

```
#include <iostream.h>

int main()
{
    //в комментария пишу псевдокод для лучшего чтения программы
    char detal[10], tipe[10];
    int how_many[10], g, add, h, none, many, sdvig_tipe, ch, now;
    bool yesorno;
    bool b1, b2;
    int name_number[10];
    char io[8], iq[8];
    char marx[] = "BMW GAZ KIA";
    char yes[] = "YES";
    char exit[] = "exit";
    for (int m=0; m!=10; m++)
    {
        // задаю значения, на складе ничего нет – количество деталей в каждой из 10 ячеек равно 0
        // ячейка с деталью количество которых равно 0 – считается пустой.
        how_many[m]=0;
        name_number[m]=0;
        cout << how_many[m];
    }
    while(iq[0]!=exit[0] && iq[1]!=exit[1] && iq[2]!=exit[2] && iq[3]!=exit[3])
    {
        cout << "\nЗапущен цикл добавления данных в базу\n";
        cout << "желаете выйти из режима ввода данных? наберите YES\n";
        cin >> iq;
        while (iq[0]!=yes[0] && iq[1]!=yes[1] && iq[2]!=yes[2])
        {
            many=0;
            for(int m=0; m!=8; m++)
            {
                iq[m]=marx[3]; // заполнение внешнего массива ввода вывода пробелами для избежания ошибок при проверке на
                               наличие
            }
            cout << "введите маркировку детали\n";
            cin >> iq; // ввод детали
            cout << "введите количество данной детали \n";
            cin >> many;
```

```
g=0;
add=0;

while (g != 10) // цикл проверки существует ли данная деталь на складе
{
    for(int m=0; m!=8; m++)
    {
        io[m]=marx[3]; // заполнение пробелами внутреннего массива ввода вывода (бетта тест)
    }

    ch=5;
    now=0;
    while (iq[ch]!=marx[3]) // Операция поиска конца ввода маркировки
    {
        ch++;
    }

    sdvig_tipe=4;

    b1=0;
    b2=0;

    io[0]=detal[g];

    if (name_number[g]==1) // name_number – массив кодировки марки автомобиля bmw -1, GAZ - 2, kia-3
    {
        io[1]=marx[0];
        io[2]=marx[1]; // генерация маркировки если бумер
        io[3]=marx[2];

    }
    if (name_number[g]==2)
    {
        io[1]=marx[4];
        io[2]=marx[5];
        io[3]=marx[6]; // генерация маркировки если волга

    }
    if (name_number[g]==3)
```

```
{
    io[1]=marx[9];
    io[2]=marx[10];
    io[3]=marx[11];    // генерация маркировки если кия
}

while (io[sdvig_tipe] != marx[3]) // Операция сдвига переменной tipe в массиве io,
{
    sdvig_tipe++;           //данная операция предназначена для марок автомобилей имеющих в названии
    более 3-х символов,
}

io[sdvig_tipe]=tipe[g];    // вывод сдвинутого символа во внутренний массив вывода.

/*switch (ch)              // операция сдвига ячейки массива в функции проверки на наличие.
{
case 6:
{
    b1=1;
    cout << ch;
} break;
case 7:
{
    b2=1;
    cout << ch;
} break;
}

if (iq[0]==io[0] && iq[1]==io[1] && iq[2]==io[2] && iq[3]==io[3] && iq[4]==io[4] && iq[(b1+5)]==io[5] &&
    iq[(b2+6)]==io[6])
{
    how_many[g]=how_many[g]+many;    // проверка соответствия детали текущей ячейки "g"
    g=10;
    cout << "good \n";
}else
{
    g++;    // если соответствие по ячейкам не обнаружено
    add++;  // тогда add = 10 и запускается процесс добавления детали в пустую ячейку
```

```
    }*/  
  
    // цикл проверки на соответствие ВЕТТА 2!!! 100% работа не гарантируется!  
    ch=ch-1;  
    while (iq[now]==io[now] && now != 9)  
    {  
        now++;  
        if(ch==now) // Если проверка дошла до конца – "now = ch" тогда увеличивается количество данного товара.  
        {  
            now=9;  
            how_many[g]=how_many[g]+many;  
            cout << "good \n";  
            g=10;  
        }  
    }  
    if (now < 4)  
    {  
        g++;  
        add++;  
    }  
}  
if (add==10) // запуск операции добавление в свободную ячейку  
{  
    cout << "\n add \n";  
    h=0;  
    none=0;  
    while (how_many[h]!=0 && none !=10)  
    {  
        h++; // операция поиска свободной ячейки  
        none++; // если обнаружена пустая текущая ячейка (none < 10) тогда запустить операцию по  
                добавлению детали в эту ячейку  
    } // если пустой ячейки не обнаружено – none = 10 перейти к следующему этапу программы  
    cout << h << "\t" << none;  
    if (none < 10)  
    {  
  
        detal[h]=iq[0];  
        if (iq[1]==marx[0] && iq[2]==marx[1] && iq[3]==marx[2])  
        {  
            cout << h << "\tбумер" ;  
            name_number[h]=1;  
        }  
    }  
}
```

```
        tipe[h]=iq[4];
        how_many[h]=many;
    }
    if (iq[1]==marx[4] && iq[2]==marx[5] && iq[3]==marx[6])
    {
        name_number[h]=2;
        cout << h << "\tволга" ;
        tipe[h]=iq[4];
        how_many[h]=many ;
    }
    if (iq[1]==marx[9] && iq[2]==marx[10] && iq[3]==marx[11])
    {
        name_number[h]=3;
        cout <<h << "\tкияааа :-)" ;
        tipe[h]=iq[4];
        how_many[h]=many ;
    }

}

else
{
    cout << "склад полон, добавление детали невозможно";
}
if (name_number[h]==0)
{
    how_many[h]=0;
    cout << "\n Ошибка ввода данных. Данные не были добавлены! Пожалуйста проверьте правильность ввода и повторите ввод \n";
}

}

cout << "\nжелаете выйти из цикла ввода данных?\n";
cin >> iq;

}
cout << "\nцикл вывода информации\n";
for (int b=0; b!=10; b++)
{
    cout << b << "\t";
    io[0]=deta1[b];
    if (name_number[b]==1)
```

```
{
    io[1]=marx[0];
    io[2]=marx[1];
    io[3]=marx[2];
    io[4]=tipe[b];
}
if (name_number[b]==2)
{
    io[1]=marx[4];
    io[2]=marx[5];
    io[3]=marx[6];
    io[4]=tipe[b];
}
if (name_number[b]==3)
{
    io[1]=marx[9];
    io[2]=marx[10];
    io[3]=marx[11];
    io[4]=tipe[b];
}
if (how_many[b]>0)    //если ячейка склада пустая печатать NONE
{
    for (int o=0; o!=8; o++)
    {
        cout << io[o];
    }
    cout <<how_many[b] <<"\n";
}
else
    cout<< "NONE\n";
}
// ВТОРОЙ ЭТАП ЗАДАНИЯ проверка на наличие товара.
cout << "\n введите интересующий вас товар \n ";
cin >> iq;
    g=0;
while (g != 10)    // цикл проверки существует ли данная деталь на складе
{
    io[0]=detal[g];
```

```
if (name_number[g]==1) // name_number – массив кодировки марки автомобиля bmw -1, audi - 2, kia-3
{
    io[1]=marx[0];
    io[2]=marx[1];    // генерация маркировки если бумер
    io[3]=marx[2];
    io[4]=tipe[g];
}
if (name_number[g]==2)
{
    io[1]=marx[4];
    io[2]=marx[5];
    io[3]=marx[6];    // генерация маркировки если волга
    io[4]=tipe[g];
}
if (name_number[g]==3)
{
    io[1]=marx[9];
    io[2]=marx[10];
    io[3]=marx[11];    // генерация маркировки если кия
    io[4]=tipe[g];
}

if (iq[0]==io[0] && iq[1]==io[1] && iq[2]==io[2] && iq[3]==io[3] && iq[4]==io[4])
{
    cout << "\n данный товар находится в ячейке " << g << "\n его количество и маркировка:";
    for (int o=0; o!=8; o++)
    {
        if (io[o]!=marx[3])    // пробелы в маркерровке не допустимы
        {
            cout << io[o];
        }
    }
    if(how_many[g]<10)
        cout <<"00";
    if(how_many[g]<100 && how_many[g]>10)
        cout <<"0";
}
```

```
        cout << how_many[g] << "\n";
        g=10;
        yesorno=1;
    }else
    {
        g++;
        yesorno=0;
    }
}
if (g==10 && yesorno==0)
    cout << "интересующего для вас товара на складе нет\n";
cout << "желаете продолжить? если нет введите exit\n" ;
cin >> iq;
}
}
```