


```

}
}

```

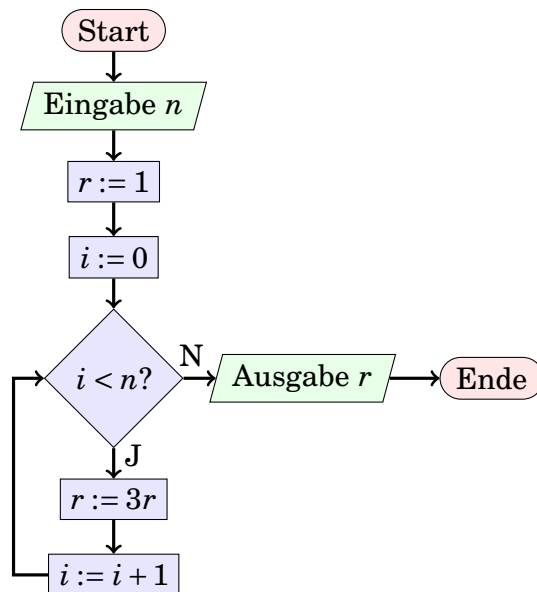
```

6. Person myFriends[] = // [] fehlte
{
    {"Berta", 42}, // Komma statt Semikolon
    {"Karlhei", 43} // nur acht Zeichen inkl. Null am Ende
};

```

7. `c1`

`c1j`: J, N hinzugefügt



```

8. if(myFriends[0].name[4] == 'a')
{
    // ...
}
if(myFriends[1].name[1] == 'a')
{
    // ...
}
if(myFriends[0].name[0] - 'B' + 'a' == 'a')
{
    // ...
}

```

und viele weitere Möglichkeiten.

9. Hier läuft `int * int` über. Mögliche Abhilfe:

```

long product(int a, int b)
{
    return a * (long)b;
}

```

10. `bool hasNext(void)`

```

{
    return count > 0;
}

int pop(void)
{
    count--;
    return stack[count];
}

```

11. $13 + 3n + \frac{1}{n^2+2}$ ist asymptotisch $O(n)$ und $O(n^2)$ und $O(n^3)$. (Begründung war nicht gefragt: Das ist $O(n)$, weil $13 + 3n + \frac{1}{n^2+2} \leq 4n$ für $n \geq 14$. Jede Funktion in $O(n)$ ist automatisch in $O(n^2)$; jede Funktion in $O(n^2)$ ist automatisch in $O(n^3)$.)

12. Zum Beispiel:

```

int f(int x)
{
    if(x < 0)
    {
        return -x;
    }
    return x;
}

```

Oder (aber eine Lösung reicht):

```

int f(int x)
{
    int result = x;
    if(x < 0)
    {
        result = -result;
    }
    return result;
}

```