

Практическое занятие 6

Именованные константы

Пименов Евгений Сергеевич

Курс «Программирование»

Сибирский государственный университет телекоммуникаций и информатики (Новосибирск)

Осенний семестр, 2016

Недостатки кода?

2

```
01. for (i = 0; i < 12; ++i) {  
02.     profit[i] = revenue[i] - expence[i];  
03. }
```

Недостатки кода?

3

```
01. for (i = 0; i < 12; ++i) {  
02.     profit[i] = revenue[i] - expence[i];  
03. }
```

Вариант 1: использование макроса

4

```
01. #define MONTH_IN_YEAR 12
02.
03. for (i = 0; i < MONTH_IN_YEAR; ++i) {
04.     profit[i] = revenue[i] - expence[i];
05. }
```

```
01. #define MONTH_IN_YEAR 12
02.
03. for (i = 0; i < MONTH_IN_YEAR; ++i) {
04.     profit[i] = revenue[i] - expence[i];
05. }
```

Этапы компиляции?

1. Препроцессинг
2. Компиляция
3. Ассемблирование
4. Линковка

- Подстановка выполняется на этапе препроцессинга
- Результат подстановки известен на этапе компиляции (вывод?)
- Может породить синтаксически некорректный код

```
01. #define MONTH_IN_YEAR 12;
02.
03. for (i = 0; i < MONTH_IN_YEAR; ++i) {
04.     profit[i] = revenue[i] - expence[i];
05. }
```



```
01. #define MONTH_IN_YEAR 12;
02.
03. for (i = 0; i < MONTH_IN_YEAR; ++i) {
04.     profit[i] = revenue[i] - expence[i];
05. }
```

```
$ gcc -Wall main.c
```

```
error: expected expression before ';' token
```

```
    for (i = 0; i < MONTH_IN_YEAR; ++i) {
```

```
                ^
```

```
$ gcc -E main.c  
for (i = 0; i < 12;; ++i) {  
    profit[i] = revenue[i] - expence[i];  
}
```

```
$ gcc -E main.c  
for (i = 0; i < 12;; ++i) {  
    profit[i] = revenue[i] - expence[i];  
}
```

Обычно рекомендуется не использовать препроцессор без крайней необходимости.

```
01. const int MONTH_IN_YEAR = 12;
02.
03. for (i = 0; i < MONTH_IN_YEAR; ++i) {
04.     profit[i] = revenue[i] - expence[i];
05. }
```

- В языке C константные переменные не являются константными выражениями.
- Рекомендуется использовать ключевое слово `const` везде, где это возможно:

```
const double discriminant = b * b - 4 * a * c;
```

```
01. enum Month {
02.     January,
03.     February,
04.     ...
05.     December
06. };
07.
08. enum Month month;
09.
10. for (month = January; month <= December; ++month) {
11.     profit[month] = revenue[month] - expence[month];
12. }
```

```
01. enum Month {  
02.     January = 1,  
03.     February, // 2  
04.     ...  
05.     December // 12  
06. };
```

```
01. enum {  
02.     January = 1,  
03.     February, // 2  
04.     ...  
05.     December // 12  
06. };
```



```
01. enum HttpResponse {
02.     Ok = 200,
03.     ClientError = 404
04.     ...
05. };
06.
07. enum SubprocessStatus {
08.     Ok,
09.     Fail
10. };
```

error: redeclaration of enumerator 'Ok'

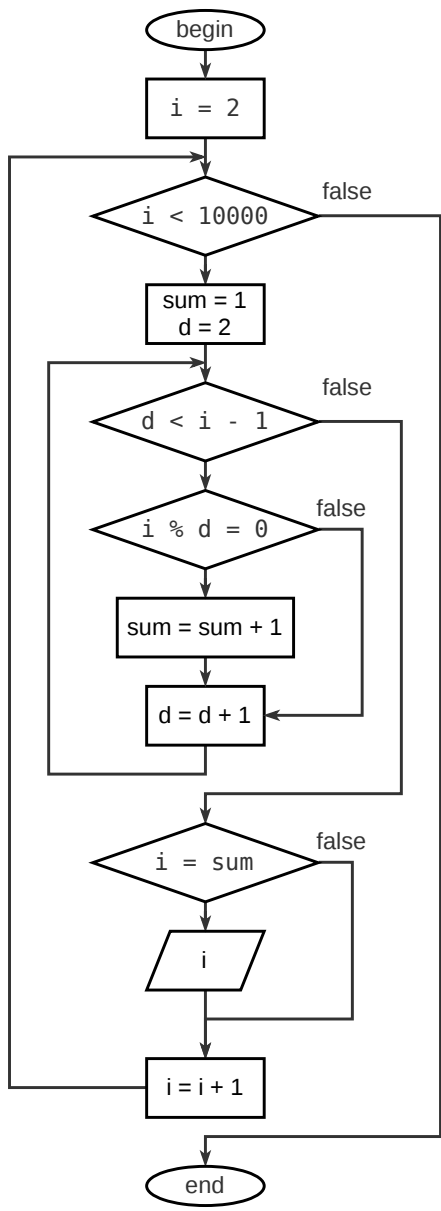
```
01. enum HttpResponse {
02.     HttpResponseOk = 200,
03.     HttpResponseClientError = 404
04.     ...
05. };
06.
07. enum SubprocessStatus {
08.     SubprocessStatusOk,
09.     SubprocessStatusFail
10. };
```

```
01. enum WeekDay unknown_week_day = December;  
02.  
03. enum WeekDay first_week_day = Sunday;  
04. enum Month first_month = January;  
05.  
06. printf("%d\n", first_week_day == first_month);
```

Компилятор сообщает только предупреждение при попытке сравнения.

Найти все совершенные числа до 10000. Совершенное число - это такое число, которое равно сумме всех своих делителей, кроме себя самого.

Например, число 6 является совершенным, т.к. кроме себя самого делится на числа 1, 2 и 3, которые в сумме дают 6.



Вывести на экран, из каких простых множителей состоит введенное натуральное число.

Задание 2: псевдокод

23

```
d = 2
n = input()

while d * d <= n:
    if n % d = 0:
        print d
        n = n / d
    else:
        d = d + 1

if n > 1:
    print n
```

Разработать приложения для решения следующих пазлов:

1. $(L + O + G + I + C)^3 = \text{LOGIC}$
2. $C^L = \text{LOGIC}$
3. $(DD)^E = \text{DEED}$

Одинаковым символам соответствуют одинаковые значения разрядов.

man 3 pow

Разработать приложение для вычисления суммы элементов последовательности вида:

2, 4, 8, 16, 32...

Пользователь вводит первый элемент и их общее количество.