значение объекта типа int. |
| INT\_MAX | Максимальное значение объекта типа int. |
| UINT\_MAX | Максимальное значение объекта типа unsigned int. |
| LONG\_MIN | Минимальное значение объекта типа long int. |
| LONG\_MAX | Максимальное значение объекта типа long int. |
| ULONG\_MAX | Максимальное значение объекта типа unsigned long int. |

Заголовочный файл *float.h* описывает характеристики типа данных с плавающей точкой для конкретной системы и компилятора. Значение с плавающей точкой (вещественное число) состоит из четырех элементов:

* знак: либо отрицательный, либо положительный;
* основание: выражает количество значений, для кодирования чисел, которые могут быть представлены одной цифрой (2 для двоичной, 10 для десятичной, 16 для шестнадцатеричной, и т. д. )
* мантисса: значение, стоящее перед основанием. Количество цифр в этом значении определяет точность.
* степень: значение в которое возводится основание следующим образом:

вещественное значение = мантисса \* основаниестепень, с соответствующим знаком.

|  |  |
| --- | --- |
| Имя | Описание |
| FLT\_RADIX | Основание для всех типов данных с плавающей точкой (long double, double, float). |
| FLT\_MANT\_DIG (float) DBL\_MANT\_DIG (double) LDBL\_MANT\_DIG (long double) | Количество цифр мантиссы, соответственно для указанных типов данных. |
| FLT\_DIG (float) DBL\_DIG (double) LDBL\_DIG (long double) | Количество десятичных цифр, которые могут быть округлены в число с плавающей точкой и обратно, без потери данных. |
| FLT\_MIN\_EXP (float) DBL\_MIN\_EXP (double) LDBL\_MIN\_EXP (long double) | Минимальное отрицательное целое значение для степени, которое генерирует нормализованное число с плавающей точкой (запятой). |
| FLT\_MIN\_10\_EXP (float) DBL\_MIN\_10\_EXP (double) LDBL\_MIN\_10\_EXP (long double) | Минимальное отрицательное целое значение степени основания 10 выражение, которое будет генерировать нормализованное число с плавающей точкой (запятой). |
| FLT\_MAX\_EXP (float) DBL\_MAX\_EXP (double) LDBL\_MAX\_EXP (long double) | Максимальное целое значение для степени в нормализованной форме представления числа с плавающей точкой (запятой). |
| FLT\_MAX\_10\_EXP (float) DBL\_MAX\_10\_EXP (double) LDBL\_MAX\_10\_EXP (long double) | Максимальное целое значение для степени с основанием 10 в нормализованной форме представления числа с плавающей точкой (запятой). |
| FLT\_MAX (float) DBL\_MAX (double) LDBL\_MAX (long double) | Максимальные значения чисел с плавающей точкой (запятой). |
| FLT\_EPSILON (float) DBL\_EPSILON (double) LDBL\_EPSILON (long double) | Разница между 1 и минимальным значением, большим единицы, которое может быть представлено указанными типами данных. |
| FLT\_MIN (float) DBL\_MIN (double) LDBL\_MIN (long double) | Минимальные значения чисел с плавающей точкой (запятой) |

Оформите программный код **строго** согласно правилам из Лабораторной работы 1.

Загрузите отчёт по лабораторной работе в систему Moodle. Отчёт должен содержать:

- исходный код конечной программы (файл с расширением **.с**)

- файл с описанием результата работы программ

Файлы заархивировать в **zip** архив.