

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №1

по дисциплине: Алгоритмы и структуры данных
тема: «Встроенные структуры данных»

Выполнил: студент группы ПВ-223
Мелехов Артём Дмитриевич

Проверил:
асс. Солонченко Роман Евгеньевич

Белгород 2023 г.

Лабораторная работа №1 «Встроенные структуры данных»

Цель работы: Изучение базовых типов данных языка C как структур данных (СД).

Содержание отчета:

- Тема лабораторной работы;
- Цель лабораторной работы;
- Условия задач и их решение;
- Вывод.

Задания к лабораторной работе: вариант №7

```
long
double (comp)
typedef enum Animal {
    cat,
    dog,
    mouse,
    tiger
} Animal; (animal(cat, dog, mouse, tiger))
```

1. Для типов данных определить

(а) Абстрактный уровень представления СД

Для типа данных long:

- Абстрактный уровень представления: целое число, представленное в двоичном формате со знаком.
- Характер организованности: последовательный набор битов, где каждый бит представляет собой степень двойки.
- Изменчивость: значение числа может изменяться путем присваивания нового значения переменной.
- Набор допустимых операций: арифметические операции (+, -, *, /, %), операции сравнения (==, !=, <, >, <=, >=), операции побитового сдвига (<<, >>), побитовые операции (&, |, ^), присваивание (=), инкремент (++), декремент (--), условный оператор (?:).

Для типа данных double:

- Абстрактный уровень представления: вещественное число, представленное в формате с плавающей точкой.
- Характер организованности: число представлено в виде мантиссы и порядка, где мантисса представляет собой дробное число в двоичном формате, а порядок определяет местоположение запятой.

- Изменчивость: значение числа может изменяться путем присваивания нового значения переменной.

- Набор допустимых операций: арифметические операции (+, -, *, /), операции сравнения (==, !=, <, >, <=, >=), присваивание (=), инкремент (++), декремент (--), условный оператор (?:).

Для перечисления Animal:

- Абстрактный уровень представления: набор констант, представляющих животных.
- Характер организованности: каждая константа имеет свой уникальный целочисленный код.
- Изменчивость: перечисление не может изменяться во время выполнения программы.
- Набор допустимых операций: присваивание (=), сравнение (==, !=), доступ к элементам перечисления.

(b) Физический уровень представления СД:

Для типа данных long:

- Схема хранения: последовательный набор битов, где каждый бит представляет собой степень двойки.
- Объем памяти: 8 байт (64 бита).
- Формат внутреннего представления: двоичное число со знаком, где старший бит отводится под знак.
- Допустимые значения: от -9,223,372,036,854,775,808 до 9,223,372,036,854,775,807.
- Тип доступа к элементам: прямой доступ к значению переменной.

Для типа данных double:

- Схема хранения: формат с плавающей точкой IEEE 754.
- Объем памяти: 8 байт (64 бита).
- Формат внутреннего представления: мантисса и порядок, где мантисса представляет собой дробное число в двоичном формате, а порядок определяет местоположение запятой.
- Допустимые значения: от $\sim 4.9 \times 10^{-324}$ до $\sim 1.8 \times 10^{308}$.
- Тип доступа к элементам: прямой доступ к значению переменной.

Для перечисления Animal:

- Схема хранения: каждая константа имеет свой уникальный целочисленный код.

- Объем памяти: зависит от количества констант в перечислении.
- Формат внутреннего представления: целочисленные коды для каждой константы.
- Допустимые значения: константы, определенные в перечислении.
- Тип доступа к элементам: прямой доступ к константам перечисления.

(с) Логический уровень представления СД

```
long name; – long;

double name; – comp;

typedef enum Animal {
    cat,
    dog,
    mouse,
    tiger
} Animal; – animal(cat, dog, mouse, tiger)
```

2. Для заданных типов данных определить набор значений, необходимый для изучения физического уровня представления СД.

```
typedef enum Animal {
    cat,
    dog,
    mouse,
    tiger
} Animal;

int main() {
    long a_max = 57;
    long a_min = -41;

    double b = 12.5;

    enum Animal c = cat, d = dog, e = mouse, f = tiger;

    return 0;
}
```

3. Преобразовать значения в двоичный код

$$a_{max} = 57$$

1. Делаем деление на 2: $57 / 2 = 28$ (остаток 1)
2. Записываем остаток в конец двоичного кода: 1
3. Делаем деление на 2: $28 / 2 = 14$ (остаток 0)

- Ответ: $57_{10} = 0'000000000000000000000000111001_2$

$$b = 12.5$$

2. Порядок равен 0110, что в двоичной системе счисления равно 6. Так как формат числа предполагает смещенный порядок, то необходимо вычесть из него 0111 (бинарное представление числа 7) и получить -1 в десятичной системе счисления. Значит порядок равен $2^{\{-1\}} = 0.5$.

3. Мантисса равна 0.1000 в двоичной системе счисления. Её значение можно вычислить по формуле: $1 + \text{мантисса} = 1.1000$. Это число в двоичной системе счисления равно 1.5 в десятичной системе счисления.

4. Итоговое значение равно: $(-1)^0 * 2^{\{0.5\}} * 1.5 = 12.5$.

5. Разработать и отладить программу, выдающую двоичное представление значений, заданных СД

```
#include <stdio.h>

typedef enum Animal {
    cat, dog, mouse, tiger
} Animal;

void print_byte(unsigned char a) {
    for (int i = 7; i >= 0; i--) {
        printf("%d", (a >> i) & 1);
    }

    printf(" ");
}

void print_var(void *a, unsigned size) {
    unsigned char *p = a;

    for (int i = size - 1; i >= 0; i--) {
        print_byte(*(p + i));
    }

    printf("\n");
}

int main() {
    long l = 57;
    long m = -41;
    double d = 3.14159265358979323846;
    Animal animal = tiger;

    printf("Binary representation of long l:\n");
    print_var(&l, sizeof(l));

    printf("Binary representation of long m:\n");
    print_var(&m, sizeof(m));
}
```

```

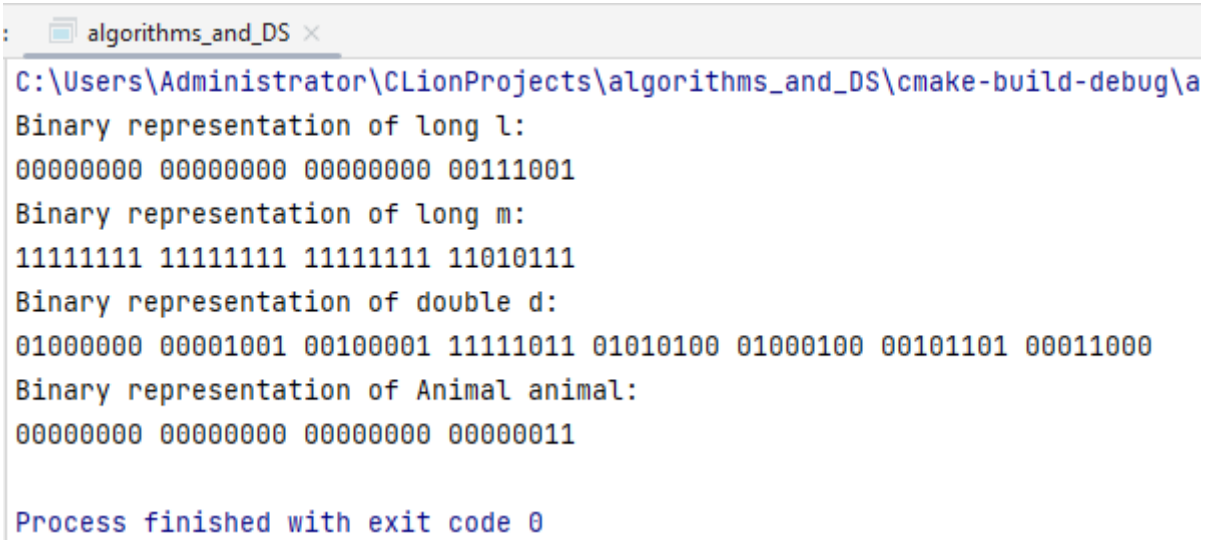
printf("Binary representation of double d:\n");
print_var(&d, sizeof(d));

printf("Binary representation of Animal animal:\n");
print_var(&animal, sizeof(animal));

return 0;
}

```

6. Обработать программой значения, полученные в результате выполнения пункта 3 задания. Сделать выводы.



```

: algorithms_and_DS x
C:\Users\Administrator\CLionProjects\algorithms_and_DS\cmake-build-debug\a
Binary representation of long l:
00000000 00000000 00000000 00111001
Binary representation of long m:
11111111 11111111 11111111 11010111
Binary representation of double d:
01000000 00001001 00100001 11111011 01010100 01000100 00101101 00011000
Binary representation of Animal animal:
00000000 00000000 00000000 00000011

Process finished with exit code 0

```

Выводы: в ходе выполнения работы были изучены базовые типы данных языка C как СД.