

BMB201 - Nesneye Dayali Programlama Dersi 2. Odev Aciklamalari

V1: Son Degisiklik Tarihi: 25 Ekim 2016 10:20

Odevi Hazirlayan: Ars. Gor. Guvenc Usanmaz

Bu odevde Bosluk Doldurma isimli oyun icin kod yazacaksiniz. Bu dokumanda ilk olarak Bosluk Doldurma oyunu tanitilmektedir. Daha sonra bu oyunun dogru kodlanmis halinin ornek terminal ciktilari verilmistir. En son kisimda bu odev kapsaminda yapmaniz gereken kodlamalar anlatilmektedir.

1 Odev Kapsami

Bu odev asagida listenen kavramlara ve araclara olan hakimiyetinizi olcmeyi amaclamaktadır.

- Fonksiyonlar (function header, arguman, parametre, lokal degiskenler, return, fonksiyon tanimi (function declaration), fonksiyon cagrilmasi (function call))
- Degisken turlerinin kullanimi (int, char, String, bool)
- Static degiskenler
- Tek boyutlu dizilerin kullanimi
- Aritmek, mantiksal (logical) ve dogruluk (conditional) operatorleri
- Kosullu dallanmalar (if, else, if else)
- Dongu yapıları (for, while, do while, foreach) ve ici ice donguler (nested loops)
- Bit duzeyinde islem yapan operatorler (Bitwise operators)
- Console.WriteLine() ve Console.ReadLine() fonksiyonlarının kullanimi ile kullanıcı etkilesimi
- Convert sinifi kullanimi ile degisken turunu degistirme
- Random sinifi ile rastgele sayi uretimi

En alttaki turuncu renkle yazilmis konulari bilmeden de bu odev yapilabilir. Bu konulari bilmeniz size verilmiş kısmi oyun kodunu tam olarak anlayabilmeniz icin gerekmektedir.

2 Bosluk Doldurmaca Oyunu

- Bosluk doldurmaca oyunu 1xn lik bir oyun tahtasi uzerinde k adet piyon ile oynanir. n 10 ile 32 arasinda bir deger alabilir. k 2 ile 26 arasinda bir deger alabilir. k sayisinin degeri (n-1)'den buyuk olamaz.
- k adet piyon 1xn lik oyun tahtasina oyunun basinda rastgele yerlestirilir.
- Her bir piyon Ingilizce Alfabeki bir buyuk harf ile temsil edilmektedir. Oyun tahtasindaki bos kareler * karekteri ile temsil edilmektedir
- Oyun k = 5 piyon ile oynaniyorsa, piyonlari A, B, C, D ve E harfleri temsil etmektedir. Oyun k = 8 piyon ile oynaniyorsa, piyonlari A, B, C, D, E, F, G, H harfleri temsil etmektedir.

- Asagida oyun baslangicinda, oyun tahtasinin belli k ve n degerleri icin olasi goruntusu verilmistir.

Ornek Tahta 1: $n = 10$, $k = 5$ icin oyun tahtasi baslangicta su gorunumde olabilir:

A	*	B	C	*	*	*	D	*	E
---	---	---	---	---	---	---	---	---	---

Ornek Tahta 2: $n = 10$, $k = 7$ icin oyun tahtasi baslangicta su gorunumde olabilir:

A	B	*	*	C	*	D	E	F	G
---	---	---	---	---	---	---	---	---	---

Ornek Tahta 3: $n = 14$, $k = 6$ icin oyun tahtasi baslangicta su gorunumde olabilir:

*	*	A	*	*	B	C	*	D	*	E	*	F	*
---	---	---	---	---	---	---	---	---	---	---	---	---	---

Ornek Tahta 4: $n = 14$, $k = 9$ icin oyun tahtasi baslangicta su gorunumde olabilir:

A	*	*	B	C	*	D	*	E	F	G	*	H	I
---	---	---	---	---	---	---	---	---	---	---	---	---	---

- Bu oyun iki kisi ile oynanir. Piyonlar oyun tahtasina yerlestirildikten sonra oyuncular sıra ile hamlelerini yaparlar.
- Hamle olarak oyuncu bir piyon secer, sectigi piyonu bulundugu kareden daha soldaki bos bir kareye kaydirtir. Piyon kendinden soldaki piyonların soluna hareket edemez. Diger bir ifadeyle hamle sonrasi piyonlar oyun tahtasindaki alfabetik dizilimini korumalidir.
- Yukarida verilen ornek oyun tahtalari icin gecerli ve gecersiz olan birkac ornek hamle asagida verilmektedir.

- Ornek Tahta 1’de B piyonu 1 birim sola kaydirilabilir. Bu hamle sonucu oyun tahtasi su gorunumu alacaktır:

A	B	*	C	*	*	*	D	*	E
---	---	---	---	---	---	---	---	---	---

- Ornek Tahta 1’de E piyonunu 3 birim sola kaydirilamaz. E piyonu kaydirma islemi sonucu D piyonun onune gecegi icin bu hamle gecersiz bir hamledir.
- Ornek Tahta 3’de B piyonu 2 birim sola kaydirilabilir. Bu hamle sonucu oyun tahtasi su gorunumu alacaktır:

*	*	A	B	*	*	C	*	D	*	E	*	F	*
---	---	---	---	---	---	---	---	---	---	---	---	---	---

- Oyunculardan birinin hamlesinin sonucu oyun tahtasindaki butun piyonlar aralarinda bos bir kare (* ile temsil edilen kare) olmadan dizildiginde oyun sona erer ve son hamleyi yapan oyuncu oyunu kazanir.
- Ornegin Ornek Tablo 3 yapilan hamleler sonucu asagidaki gorunumu kazandiysa oyun sona erer:

*	*	A	B	C	D	E	F	*	*	*	*	*	*
---	---	---	---	---	---	---	---	---	---	---	---	---	---

- Bu oyunun dogru kodlanmis halinin ornek terminal ciktilari oyun_terminal_output.txt isimli dosyada verilmistir. Bu dosyaya ADYS’deki Odevler sayfasindan ulasabilirsiniz. Bu dosyadaki ciktilara bakmak oyunun nasil oynandigini kavramanizi ve odev icin gerekli kodlamalari yapmanizi kolaylastirabilir.

3 Odev Aciklamalari

ADYS uzerinden ikinci odeve ait zip uzantili dosyayi indiriniz. Ilgili dosya icindeki Solution klasoru altindaki Program.cs dosyasini bulunuz. Oyuna dair tum kodlar bu dosya icinde yazilmalidir. Program.cs icinde asagidaki fonksiyonlar bulunmaktadır.

- **public static void CreateBoard(int boardLen, int pawnNumber)**
- **public static bool MovePawn(char pawnSymbol, int step)**
- **public static bool GameOver()**
- **public static void PrintBoard()**
- **public static void Main(string[] args)**

CreateBoard(int boardLen, int pawnNumber), PrintBoard() ile Main(string[] args) fonksiyonlari icine gerekli kodlar sizin icin yazilmistir. Bu fonksiyonlarin kodlarinda herhangi bir degisiklik yapmayiniz. Bu fonksiyonlar icin yazilmis kodlari inceleyin ve fonksiyonlarin icinde adim adim hangi islemler yapildigini anlamaya calisin. movePawn ve GameOver fonksiyonlarin ici ise dogru sekilde kodlanmamistir. Bu fonksiyonlarin iclerini bu fonksiyonlara istenilen islevi kazandiracak sekilde kodlamaniz gerekmektedir. Bu fonksiyonlarin basliklarinda (function header) herhangi bir degisiklik yapmayiniz.

Program.cs altinda 3 farkli static degisken tanimlanmistir. Bu tanimlari su sekildedir:

- **static char[] board;**
- **static int boardLen;**
- **static int pawnNumber;**

Bu degisken tanimlarnin bulundugu satirlarda herhangi bir degisiklik yapmayiniz.

Bu degiskenlerin islevleri su sekildedir:

- board dizisinde oyun tahtasindaki piyonlarn konum bilgisi tutulur
- boardLen degiskeninde oyun tahtasinin uzunluk (kac kareden olustugu) degeri tutulur.
- pawnNumber degiskeninde oyun tahtasinda bulunan piyon sayisinin degeri tutulur.

Bu degiskenler CreateBoard ve Main fonksiyonlari icinde kullanilmaktadir. Benzer sekilde iclerini dolduracaginiz movePawn ve GameOver fonksiyonlarnin bu degiskenleri kullanmasi gerekebilir.

Fonksiyonlarn static degisken kullanmalari cogu zaman tavsiye edilmemektedir. Ne var ki bu odev fonksiyonlarniznin static degisken kullanmasini gerektirmektedir.

Kodunuzda bulunan static degiskenler (board, boardLen, pawnNumber) disinda bir static degisken tanimi yapmayiniz.

Asagida Program.cs altinda bulunan 5 fonksiyonun islevleri aciklanmaktadir.

3.1 public static void CreateBoard(int boardLen, int pawnNumber)

- CreateBoard fonksiyonu BoardLen uzunlukta oyun tahtasina pawnNumber sayida piyonu rastgele yerlestirir.
- Oyun tahtasindaki piyonlara ait bilgiler board degiskenine yazilir.
- boardLen parametresi 32'den buyuk bir deger alirsa, 32 kare uzunlugunda oyun tahtasi olusturulur.
- boardLen parametresi 10'dan kucuk bir deger alirsa, 10 kare uzunlugunda oyun tahtasi olusturulur.
- pawnNumber parametresi boardLen'den bir eksik veya daha buyuk bir deger alirsa, pawnNumber degeri boardLen - 1 olarak degistirilir.

- pawnNumber parametresi 26'dan büyük bir deger alirsa, pawnNumber degeri 26 olarak degistirilir.
- Bu fonksiyon herhangi bir deger dondurmez.
- Bu fonksiyon icinde piyonlari oyun tahtasina yerlestirmek icin bit bazinda calisan operatorler (&, |, ^) kullanilmistir. Bu operatorlerin gectigi satirlarda amaclanan islemlerin neler oldugunu anlamaya calisiniz.

3.2 public static bool MovePawn(char pawnSymbol, int step)

- Bu fonksiyon pawnSymbol harfli piyonu step degeri kadar sola kaydirmaya calisir.
- Oyun tahtasinda pawnSymbol harfli piyon yoksa fonksiyon false dondurur.
- Oyun tahtasinda pawnSymbol isimli piyonu step degeri kadar sola kaydirmek oyun kurallarini ihlal ediyorsa fonksiyon false dondurur.
- Bu fonksiyon false dondurecegi durumlarda oncelikli olarak (false dondurmeden) ekrana yapilmak istenen hamlenin gecersiz bir hamle olduguna dair bir uyari yazdirmalidir.
- Oyun tahtasinda pawnSymbol isimli piyonu step degeri kadar sola kaydirmek oyun kurallarini ihlal etmiyorsa bu fonksiyon pawnSymbol isimli piyonun step degeri kadar kaydirildigi bilgisini board dizisine isler. Board dizisi guncellendikten sonra fonksiyon true dondurur.

3.3 public static bool GameOver()

- Bu fonksiyon board dizisindeki butun piyonlarin aralarinda bosluk olmadan ard arda dizilip dizilmedigini inceler.
- board dizisinde butun piyonlar aralarinda bosluk olmadan ard arda siralanmissa fonksiyon true dondurur.
- board dizisinde butun piyonlar aralarinda bosluk olmadan ard arda siralanamamissa fonksiyon false dondurur.

3.4 public static void PrintBoard()

- Bu fonksiyon board dizisindeki piyonlarin konum bilgilerini okur ve bu dizinin icerigine gore oyun tahtasini ekrana cizer.
- Bu fonksiyon herhangi bir deger dondurmez.

3.5 public static void Main(string[] args)

- Bu fonksiyon icinde oncelikli olarak oyun tahtasinin ve oyun tahtasinda istenen piyon sayisinin bilgisi kullanicidan alinir. Bu isteklere uygun bir sekilde olusturulan tahtaya piyonlar rastgele yerlestirilir.
- Bu fonksiyon daha sonra oyunu oynayacak iki oyuncunun isimlerini kullanicidan alir.
- Oyun sonlanana kadar oyunculardan yapmak istedikleri hamlelere dair bilgiler alinir. Yapilan her gecerli hamle sonrasi oyun tahtasinin yeni goruntusu ekranda gosterilir.

ADYS'den indireceginiz kod derlenip calistirabileceginiz bir durumdadir. Ne var ki MovePawn ve GameOver fonksiyonlari dogru calismadigi icin oyun dogru bir sekilde calismamaktadir.

MovePawn fonksiyonu yaklasik 40 satirlik kod, GameOver fonksiyonu ise 20 satirlik kod ile yazilabilmektedir. Kurdugunuz algoritmaya ve kod yazma stilinize bagli olarak bu fonksiyonlari kac satirlik kodla yazabileceginiz degiskenlik gosterebilir.

4 Odev Gonderme (Submission)

- Bu odevde ogrencilerin bireysel calismasi beklenmektedir. Grup olarak odev sorularini cevaplamayiniz ve grup gonderiminde bulunmayiniz.
- Bu odevden dersi alan butun ogrenciler sorumludur.
- Olusturdugunuz C# projesindeki tum dosyalari gondermeyiniz. Sadece yazdiginiz fonksiyonların bulunduğu Program.cs dosyasini ADYS üzerinden gonderiniz.
- Yazdiginiz tum kod Program.cs dosyasinda bulunsun.
- Yazdiginiz kodda yeni bir static degisken tanimi yapmayiniz.
- Fonksiyon basliklarında (function header) herhangi bir degisiklik yapmayiniz.
- Kodlarınızda hazır fonksiyon (API fonksiyonu, kutuphane fonksiyonu vs.) kullanmayiniz.
- Bu odev için Program.cs içinde mumkunse var olan fonksiyonlar haricinde yeni bir fonksiyon tanimi yapmayiniz.
- Gonderdiginiz kodun derleme (compile error) ve calisma zamanı (runtime error) hatası icermediginden emin olunuz.
- Program.cs dosyasinin ust satirlarında bulunan ISIM, SOYISIM, OGRENCI_NO, EMAIL kisimlerini kendinize ait bilgilere degistirmeyi unutmayiniz!
- EMAIL kismina duzenli kontrol ettiginiz bir email adresi yaziniz. Gonderdiginiz odevle ilgili bir sorun olması durumunda email araciligi ile sizinle iletisime gecilecektir. Ayrica verilen odevlerle ilgili bir degisiklik olması durumunda bildirimis oldugunuz email adresleri üzerinden duyuru yapılacaktır.
- Odev kodları otomatik programlar tarafından degerlendirilebilir. Bu durumda bir fonksiyon için yazdiginiz kod mantikli da olsa istenileni yapmiyorsa o fonksiyon için puan alamayabilirsiniz.
- Gonderdiginiz odevler kopya programından gecirilebilir ve kopya olduğu tespit edilen odevler puanlanmaz. Kopya odev gonderen ogrencilerin diger odevleri de notlandırılmayabilir!
- Rastgele belirlenen bazı ogrencilerin veya kopya odev gondermis olabilecegi konusunda kusku duyulan odev gonderen ogrencilerin yazdikları kodları sozlu olarak aciklamaları istenebilir.
- İlan edilen son gonderme tarihinden (deadline) sonra gonderilen odevler degerlendirilmeyecektir!
- Uygulama derslerinde verilen odevle ilgili sozlu aciklama yapılacaktır. Dersi ustten alanlar da laboratuvar da yer olması durumunda odev aciklamalarını dinlemek için derse gelebilirler.
- Odev ile ilgili sorularınızı uygulama derslerinde sorabilirsiniz.
- Odevle ilgili sorularınız için gusanmaz@nku.edu.tr adresine email gonderebilirsiniz.
- Bu dokuman bir takım eksiklikler veya hatalar icerdiginin fark edilmesi durumunda guncellenebilir. ADYS üzerinden dokumanın son haline odev dosyalarını yeniden indirerek ulasabilirsiniz. Dokumanın guncellenmesi durumunda basligın altındaki dokuman versiyon numarası (Su anki versiyon numarası v1) degistirilecektir. Dokumanda degisiklik yapılan kisimler kirmizi renkte yazılacaktır.

5 Son Gonderim Tarihi (Deadline)

Odevin son gonderim tarihi **14 Kasim 2016, Pazartesi 23:59** dur.