

Лабораторна робота №6. Масиви

1 Вимоги

1.1 Розробник

- Силка Михайло Миколайович;
- студент групи КІТ-121Г;
- 18-гру-2021.

1.2 Загальне завдання

Переробити лінійну програму у функціональну

1.3 Індивідуальне завдання

Створити функціональну програму яка буде конвертувати числа у слова

2 Опис програми

2.1 Функціональне призначення

Програма призначена для конвертації числа в слова

При конвертації числа діють наступні обмеження:

- числа які передаються в функцію повинні бути int
- число повинно бути в діапазоні від 0 до 9999 включно

Результат зберігається у змінній m2.

Подивитися результат виконання можна тільки в дебагері

2.2 Опис логічної структури

Програма визначає тип числа, цифри розрядів, та на основі цієї інформації записує в результуючу строку m2 конвертоване число у словесному вигляді

2.2.1 Функція визначення типу числа

```
int x_rank (int x)
```

Призначення: повертання типу числа переданого як аргумент

Опис роботи: завдяки блоку перевірок умовами яких є різне ділення, визначається тип числа (цифра, десятків, сотня, тисяча)

2.2.2 Функція визначення цифр розрядів

```
void get_ntht_num (int x, int x_is, int* n_num, int* ten_num, int* hun_num, int*  
thous_num)
```

Призначення: завдяки вказівникам записує в змінні розрядів їх цифри

Опис роботи: завдяки блоку перевірок записує в змінні, на які були передані вказівники як аргументи, цифри цих розрядів

2.2.3 Функція визначення типу десятка

```
int get_ten_type (int n_num, int ten_num)
```

Призначення: визначає та повертає тип десятка (чистий, чистий з цифрою, неправильний)

Опис роботи: завдяки блоку перевірок та змінним переданим як аргументи, визначає тип десятку

2.2.4 Основна функція

```
int main (int argc, char** argv)
```

Призначення: головна функція.

Опис роботи:

- якщо через аргументи командоні строки нічого не передається, то для змінної x створюється випадкове число;
- число, яке було передано через аргументи строки або випадкове, передаються в функцію `int cycle_for (int x);`
- виводи необхідних функцій для визначення типу числа, цифр розрядів та типу десятка
- на основі отриманих результатів з функцій йде запис результату конвертації числа у слова у результуючу строку m2

2.2.3 Структура проекту

lab06

├── Doxyfile

├── Makefile

├── doc

| ├── rep_lab06.docx

| ├── rep_lab06.md

```
|   └── rep_lab06.pdf
└── src
    └── main.c
```

2.3 Важливі фрагменти програми

Визначення числа

```
if (argc == 1){
    srand (time(NULL));
    x = (int)random() % 10000;
} else {
    x = (int)strtod(argv[1], 0);
}
```

Визначення типу числа

```
int x_rank (int x) {
    int x_is;
    if (x < 10000 && x >= 0){
        if (x / 10 < 1){
            x_is = 0;
            //return (0);
        } else if (x / 10 >= 1 && (x / 10) < 10) {
            x_is = 1;
            //return (1);
        } else if (x / 10 >= 10 && x / 10 < 100){
            x_is = 2;
            //return (2);
        } else if (x / 10 >= 100){
```

```

        x_is = 3;
        //return (3);
    }
}
return (x_is);
}

```

Визначення цифр розрядів

```

void get_ntht_num (int x, int x_is, int* n_num, int* ten_num, int* hun_num, int*
thous_num) {
    if (x_is == 1){
        *ten_num = x / 10;
        *n_num = x % 10;
    } else if (x_is == 2){
        *hun_num = x / 100;
        *ten_num = (x / 10) % 10;
        *n_num = x % 10;
    } else if (x_is == 3){
        *thous_num = x / 1000;
        *hun_num = (x / 100) % 10;
        *ten_num = (x / 10) % 10;
        *n_num = x % 10;
    } else {
        *n_num = x;
    }
}
}

```

Визначення типу десятка

```

int get_ten_type (int n_num, int ten_num){

```

```

int ten_type; //0 - Чистый; 1 - Чистый с цифрой; 2 - Неправильный
if (n_num > 0 && n_num < 10 && ten_num == 1){
    ten_type = 2;
} else if (ten_num != 0 && n_num == 0){
    ten_type = 0;
} else if (ten_num > 1 && n_num != 0){
    ten_type = 1;
}
return (ten_type);
}

```

Варіант використання

Для демонстрації результатів задачі використовується:

- покрокове виконання програми в утиліті lldb;

Варіант використання: послідовність дій для запуску програми у режимі відлагодження:

- запустити програму у відлагоднику lldb;
- поставити точку зупинки на функції main (строка з `return 0;`);
- запустити програму;
- подивитись результати виконання програми, оглядаючи значення змінної m2

Process 7526 stopped

* thread #1, name = 'main.bin', stop reason = breakpoint 1.1

frame #0: 0x0000000000401b20 main.bin`main(argc=1, argv=0x00007ffffffdf18)
at main.c:222:5

```
219          }
```

```
220      }
```

```
221
-> 222     return(0);
223 }
224
225 //Return: Единица = 0; Десяток = 1; Сотня = 2; Тысяча = 3
(lldb) p m2
(char [50]) $1 = "seven thousand three hundred eighty eight"
(lldb)
```

Висновки:

При виконанні даної лабораторної роботи я покращив свої навички роботи з циклами, строками, масивами строк та функціями