

Псевдокод



Типове конструкции с код

Клиф свърши чудесна работа, запознавайки ви с Visual Studio 2010 . Какво беше усещането ви, когато писахте първата си програма? Знаме, че все още не сте готови да пишете много код, но и това ще стане. Сега ще ви покажа как да планирате и организирате програмата си преди да започнете да пишете код.

Знаете ли как можете да напишете компютърна програма без да знаете език за програмиране? Много програмисти първо пишат програмите си на "псевдокод", след което ги преобразуват до реален език за програмиране. Това им помага да дефинират логическата последователност на програмата и това, което тя трябва да прави. Псевдокодът се състои от прости изречения на английски (или на родния ви език), които определят действието на програмата. След като сте написали псевдокода написването на истинския код е лесно. Псевдокодът дефинира структурата на програмата.

Преди да ви покажа как да използвате псевдокод, искам да ви запозная с някои основни типове конструкции, изграждащи компютърната програма. Всички езици за програмиране използват еднакви базови типове конструкции с код, които се различават по своя синтаксис. Комбинирайки ги, вие можете да пишете доста сложни програми.

Спомняте ли си реда с код, който написахте за бутона в програмата Hello World?

Той изглеждаше по следния начин:

```
MessageBox.Show("Hello World.")
```

Този код показва информация на потребителя. Това е основната конструкция за визуализиране на прозорец с предупреждение или съобщение. Всички езици за програмиране притежават подобна конструкция за извеждане на информация за потребителя, която обаче е с различен синтаксис. Съществуват и други такива конструкции.

Ето някои примери на други типове конструкции, поддържани от езиците за програмиране.



Променливите се използват за съхраняване на данни като числа и текст. Те пазят информация и резултати от изчисления.

Операторите за присвояване установяват какво се съхранява в променливите. Стойността може да бъде число, текст, друга променлива или калкулирана стойност.

Операторите за сравнение се използват за сравняване на стойности или променливи с числа или други стойности или променливи. Те задават въпрос отговорът, на който е "true" или "false".

Условните оператори се използват за реализиране на избор. Изборът се основава на сравнение. Ако отговорът от сравнението е "true", се избира първият вариант. Ако той е "false", се избира вторият вариант. Условните оператори често се наричат оператори за разклонение, тъй като програмата продължава изпълнението си в една или друга посока.

Операторите за цикъл се използват за многократно извършване на определени действия. Те намаляват обема на кода, който трябва да напишете, когато програмата повтаря едни и същи действия.

Като се учите да програмирате, ще учите и как да използвате тези основни конструкции, за да накарате програмата си да прави точно това, което искате. Изучавайки Visual Basic (или други езици за програмиране) ваша задача ще бъде да научите определен синтаксис, дефиниран от езика за програмиране.

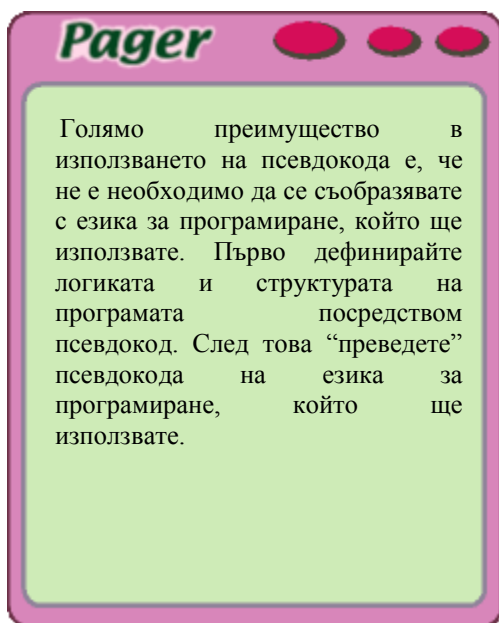


Добре дошли тук тази вечер! Знаете, че бях непоправим самохвалко. Ако съм в града и ако е отворено в полунощ, ще ме откриете тук на тази сцена.



Какво е псевдокод?

Както споменахме преди, програмистите често пишат своите програми на псевдокод. Представката псевдо- означава “вместо”. Например, някои автори използват псевдоними вместо истинските си имена. Псевдокодът не е написан на истински език за програмиране. Той се състои от прости изречения, които описват какво ще прави програмата. Можете да пишете псевдокод на английски, испански, латински или на някакъв друг език, защото компютърът никога няма да го прочете или разбере. Псевдокодът ви трябва да е достатъчно ясен за вас и другите програмисти, защото той осигурява инструкции за написването на истинската програма. При написването на псевдокода можете да използвате различните типове основни конструкции -- присвояване, сравнение, условие и т.н.





Ще ви покажа пример за това как да използвате псевдокод при създаването на програма. Да кажем, че искам да напиша програма за пресмятане средния разход на бензина, изразходен при пътуването ни, в долари за галон. Какво трябва да прави програмата? За да отговоря на този въпрос, написах програмата като първо използвах псевдокод. Ето стъпките, които моята програма трябва да направи, за да пресметне средната стойност на бензина:

1. Събира платените за бензина пари и присвоява получената стойност на променлива (total dollars)
2. Събира изразходените галони бензин и присвоява получената стойност на втора променлива (total gallons)
3. Разделя първата променлива (total dollars) на втората (total gallons) и присвоява резултата на трета променлива (dollars per gallon).
4. Показва съобщение със средния разход в долари за галон.

Псевдокодът ясно показва какво трябва да прави програмата. Сега мога да напиша код, за да завърша скицираната с псевдокод задача.

Ще ви дам друг пример с повече стъпки. Той може да се използва в програмируем робот за подмяна на спукани гуми.

1. Определете коя гума е спукана. (оператор за сравнение)
2. Определете размера на спуканата гума (оператор за сравнение)
3. Ако имате резервна гума със същия размер продължете
4. или вземете резервна гума от магазина. (оператор за сравнение и условие)
5. Повдигнете микробуса с хидравличен крик.
6. Махнете тасовете с отвертка, за да открийте болтовете.
7. Развивайте болтовете с гаечен ключ докато отвиете всички. (оператор за цикъл)
8. Махнете спуканата гума.
9. Сложете новата гума.
10. Завивайте болтовете с гаечен ключ докато завиете всички. (оператор за цикъл)
11. Поставете тасовете и ги ударете, за да се наместят.
12. Махнете хидравличния крик.



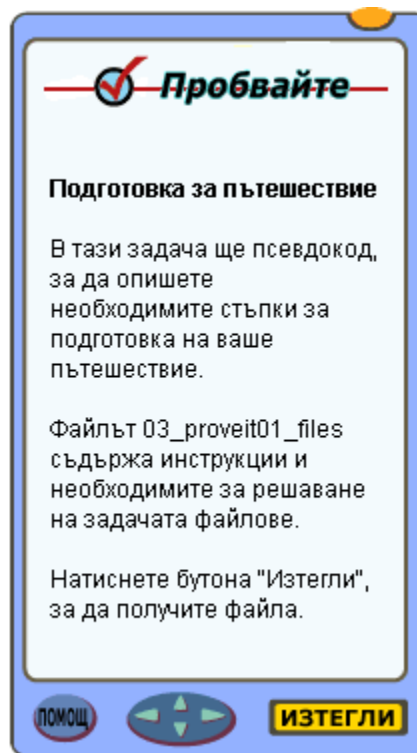
В псевдокода по-горе има комбинация от оператори за присвояване, условие, сравнение и цикли. Например, трябва да съхраните размера на спуканата гума посредством оператор за присвояване. Трябва да сравните размера на спуканата гума с тази от магазина посредством оператор за сравнение. Операторът за условие се използва с оператора за сравнение. Ако притежавате гума с необходимия размер продължавате, в противен случай отивате в магазина за гуми. Отвиването и завиването на болтовете може да се реализира с оператор за циклична обработка. Всяка една от обработките трябва да бъде повторена определен брой пъти, докато всички болтове са отвити или всички болтове са завити.



Защо не опитате да напишете някакъв псевдокод като използвате джобния ми компютър?

03 Пробвайте 01 Подготовка за пътешествие

Напишете псевдокод, за да опишете необходимите стъпки за подготовка на ваше пътешествие. Вашето решение трябва да има най-малко седем стъпки. Ако всяка стъпка започва с оператора `if`, то вие не сте успели да създадете псевдокод.

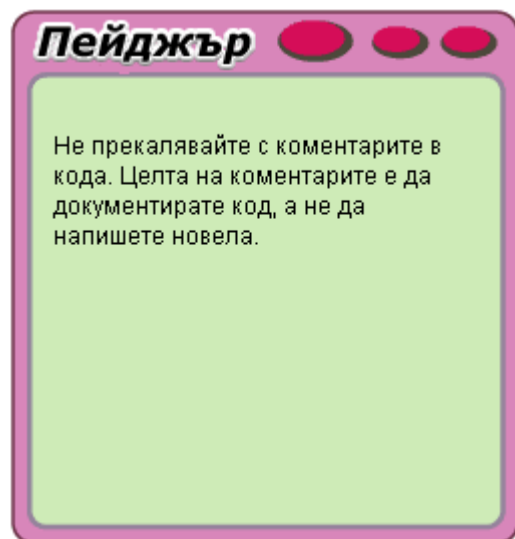




Коментари

Съвременните езици за програмиране, включени в Visual Studio, ви позволяват да добавяте коментари към кода. Коментарите не се компилират като част от програмата и не се изпълняват при нейното стартиране. Те са полезни за документиране и по-късно за разбиране на кода от вас и останалите програмисти.

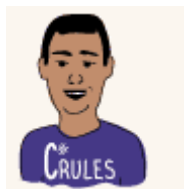
Вие винаги трябва да коментирате кода си. Коментарите са белег за добро програмиране. Добрият програмист отделя време, за да разясни и документира кода, който е написал. Трябва да добавяте коментари, когато променяте кода или добавяте функционалност. Също така коментирайте частите от вашата програма, които са трудно разбираеми или изпълняват особено тежки или сложни изчисления.





Във Visual Basic всеки ред от коментара започва с апостроф ('). Всеки ред от кода, който започва с апостроф се интерпретира като коментар и не се компилира и изпълнява при стартирането на програмата. Ето пример за коментар в Visual Basic

```
'Добавяне на функционалност за пресмятане на  
средния изминат път в мили за галон.
```



Хей, коментарите във Visual Basic започват с единичен апостроф, но в C# се използват двойка наклонени черти (//).

Например,

```
//Добавяне на функционалност за пресмятане  
на средния изминат път в мили за галон.
```



Коментарите имат и друга важна функция освен документирането на кода. Тъй като те не се изпълняват, могат да бъдат използвани като средство за съхраняване на конкретни редове код за изпълнение. Това се нарича “коментиран” код. Коментираният код може да се използва за откриване на грешки. Ако е възникнала грешка, можете да коментирате редове или цели блокове от код, докато грешката спре да се появява. Последният код, който сте коментирали вероятно съдържа грешката, която възниква. Много удобно, нали?

Сега ще ви дам пример за коментиран код. Забележете, че редовете с код са три. Вторият ред е коментиран посредством единичен апостроф, поставен в началото му. Той няма да се изпълни!

```
MessageBox.Show("I am from"  
'MessageBox.Show(" a cool town called")  
MessageBox.Show(" San Francisco.")
```



Псевдокод като коментар

А сега ще ви покажа още едно удивително приложение на коментара. Сигурно сте нетърпеливи! Можете да използвате коментарите, за да описвате и планирате вашата програма. Лесно е! Вие вече сте писали псевдокод, който описва какво прави програмата ви. Възползвайте се от това преимущество. Копирайте псевдокода в програмата си и поставете единичен апостроф пред всеки ред, за да го преобразувате в коментар. Под всеки ред с псевдокод напишете конкретен Visual Basic код и програмата ви не само, че ще бъде завършена, но и коментирана!

Ето пример, базиран на псевдокода, който написах за пресмятане на средния разход за бензин по време на пътуването ни. Забележете, че псевдокодът представлява коментар.

'Съберете всички дадени за бензин пари и
получената стойност присвоете на
променлива(total dollars)

```
Dim totalDollars As Double  
totalDollars = 88.76 + 100.8 + 120.3
```

'Съберете броя на изразходените галони с
бензин и присвоете получената стойност на втора
променлива (total gallons)

```
Dim totalGallons As Double  
totalGallons = 49.3 + 55.5 + 46.3
```

'Разделете първата променлива (total dollars)
на втората (total gallons) и присвоете резултата
на трета променлива (dollars per gallon).

```
Dim DPerGallon As Double  
DPerGallon = totalDollars / totalGallons
```

'Визуализирайте съобщение, което показва
средния разход в долари за галон.

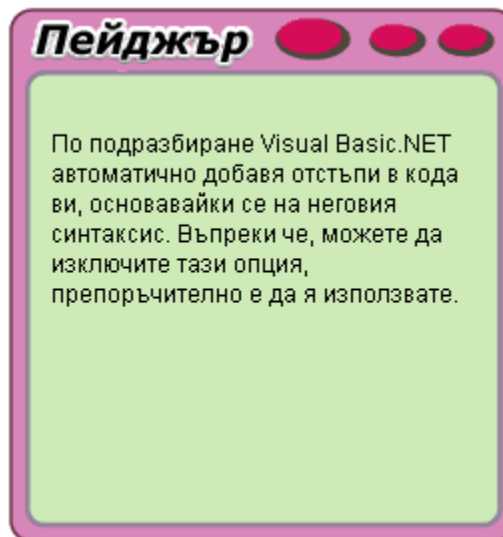
```
MessageBox.Show (dollarsPerGallon)
```



Отстъпи и интервали

Макар и да не разбирате целия Visual Basic код по-горе, искам да насоча вниманието ви към някои неща. Преди всичко се вижда, че коментарите започват с единичен апостроф. Също така цвета на коментирания текст е зелен. Това го прави лесно различим от кода. Второ, забележете, че преди всеки коментар е добавен празен ред. Празните редове разделят коментарите и кода на групи според функциите им. Те правят програмата ви логически ясна и лесна за разбиране.

В повечето случаи Visual Basic автоматично добавя интервали и отстъпи, за да направи кода ви по-четлив. Ако желаете, допълнително можете да добавите интервали и отстъпи. Ето и моята гледна точка в това отношение – използвайте ги, за да направите кода си четлив. Visual Basic и другите езици от Visual Studio игнорират интервалите и отстъпите при компилацията на програмата. Докато Visual Basic не се нуждае от тях, то със сигурност те са полезни за човека, който чете кода ви! (Вие сте човекът, който ще има най-голяма полза.)





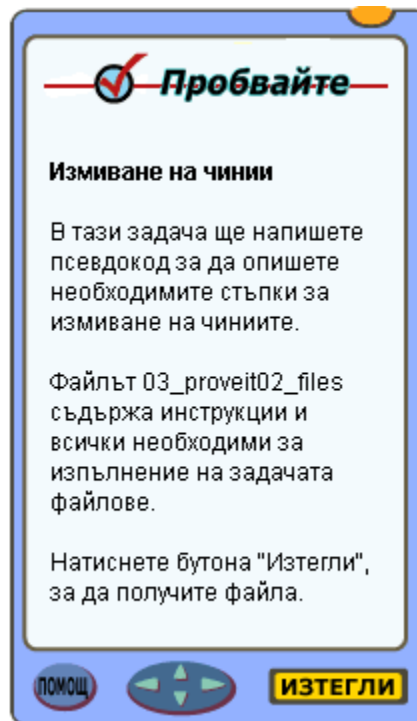
03 Пробвайте 02 Измиване на чинии

Получих друга задача на джобния си компютър, която трябва да направите след като измиете чиниите!

Караваните за къмпинг имат кухня, която се използва, когато наблизко няма подходящо заведение за хранене. Готвенето означава натрупване на чинии за измиване, караваните нямат автоматични миялни машини.

Напишете псевдокод, за да опишете необходимите стъпки за измиване на чиниите. Ако имате личен опит в това, можете да направите някакво проучване.

Вашето решение трябва да има най-малко седем стъпки.





03 Можете ли?

Аквариум

Вижте аквариума с риби в магазина за домашни любимци?

Пребройте рибите от всеки цвят и изчислете процентното им съотношение спрямо общия им брой.

В действителност притежавам програма, която прави това.

Намерете файл с име FishTank.sln. Щракнете два пъти върху него, за да го отворите с Visual Studio. Отворете страницата с изходния код.

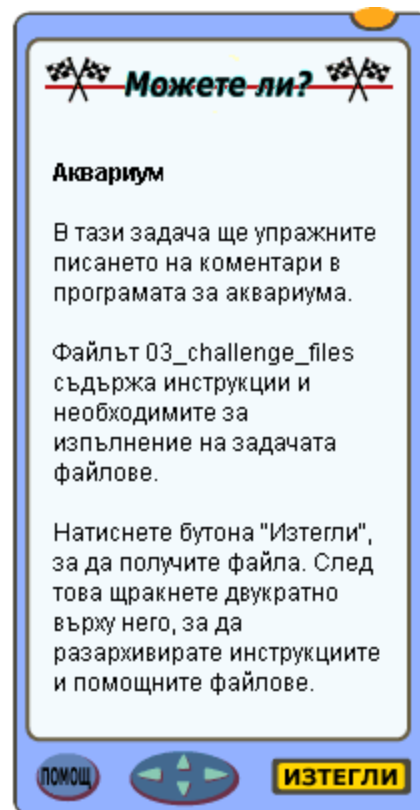
Вашата задача е да разгледате страницата с изходния код и да копирате всички коментари в текстов файл.

Изглежда ли резултатът като псевдокод?

Покажете списъка с коментарите на вашия учител.

Продължение / Обобщение


Прекалено много ли са коментарите? Прекалено малко? Защо?





Пристигане в Барселона

Тъй като пристигнахме в Барселона, Испания трябва да направите един малък тест. Той може да ви помогне да отговорите на въпросите, които ще ви зададе Комисията по състезанията.



Проверка на знанията

НАПРАВЕТЕ ТЕСТА ОТНОВО

1 Какво се използва, за да се изпълни нещо многократно?

- ☐ А. Изрази за сравнение
- ☐ В. Изрази за цикли
- ☐ С. Изрази за присвояване

2 Псевдокодът е:

- ☐ А. Истински програмен език
- ☐ В. Прости изречения, които описват какво ще прави програмата
- ☐ С. Име на автора

3 Програмните грешки могат да се отстранят чрез:

- ☐ А. Коментари
- ☐ В. Отстъпи
- ☐ С. И двете

4 Условните изрази:

- ☐ А. Осигуряват сравнение
- ☐ В. Предизвикат едно действие, когато условието е истина и друго - ако е лъжа
- ☐ С. И двете