

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВВГУ»)
ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ
КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №4
по дисциплине
«Информатика и программирование»

Студент
гр. БИН-25-3 _____ Л.А. Сидоров
Ассистент
преподавателя _____ М.В. Водяницкий

Задание

Выполнить задания и оформить отчет по стандартам ВВГУ.

Задание 1. Написать программу, которая определяет, как будет вести себя кондиционер. Если температура в помещении 20 градусов и выше, то кондиционер выключается, если меньше - включается. Температура вводится пользователем с консоли.

Пример:

Введите температуру: 18

Кондиционер включен

Задание 2. Год делится на четыре сезона: зима, весна, лето и осень. Написать программу, которая запрашивает у пользователя номер месяца и выводит к какому сезону этот месяц относится.

Пример:

Введите номер месяца: 4

Это весна

Задание 3. Считается, что один год, прожитый собакой, эквивалентен семи человеческим годам. При этом зачастую не учитывается, что собаки становятся абсолютно взрослыми уже к двум годам. Таким образом, многие предпочитают каждый из первых двух лет жизни собаки приравнивать к 10.5 годам человеческой жизни, а все последующие к 4.

Написать программу, которая будет переводить собачий возраст в человеческий. Программа должна корректно обрабатывать входные данные и выводить соответствующие сообщения об ошибках:

- Если вводится не число
- Если вводится число меньше 1
- Если вводится число большее 22

Пример:

Введите возраст собаки (в годах): 5

Возраст собаки в человеческих годах: 33.0

Пример:

Введите возраст собаки (в годах): 0

Ошибка: возраст должен быть не меньше 1

Задание 4. Число делиться на 6 только в случае соблюдения двух условий:

- Последняя цифра четная
- Сумма всех цифр делиться на 3

Написать программу, которая выведет делиться ли введенное число на 6 или нет.

Задание 5. Написать программу, которая будет проверять пароль на надежность.

Пароль считается надежным, если его длина не менее 8 символов и если он содержит:

- Заглавные буквы латиницы
- Строчные буквы латиницы
- Числа
- Специальные знаки

В случае, если пароль не проходит по одному из условий, необходимо сообщить пользователю каким именно условиям он не удовлетворяет.

Пример:

Введите пароль: qwerty

Пароль ненадежный: отсутствуют заглавные буквы, числа и специальные символы

Задание 6. Написать программу, которая определяет, является ли введенный пользователем год високосным. Год считается високосным, если он делится на 4, но не делится на 100, либо если он делится на 400.

Пример:

Введите год: 2024

2024 - високосный год

Задание 7. Написать программу, которая запрашивает у пользователя три числа и выводит на экран наименьшее из них. При решении нельзя использовать встроенные функции `min()` и `max()`.

Пример:

Введите три числа: 8 3 5

Наименьшее число: 3

Задание 8. В магазине проводится акция. Акция работает по следующим правилам:

- Сумма < 1000 => скидка - 0%
- Сумма < 5000 => скидка - 5%
- Сумма < 10000 => скидка - 10%
- Сумма > 10000 => скидка - 15%

Напишите программу, которая запрашивает сумму покупки и выводит размер скидки и итоговую сумму к оплате.

Пример:

Введите сумму покупки: 7500

Ваша скидка: 10%

К оплате : 6750.0

Задание 9. Написать программу, которая определяет время суток по введенному часу (целое число от 0 до 23).

- С 0 до 5 часов - ночь
- С 6 до 11 часов - утро
- С 12 до 17 часов - день
- С 18 до 23 часов - вечер

Пример:

Введите час (0-23): 20

Сейчас вечер

Задание 10. Написать программу, которая определяет, является ли введенное число простым. Число называется простым, если оно больше 1 и делится только на 1 и само себя. Программа должна корректно обрабатывать некорректный ввод и выводить соответствующие сообщения об ошибках.

Пример:

Введите число: 17

17 - простое число

Содержание

1 Выполнение работы	3
1.1 Задание 1	3
1.2 Задание 2	3
1.3 Задание 3	4
1.4 Задание 4	4
1.5 Задание 5	5
1.6 Задание 6	7
1.7 Задание 7	8
1.8 Задание 8	8
1.9 Задание 9	9
1.10 Задание 10	10

1 Выполнение работы

1.1 Задание 1

Сначала обозначим переменную temperature типа float. Далее просим пользователя ввести значение температуры, которое записываем в нашу переменную. После чего обрабатываем значение по условиям, если $\text{temperature} \geq 20$, то выводим в консоль, кондиционер выключен, в ином случае, выведем, что он включен. На рисунке 1 представлен код полученной программы.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     float temperature;
6
7     printf("Введите температуру: ");
8     scanf("%f", &temperature);
9
10    if (temperature >= 20) printf("Кондиционер выключен \n");
11    else printf("Кондиционер включен \n");
12 }
```

Рисунок 1 – Листинг программы для задания 1

1.2 Задание 2

Сначала заводим переменную monthNum типа integer. Далее просим пользователя ввести интересующий его месяц. Затем проверяем данные на корректность по условию $1 \leq \text{monthNum} \leq 12$. И наконец обрабатываем по условиям. Если $3 \leq \text{monthNum} \leq 5$, то выводим весна. Если $6 \leq \text{monthNum} \leq 8$, то выводим лето. Если $9 \leq \text{monthNum} \leq 11$, то выводим осень. А на оставшиеся варианты выводим зима. На рисунке 2 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int monthNum;
6
7     printf("Введите номер месяца: ");
8     scanf("%d", &monthNum);
9
10    if (monthNum >= 1 && monthNum <= 12)
11    {
12        printf("Это ");
13        if (monthNum >= 3 && monthNum <= 5) printf("весна\n");
14        else if (monthNum >= 6 && monthNum <= 8) printf("лето\n");
15        );
16        else if (monthNum >= 9 && monthNum <= 11) printf("осень\n");
17        else printf("зима\n");
18    }
19    else printf("Ошибка! Данные некорректны!");
20 }
```

Рисунок 2 – Листинг программы для задания 2

1.3 Задание 3

Сначала заводим целочисленную переменную dogAge. Далее просим ввести возраст собаки и проверяем введенные данные на корректность по условию $1 \leq \text{dogAge} \leq 22$. Затем конвертируем возраст собаки в человеческий по следующему алгоритму: если $\text{dogAge} \leq 2$, то $\text{dogAge} * 10.5$, в ином случае мы получаем следующее выражение $-21 + (\text{dogAge}-2)*4$. После выводим в консоль результат. На рисунке 3 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int dogAge;
6
7     printf("Введите возраст собаки в( годах): ");
8     scanf("%d", &dogAge);
9
10    if (dogAge >= 1 && dogAge <= 22)
11    {
12        printf("Возраст собаки в человеческих годах: ");
13        if (dogAge <= 2) printf("%.1f \n", dogAge*10.5f);
14        else printf("%d \n", 21+(dogAge-2)*4);
15    }
16    else printf("Ошибка ! Вы вышли за допустимый диапазон =
17 [1;22] или ввели не число.\n");
```

Рисунок 3 – Листинг программы для задания 3

1.4 Задание 4

Сначала заведем переменную num типа integer. Далее просим пользователя ввести интересующее его число. Затем проверим число на четность, используя операцию побитового

умножения, проверяя последний бит числа в двоичной системе счисления на наличие единицы - в случае ее наличия, число нечетное. После проверяем делится ли сумма цифр числа на 3 без остатка. В случае соблюдения всех условий, выводим в консоль - число делится на 6, в ином случае - не делится. На рисунке 4 представлен код решения.

```

1 #include <stdio.h>
2
3 int getSumDigitsOfNum(int num)
4 {
5     int res = 0;
6     while (num > 0)
7     {
8         res += num%10;
9         num /= 10;
10    }
11    return res;
12 }
13
14 int main()
15 {
16     int num;
17
18     printf("Введите число: ");
19     scanf("%d", &num);
20
21     if (!(num & 1))
22     {
23         if (getSumDigitsOfNum(num) % 3 == 0) printf("Число
24             делится на 6. \n");
25         else printf("Число не делится на 6. \n");
26     }
27 }
```

Рисунок 4 – Листинг программы для задания 4

1.5 Задание 5

Для решения нам понадобятся следующие переменные:

- массив password для типа char, в нем мы будем хранить пароль.
- literal типа integer, ее мы будем использовать для временного хранения числового представления обрабатываемого символа.
- i типа integer, ее будем использовать в качестве итератора, для перемещения по массиву из строк в цикле while.
- haveUpperCase типа integer, будем использовать в качестве логической. Присваиваем 1, если в пароле есть символы верхнего регистра.
- haveLowerCase типа integer, будем использовать в качестве логической. Присваиваем 1, если в пароле есть символы нижнего регистра.
- haveNum типа integer, будем использовать в качестве логической. Присваиваем 1, если в пароле есть цифры.

- haveSpecLit типа integer, будем использовать в качестве логической. Присваиваем 1, если в пароле есть специальные символы.

Сначала просим пользователя ввести придуманный пароль, записываем его в password. Далее проверяем его длину, если она более или равна 8 символам, то продолжаем, в ином случае сообщаем о ненадежности пароля. После в цикле while пробегаемся по символам пароля, проверяя каждый и отмечая с помощью логических переменных наличие: символов верхнего и нижнего регистров, специальных символов и цифр. Символы обрабатываем по их числовому представлению, согласно базовой таблице ASCII. Когда символы закончились проверяем, что у нас есть все необходимые символы, если они есть, от выводим в консоль, что пароль надежный, в ином случае говорим, что пароль не надежный и с помощью логических переменных определяем почему. На рисунке 5 представлен код программы.

```

1 #include <stdio.h>
2
3 int getStrLength(char string[])
4 {
5     int i = 0;
6     while (string[i++] != '\0');
7     return i;
8 }
9
10 int main()
11 {
12     char password[50];
13     int literal;
14     int i = 0;
15
16     int haveUpCase = 0;
17     int haveLowCase = 0;
18     int haveNum = 0;
19     int haveSpecLit = 0;
20
21     printf("Допустимые* спец. символы: ! \" # $ %% & \' ) ( * +\\n");
22
23     printf("Придумайте пароль: ");
24     scanf("%50s", password);
25
26     if(getStrLength(password) >= 8)
27     {
28         while (password[i] != '\0')
29         {
30             literal = (int)password[i++];
31             if (literal >= 97 && literal <= 122) haveLowCase = 1;
32             else if (literal >= 65 && literal <= 90) haveUpCase = 1;
33             else if (literal >= 48 && literal <= 57) haveNum = 1;
34             else if (literal >= 33 && literal <= 43) haveSpecLit = 1;
35         }
36
37         if(haveNum && haveUpCase && haveLowCase && haveSpecLit)
38             printf("Пароль надежный.\n");
39         else
40         {
41             printf("-----\n");
42             printf("Пароль ненадежный. В нем нет:\n");
43             if (!haveNum) printf("числа\n");
44             if (!haveUpCase) printf("заглавной буквы латыни\n");
45             if (!haveLowCase) printf("строчной буквы латыни\n");
46             if (!haveSpecLit) printf("специального символа\n");
47         }
48     }
49     else printf("Пароль ненадежный в нем менее 8 символов.\n");
50 }

```

Рисунок 5 – Листинг программы для задания 5

1.6 Задание 6

Сначала заводим переменную year типа integer. Затем просим пользователя ввести год и записываем в year, проверяя на корректность по условию - year ≥ 0 . Далее обрабатываем значение, если оно делится одновременно на 4 и 100 без остатка или на

400, то выводим в консоль, что год високосный, в ином случае - не високосный. На рисунке 6 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int year;
6
7     printf("Введите год: ");
8     scanf("%d", &year);
9
10    if (year > 0)
11    {
12        if ((year % 4 == 0 && year % 100 != 0) || (year % 400 ==
13            0)) printf("%d - високосный.\n", year);
14        else printf("%d - не високосный\n", year);
15    }
16    else printf("Ошибка! Некорректный год.");
17 }
```

Рисунок 6 – Листинг программы для задания 6

1.7 Задание 7

Сначала зеведем массив для 3 целочисленных чисел и переменную maxNum типа integer и присвоим ей наименьшее значение, допустимое типом данных. Получаем от пользователя 3 числа, которые записываем в массив. Далее с помощью цикла бегаем по массиву и ищем самое большое число сравнивая с maxNum. После выводим в консоль наибольшее значение.На рисунке 7 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int nums[3];
6     int maxNum;
7
8     printf("Введите 3 числа через пробел: ");
9     scanf("%d %d %d", &nums[0], &nums[1], &nums[2]);
10
11    maxNum = nums[0];
12
13    for (int i = 1; i <= 2; i++)
14    {
15        if (maxNum < nums[i]) maxNum = nums[i];
16    }
17
18    printf("%d \n", maxNum);
19 }
```

Рисунок 7 – Листинг программы для задания 7

1.8 Задание 8

Сначала заведем переменные sum и discount типа float. Далее получаем сумму покупки от пользователя и записываем в sum. После проверяем корректность данных,

сумма должна быть ≥ 0 . Затем задаем условия, в которых описываем при каких условиях, какая скидка.

- Если $\text{sum} < 1000$, то $\text{discount} = 0$.
- Если $1000 \leq \text{sum} < 5000$, то $\text{discount} = 0.05$
- Если $5000 \leq \text{sum} < 10000$, то $\text{discount} = 0.1$
- Во всех остальных случаях $\text{discount} = 0.15$

После выводим скидку и сумму к оплате, после применения скидки. На рисунке 8 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int sum;
6     float discount;
7
8     printf("Введите сумму покупки: ");
9     scanf("%d", &sum);
10
11    if (sum > 0)
12    {
13        if (sum < 1000) discount = 0;
14        else if (sum >= 1000 && sum < 5000) discount = 0.05f;
15        else if (sum >= 5000 && sum < 10000) discount = 0.1f;
16        else if (sum >= 10000) discount = 0.15f;
17
18
19        printf("Скидка %.0f%% \n", discount*100);
20        printf("Итоговая сумма к оплате: %.1f \n", sum - sum*
21            discount);
22    }
23    else printf("Ошибка. Некорректная сумма.\n");
24 }
```

Рисунок 8 – Листинг программы для задания 8

1.9 Задание 9

Сначала обозначим переменную `hours` типа `integer`. После просим пользователя ввести интересующий его час суток, проверяя на корректность по условию $- 0 \leq \text{hours} \leq 23$. Далее определяем условия по которым определяем текущее время суток:

- Если $0 \leq \text{hours} \leq 5$, то выводим в консоль, что сейчас ночь.
- Если $6 \leq \text{hours} \leq 11$, то выводим в консоль, что сейчас утро.
- Если $12 \leq \text{hours} \leq 17$, то выводим в консоль, что сейчас день.
- Во всех остальных случаях выводим в консоль, что сейчас вечер.

На рисунке 9 представлен код программы.

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int hours;
6
7     printf("Введите час [0;23]: ");
8     scanf("%d", &hours);
9
10    if (hours >= 0 && hours <= 23)
11    {
12        printf("Сейчас ");
13        if (hours >= 0 && hours <= 5) printf("ночь.\n");
14        else if (hours >= 6 && hours <= 11) printf("утро.\n");
15        else if (hours >= 12 && hours <= 17) printf("день.\n");
16        else printf("вечер.\n");
17    }
18    else printf("Ошибка. Некорректное время.");
19 }
```

Рисунок 9 – Листинг программы для задания 9

1.10 Задание 10

Заводим переменные num и numOfDeviders, инициализируемую 0. Далее просим пользователя ввести число, которое он хочет проверить на простоту, и запишем в num. После проверим, что $num > 0$. Затем, чтобы уменьшить дальнейшее кол-во итераций, проверим на четность. Поскольку любое четное число, кроме двойки, уже имеют минимум 4 делителя - 1, 2, само число и результат деления на 2. Дальше в цикле for пробегаем по числам от 1 до num и проверяем делимость без остатка, в случае наличия остатка к numOfDeviders прибавляем 1. Если $numOfDeviders > 2$, то выводим, что число составное, в ином случае выводим - простое. На рисунке 10 представлен код программы.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int num;
6     int numOfDeviders = 0;
7
8     printf("Введите число: ");
9     scanf("%d", &num);
10
11    printf("Число %d - ", num);
12
13    if (num > 0)
14    {
15        if ((num & 1) || num == 2)
16        {
17            for (int i = 1; i <= num; i+= 2)
18            {
19                if (num % i == 0) ++numOfDeviders;
20                if (numOfDeviders > 2)
21                {
22                    printf("составное.\n");
23                    break;
24                }
25            }
26            printf("простое.\n");
27        }
28        else printf("составное.\n");
29    }
30    else printf("Ошибка! Некорректное число. ");
31 }
```

Рисунок 10 – Листинг программы для задания 10