

DERS NO:

ÖĞRENCİ İSMİ:

SINAV BİTİRME SAATİ (GÖZETMEN
TARAFINDAN DOLDURULACAKTIR.)

_____ : _____

BİL141 ARA SINAV II - 14.07.2019

UYARI: SORULARI DİKKATLİCE OKUYUNUZ. SINAV ESNASINDA SORU ALINAMAMAKTADIR.

SINAV SONRASINDA BU KÂĞIDI, YUKARIDAKİ BİLGİLERİNİZİ YAZMIŞ VE AŞAĞIDAKİ DÜRÜSTLÜK BEYANINIZI İMZALAMIŞ OLARAK GÖZETMENİNİZE TESLİM EDİNİZ.

- Sınav esnasında soruları anlamakta yaşadığınız sorunları, bilgisayarınızda yaşadığınız sorunları vs. lütfen **kendi başınıza** çözünüz. GÖZETMENLERİN GÖREVİ SINAVI GÖZETLEMektir.
- Bilgisayarınız **uzun süreli tıkalı kalırsa** kapatma düğmesine 3-5 saniye basılı tutarak kapatıp sonra yeniden açınız. Böyle bir duruma karşı önlem olarak kodunuzu **SIK SIK KAYDEDİNİZ.**

SINAVDA UYULMASI GEREKEN TEMEL KURALLAR:

- Sınav esnasında yanınızda getirdiğiniz, el yazınızla yazılmış ipucu kâğıdının ve DevCpp'ın kod parçaları özelliğinin haricinde herhangi bir materyalden/kaynaktan/kişiden **yardım alamazsınız.**
- Sınav süresince(kod gönderim kısmı hariç) DevCpp haricinde herhangi bir program kullanamazsınız. DevCpp'da da **SADECE KOD YAZDIĞINIZ SEKMELER(iki soru olduğu için sadece iki sekme)** açık olmalıdır. Herhangi başka bir kod, yardım kutucuğu, ipucu kutucuğu, slayt vs. açamazsınız.
- Sınav süresine bilgisayarın internet erişimini açmak, son sayfada adresi verilen siteye girmek, bu sitedeki forma kodlarını aktarmak ve formu başarılı şekilde göndermek **DÂHİLDİR.** Sınavın süresi bittikten sonra kod gönderimi yapılmamalıdır.
- Sınav esnasında gözetmenin izni olmaksızın sınav salonundan ayrılamazsınız. Sağlık raporuyla belgelenmiş sağlık sorunları haricince lavabo vb. taleplerle öğrencinin sınavı geçici süreli terk etmesine **izin verilememektedir.**
- Sınavın **ilk yarım saati** sınav salonundan ayrılamazsınız.

Sınav esnasında herhangi bir kopya girişiminde bulunmadığımı/bulunmayacağımı, yukarıdaki kuralları ihlal etmediğimi/etmeyeceğimi ve böyle bir girişime/kural ihlaline şahit olduysam gözetmenime bildirdiğimi/bildireceğimi beyan ederim.

Lütfen imzalayınız:

KODLAMAYA DAİR UYARILAR (LÜTFEN DİKKATLE OKUYUNUZ):

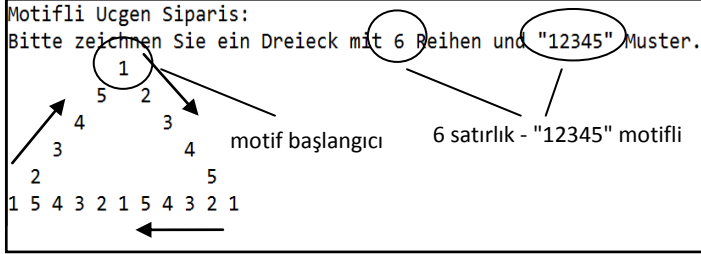
- Çözümlerinizin sadece istendiği şekilde çalışıyor olması yeterli OLMAYABİLİR! **KÖTÜ PROGRAMCI ALIŞKANLIKLARI** (goto ve küresel değişken kullanımı gibi) ve **GEREKSİZ YERE UZUN YAPILAR** (peş peşe bir sürü if else ile upuzun bir kod yazmak gibi) kullanmanız durumunda kodunuz tam olarak çalışsa bile **çok düşük/sıfır puan** alabilirsiniz.
- Kodunuzun **hata vermeden** istenen şekilde çalışıyor olması önceliğiniz olsun. Ardından **vakit kalırsa**, detaylandırılmış uyarı ayarları açıkken **uyarı vermiyor olmasına** dikkat ediniz. Öte yandan, eğer kodunuz doğru çalışmıyorsa bunun nedeni uyarı olarak aşağıda belirtili olabilir; bu açıdan **uyarılarla önem veriniz.**
- Sorunun istediklerine **ek özellikler eklemek** (süre, renk, daktilo kodu vs. gibi) size ekstra puanlar getirmez.

SORU 1 – MOTİFLİ ÜÇGEN SORUSU (50 PUAN + 5 EKSTRA PUAN)

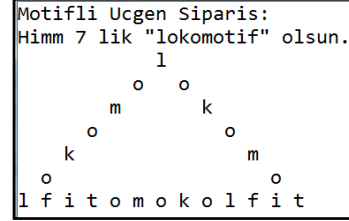
Bu soru parçalardan oluşmaktadır. Bazı parçaları yapamasanız bile diğer parçalardan puan almanız mümkündür. Yaptığınız parçalardan, kodunuzun doğruluk miktarına göre tam ya da kısmi puanlar alırsınız. **KOD GÖNDERİM FORMUNDA BU SORUYA DAİR SADECE 1 KOD GÖNDERECEKSİNİZ.** Bu soruda sizden örnek çıktı 1.1 ve 1.2'de gözüktüğü şekilde,

- kullanıcıdan **satır adedi** ve **motif girişini** alan (kullanıcı girişi biçimi aşağıda anlatılacaktır)
- Dikeyde **satır adedi** adedince yer tutan, içi boş, örneklerdeki gibi ikizkenar üçgen şeklinde ve **verilen motiftteki harflerle** soruda istenen motif işleme yönünde ekrana şekil bastıran bir kod yazmanız istenmektedir.

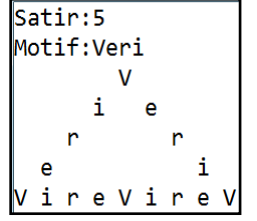
UYARI: Dikey ve yatay karakterler arasındaki uzaklığın kabaca birbirine denk olabilmesi için yatayda bir kareye ait karakterler arasında bir boşluk karakteri genişliğinde aralık olmasına (örn. abc yerine a b c şekliyle) dikkat ediniz.



Örnek çıktı-1.1 Kullanıcı "lütfen üçgen çizdir..." anlamında Almanca bir ifade girmiştir. İfadedeki çift tırnak içindeki kısım("12345") motif olarak algılandıktan sonra geriye kalan kısımdaki sayı (6) satır olarak algılanmıştır. Algılanan motif teki karakterler (rakamlar da bir karakterdir) algılanan satırlık üçgene soruda oklarla gösterili yönde işlenmiştir



Örnek çıktı-1.2 Kullanıcı "Himm 7 lik "lokomotif" olsun." girişi yapıp enter'a basmıştır. Çift tırnaklar içerisi("lokomotif") motif, geriye kalan kısımdaki sayı(7) ise satır olarak algılanmıştır. Soruda istenen şekilde ekrana yazdırılmıştır.



Örnek çıktı-1.3 Kullanıcı sırasıyla 5 ve "Veri" girişleri yapıp her girişten sonra enter'a basmıştır. Bu kod Parça4'ten puan alamaz; Parça 4 haricindeki parçalardan puan alabilir.

Çözümünüz parçalı puanlama ile puanlandırılacaktır. Bu nedenle yapamadığınız bir parçayı atlayıp diğer parçalara odaklanabilirsiniz. Parçalar ve puanları şu şekildedir:

- İSTENEN BİÇİMDE BİR ÜÇGEN (İÇİ DOLU YA DA BOŞ) OLUŞTURABİLMEK (5 PUAN):** (Kullanıcının girişinin alınması ile ilgili aşağıdaki 4 nolu parçaya bakınız.) Kullanıcının girdisinden edinilen satır adedi n ise, **ikizkenar üçgen şekli** n adet satır kaplamalı, birbirine eşit olan kenarlarında n adet karakter bulunmalı. Yataydaki karakterler **arasında boşluk karakteri** olması ile ilgili yukarıdaki uyarıya dikkat ediniz. Bu uyarıyı dikkate almayan kodlar bu parçadan tam puan alamaz. Bu kısımdan puan alabilmeniz için, motif yerleştirmenize gerek yoktur; yıldız (*) karakteriyle yazdırılmış şekiller de bu kısımdan puan alabilir.
- OLUŞTURULAN ÜÇGENİN İÇİNİN BOŞ OLMASI (5 PUAN):** Bu kısmı yapan, Parça1'den de ayrıca puan alır. Sorunun devamı için bu parçanın yapılması zorunlu değildir. İçi dolu üçgenler oluşturursanız bu parça haricinde diğer parçalardan puan alabilirsiniz. İçi boş üçgen oluşturabilmek için satır değerinin en az 3 olması gerektiğine dikkat ediniz. Bu nedenle satır değeri 1 ya da 2 olduğunda içi boş üçgen oluşturulması beklenmemektedir. Yıldız (*) karakteriyle (motifsiz) yazdırılmış şekiller de bu kısımdan puan alabilir.
- MOTİFLİ ŞEKİLDE BASTIRABİLMEK (20 PUAN):** Üçgenin kullanıcının girdiği motif ile çevrelenmiş şekilde ekrana bastırılması. Örnek çıktılarda gözüktüğü üzere, motif en üstten başlayacak, sağ alta doğru ilerleyecek, sağ alt kenara geldikten sonra sola doğru ilerleyecek ve sol alt kenara geldikten sonra en üste doğru ilerleyecektir.
- GİRİŞİ BİR CÜMLE ŞEKLİNDE ALIP İŞLEYEBİLMEK (20 PUAN):** Kullanıcı satır adedini ve motifi örnek çıktılarda gözüktüğü gibi **bir cümle şeklinde** sunabilmeli. Bu cümle içerisinde çift tırnak("") içerisinde geçen kısım motif olacaktır. Motif tek sözcükten oluşacaktır (boşluk içermeyecektir). Geriye kalan kısımda geçen sayı ise **satır adedi** olacaktır. Sayı **birkaç basamaklı** (örn. 14) olabilir. Sayının hemen peşinden başka bir harf gelemaz (Örn. satır adedi cümlede "7lik" olarak geçemez.) Kullanıcının **GEÇERLİ GİRİŞLER yapacağını** (iki sözcüklü motif, iki adet sayı, bitişik ekli sayı, iki adet cümle gibi geçersiz girişler yapmayacağını) varsayınız. Bu kısım yerine **Örnek çıktı-1.3'deki** gibi sade şekilde verileri alırsanız, bu parçadan puan alamazsınız; ama diğer parçalardan puan alabilirsiniz.
- EKSTRA:FONKSİYON KULLANIMI (5 PUAN):** Yukarıdaki parçalardan herhangi biri hariç hepsinden puan alabilmiş öğrenciler için geçerlidir. **Fonksiyonları** genel ve etkin şekilde kullanmak ve kodu rahat geliştirilebilir hale getirmek (örn. void alıp void dönen vb. şekilde çok sade bir adet fonksiyon kullanımı fonksiyonların genel ve etkin kullanımı sayılmaz.)

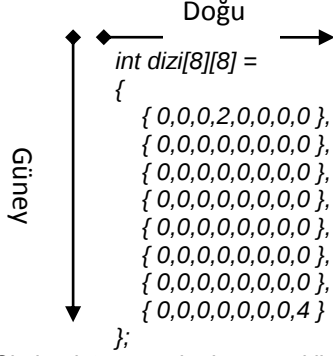
ÖNEMLİ UYARI:

- Sınav sonunda süreniz kalırsa, 1) detaylandırılmış uyarı açık şekilde kodunuzu inceleyiniz ve kodunuzun uyarı üretmiyor olmasına dikkat ediniz 2) kodunuzu değişik değerlerle iyice test ederek doğru çalıştığına emin olunuz.
- Sadece, doğru çıktı veren bir kod yazarak tam puan alamayabilirsiniz. Aynı zamanda sizden istenen, **OLABİLDİĞİNCE SADE** bir kod yazabilmenizdir. (bkz. 1 nolu sayfanın sonundaki **KODLAMAYA DAİR UYARILAR** kısmı)

SORU 2 – YAPAY ZEKA YUMİ SORUSU (50 PUAN + 5 EKSTRA PUAN)

Bu soru parçalardan oluşmaktadır. Bazı parçaları yapamasanız bile diğer parçalardan puan almanız mümkündür. Yaptığınız parçalardan, kodunuzun doğruluk miktarına göre tam ya da kısmi puanlar alırsınız. **KOD GÖNDERİM FORMUNDA BU SORUYA DAİR SADECE 1 KOD GÖNDERECEKSİNİZ.**

Yapay zeka insan zekası süreçlerinin makinelerce benzetiminin yapılmasıdır. Bu süreçler öğrenmeyi, mantık yürütmeyi, problem çözmeyi içerir. Yapay zekanın bir uygulama alanı olan *"Doğal Dil İşleme"* nin basit bir şeklini 1.soruda gördünüz. Bu soruda ise, hayal gücümüzle 2041 yılına gidiyoruz. İnsanoğlundaki sağlıklı yaşam tutkusu yeni bir boyut almış ve robotlarla eve organik yumurta siparişi dönemi başlamıştır. Bu sorudaki robot Yumi'nin görevi, içindeki mikroipte yüklü olan "müşteri uzaklık" ve "müşteri yumurta kolisi talep" verilerini göz önünde bulundurarak mantık yürütmesi ve hangi müşteriye yönelmesi gerektiğine karar vermesidir.



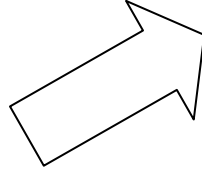
Bu soruda, kodunuzu rahatça test edebilmemiz için kullanıcıdan değer **alınMA**maktadır. Kodunuzun başında **main** içerisinde solda gözüktüğü biçimde 8*8'lik bir *int* dizisi tanımlamanız istenmektedir. Bu dizi, Yumi'nin hizmet sahasının (8km*8km) kuş bakışı görüntüsünü temsil etmektedir. Müşteriler yumurta kolisi talepleriyle ilgili adreslere not düşülmüştür. Yumi, (0,0) koordinatında ([0][0] indislerinde) bulunmaktadır. Soldaki dizi içeriğine göre, 3 km doğu yönünde (dizi[0][3]) 2 koli yumurta isteyen bir müşteri bulunmaktadır; 7 km doğu 7 km güney yönünde ise 4 koli yumurta isteyen başka bir müşteri bulunmaktadır. **Yumurta kolisi(KDV dahil) adet fiyatı 50 TL olmaktadır.** Yumi'nin **elektrik masrafı ise km başına 10 TL'dir.** O gün sadece bir müşteriye hizmet verebilecek olan Yumi **hangi müşterisini seçmelidir?**

Sizden istenen adımlar şu şekildedir:

1. Kodun içindeki dizideki verilerden yola çıkarak, en çok kâr getirecek müşteriye belirleyiniz.
2. Bu müşterinin adresini ekrana yazdırınız.
3. Bu müşteriye teslim edilecek yumurta kolisi adedini ekrana yazdırınız.
4. Müşterinin yapacağı ödemeyi (yumurtanın kolisi 50 TL olacak şekilde hesaplanır) ekrana yazdırınız.
5. Yumi'nin elektrik masrafını (gidis-gelis olmak üzere - km başına 10 TL olacak şekilde hesaplanır) ekrana yazdırınız.
6. Kâr değerini ekrana yazdırınız.

Hizmet sahasının her zaman 8*8'lik olacağını varsayınız. Yumi, doğadaki kuşları gözlemleyerek zaman içerisinde uçmayı öğrenmiştir. Bu nedenle müşteriye **doğrudan gidebilmektedir.** (Örn. Güney:3 - Doğu:4 konumlu müşterisine 5 km'lik bir uçuşla erişir.)

```
int dizi[8][8] =
{
  { 0,0,0,2,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,4 }
};
```

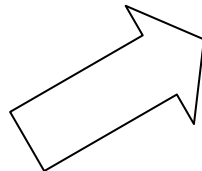


Güney yönünde:	0 km
Doğu yönünde:	3 km
Organik yumurta koli teslim:	2 adet
Musteri odemesi (KDV dahil):	100.00 Turk Lirası
Elektrik masrafı (gidis-gelis):	60.00 Turk Lirası
Elde edilen kar:	40.00 Turk Lirası

Örnek çıktı-2.1 main içerisindeki dizi tanımı solda gözüktüğü şekildedir. Bu durumda kod çalıştırıldığında Yumi, *dizi[0][3]*'te konumlu müşteriye seçmiştir. Müşteri 3 km uzaklıkta olduğu için gidis-gelis 6 km elektrik harcaması olacaktır. Hesap neticesinde bu müşteri 40.00 TL kâr getirmektedir.

Haritaya bakıldığında *dizi[7][7]*'de konumlu müşteri daha çok yumurta alacaktır ama kâr hesabı yapıldığında **7√2 km uzaklıkta yer alan** bu müşterinin 2.01 TL kâr getireceği görülmüştür ve bu nedenle kendisi tercih edilmemiştir. Neticesinde, verilen karar detaylarıyla ekrana yazdırılmıştır.

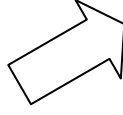
```
int dizi[8][8] =
{
  { 0,0,0,2,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 6,0,0,0,0,7,0,0 },
  { 0,0,8,7,0,0,0,0 },
  { 0,5,0,0,0,0,0,0 },
  { 0,0,0,0,0,0,0,0 },
  { 4,0,0,0,0,0,0,0 },
  { 9,0,0,0,0,0,0,4 }
};
```



Güney yönünde:	3 km
Doğu yönünde:	2 km
Organik yumurta koli teslim:	8 adet
Musteri odemesi (KDV dahil):	400.00 Turk Lirası
Elektrik masrafı (gidis-gelis):	72.11 Turk Lirası
Elde edilen kar:	327.89 Turk Lirası

Örnek çıktı-2.2 main içerisindeki dizi tanımının solda gözüktüğü şekildedir. Bu durumda kod çalıştırıldığında Yumi, 9 müşteri arasından *dizi[3][2]*'de konumlu müşteriye seçmiştir. 8 adet koli, her biri 50 TL'den 400 TL getirmektedir. Elektrik masrafı düşüldüğünde bu müşterinin 327.89 TL kâr getireceği anlaşılmaktadır.

```
int dizi[8][8] =
{
  { 0, 0, 0, 0, 0, 0, 0, 0 },
  { 23, 0, 0, 0, 0, 0, 0, 0 },
  { 0, 0, 0, 0, 0, 0, 0, 0 },
  { 0, 0, 0, 0, 25, 0, 0, 0 },
  { 0, 0, 0, 24, 0, 0, 0, 0 },
  { 0, 0, 0, 0, 0, 0, 0, 0 },
  { 0, 0, 0, 0, 0, 0, 0, 0 },
  { 10, 0, 0, 0, 0, 0, 0, 26 }
};
```



Güney yönünde:	3 km
Doğu yönünde:	4 km
Organik yumurta koli teslim:	25 adet
Müşteri ödemesi (KDV dahil):	1250.00 Türk Lirası
Elektrik masrafı (gidis-gelis):	100.00 Türk Lirası
Elde edilen kar:	1150.00 Türk Lirası

Örnek çıktı-2.3 main içerisindeki dizi tanımının solda gözüktüğü şekildedir. Bu durumda kod çalıştırıldığında Yumi, 5 müşteri arasında *dizi[3][4]*'te konumlu müşteriyi seçmiştir. 25 adet koli, her biri 50 TL'den 1250 TL getirmektedir. Müşterinin uzaklığı, Yumi doğrudan uçarak gittiği için 5 km'dir. Gidiş-geliş 10 km uçuş için Yumi'nin elektrik masrafı 10 km x 10 TL/km = 100 TL'dir.

ÖNEMLİ UYARI:

- Sınav sonunda süreniz kalırsa, 1) detaylandırılmış uyarı açık şekilde kodunuzu inceleyiniz ve kodunuzun uyarı üretmiyor olmasına dikkat ediniz 2) kodunuzu değişik değerlerle iyice test ederek doğru çalıştığına emin olunuz.
- Sadece, doğru çıktı veren bir kod yazarak tam puan alamayabilirsiniz. Aynı zamanda sizden istenen, **OLABİLDİĞİNCE SADE** bir kod yazabilmenizdir. (bkz. 1 nolu sayfanın sonundaki **KODLAMAYA DAİR UYARILAR** kısmı)

Çözümünüz parçalı puanlama ile puanlandırılacaktır. Bu nedenle yapamadığınız bir parçayı atlayıp diğer parçalara odaklanabilirsiniz. Parçalar ve puanları şu şekildedir:

- EN UYGUN MÜŞTERİYİ SEÇİP ADRESİNİ DOĞRU ŞEKİLDE BELİRTEBİLMEK (25 PUAN):** Kod dizideki verileri esas alarak müşteriler arasından en çok kâr getireni seçebilmeli. Eşit miktarda kâr getiren müşteriler olması durumunda herhangi birini seçmesi yeterlidir. Seçtiği müşteriye ait koordinatları istenilen şekilde yazdırabilmeli.
- SEÇİLEN MÜŞTERİYE AİT VERİLERİ EKRANA YAZDIRABİLMEK (10 PUAN):** Seçilen müşterinin verileri (çıkartılardaki 4., 5. ve 6. satırlar) doğru şekilde hesaplanmalı ve örneklerde gözüktüğü şekilde (örn. noktadan sonra 2 basamaklı şekilde) ekrana yazdırılmalı.
- HİZALAMALARI UYGUN ŞEKİLDE YAPABİLMEK (5 PUAN):** Örnek çıkartılarda gözükten tüm satırlardaki sayısal değerlerden önce uygun şekilde boşluk koyan hizalama kodunu yazabilmek (yol haritasını yazdırmanız istenmemektedir). Bu hizalama peş peşe boşluk karakterleriyle (örn. printf(" "); gibi) yapıldığında buradan puan alınamaz.



musteri_yapi: struct

dogu: int

guney: int

kar: double

yol_masrafı: double

yumurta_ucret: double

4. **YAPI KULLANABİLMEK (5 PUAN):** DevCpp->View->Project/Class Browser sol panelinde Classes sekmesi altındaki görünümü soldaki gibi olan bir yapı tanımını ETKİN şekilde kullanarak soruyu çözmek.

5. **FONKSİYON KULLANABİLMEK (5 PUAN):** Kodunda *musteri_tespit* isimli *void musterit_tespit(int j[8], musteriyapi*)*; prototipine sahip bir fonksiyonu ETKİN şekilde kullanarak soruyu çözmek.

- EKSTRA: GELİŞTİRİLEBİLİR BİR KOD YAZMIŞ OLMAK (5 PUAN):** Yukarıdaki parçalardan herhangi biri hariç hepsinden puan alabilmiş öğrenciler için geçerlidir. Kod *geliştirilebilirlik* özelliğine sahip olmalı. Parametrik şekilde kodlanmış bir kod olmalı. Örneğin, yumurta bedeli 60 TL yapılacak olduğunda koddaki *sadece bir yerdeki* 50 değerinin 60 ile değiştirilmesi yeterli olmalı. Benzer şekilde hizmet sahasının EN-BOY değerleri 8-8 yerine 12-15 yapılacak olsa, koddaki *sadece iki yerde* değişiklik yapmak ve dizinin içindeki değerleri güncellemek yeterli olmalıdır. (Bunu sağlamak için soruda kullanmanız istenen dizi tanımında geçen 8 değerlerini değiştirmeniz normaldir.)

LÜTFEN KOD GÖNDERİM AŞAMASINA **SINAVIN SON 5 DAKİKASINDAN ÖNCE** GEÇİNİZ. BU AŞAMAYA GEÇTİĞİNİZDE EN YAKININIZDAKİ GÖZETMENE HABER VERİNİZ (Böylece DevCpp dışında bir program açtığınız için sizi uyarmaya gelmezler.)



Kodunuzu göndereceğiniz form:

BU KÂĞIDI ÖN KISMINDAKİ BİLGİLERİ DOLDURDUKTAN VE İMZALADIKTAN SONRA **GÖZETMENİNİZE TESLİM ETMEYİ** UNUTMAYINIZ.

OLASI ÇÖZÜMLER

SORU1 ÇÖZÜM ALGORİTMALARI:

1. Verinin alınışı:
 - Tampondaki satırın tamamı bir dizgiye çekilip strtok'la vb. parçalara ayrılabilir, veri işlenebilir. Ancak bu işleri uzatabilir.
 - scanf'li çözüm: Tampondan scanf %d/%s ile tek tek çekilebilir. *scanf başarısız olduğunda 1 değeri dönmeyi ve alamadığı ifadeyi tamponda bırakır. scanf %d 1 dönmüyorsa tampondaki sayı değildir.*
2. Üçgenin motifle yazdırılışı:
 - Klasik çözüm: Kenarlar için ayrı indis sayaçları tutarak motif işlenebilir.
 - Teknik çözüm:İmleci ekranda oynatmaktansa iki boyutlu bir diziye istediğimiz şekli yerleştirip sonra ekrana bastırabiliriz.
 - İmleçli çözüm: imlec_xy ile imleci ekranda dolaştırarak motifi ekrana bastırabiliriz.

SORU 1 - SCANF %d/%s İLE VERİ ALIMI VE (TEKNİK ÇÖZÜM)2 BOYUTLU DİZİYE İŞLENEREK ÇÖZÜM

```
#include <stdio.h>
#include <string.h>
#define PARCA4 0 //PARCA4lu cozum icin 1 degilse 0 yapiniz.

int motifMi(char sozcuk[]);
void kirp (char sozcuk[], char motif[]);
void tabloyu_isle(int satir, int sutun, char motif[], char tablo[][sutun] );
void yazdir(int satir, int sutun, char tablo[][sutun]);

int main()
{
    // File-> Properties -> Actual code Lines: 78 (+ 11 diğ er PARCA4'süz halini de kapsadığı için)
    char sozcuk[100], motif[100];
    int satir;
    // motif icerisinde bosluk olamaz - kullanicı uygun sekilde girer.
    if(PARCA4)
    {
        /*
        Keşke Uygulamalı Dersteki codingame sorularını çözseydim.
        */
        printf("Motifli Ucgen Siparis:\n");
        int alinan_veri = 0;
        while( alinan_veri!=2 )
        {
            int sayiMi=0;
            /* scanf veri alabilirse, aldigi veri adedini(1) doner.
            Alamazsa başka şey döner ve alamdığını da tamponda bırakır.
            bkz. scanf slaytları - keşke ek slaytlara baksaydım*/
            sayiMi = scanf("%d", &satir);

            if( sayiMi == 1 )
            {
                // SAYI ALINDI
                alinan_veri++;
            }
            else
            {
                // SAYI OLMAYAN İFADE TAMPONDA KALMISTI
                scanf("%s", sozcuk); //SOZCUK ALINDI
                if( motifMi(sozcuk) ) // ACABA MOTIF MI?
                {
                    kirp(sozcuk, motif); //bastaki ve sondaki " ları atar. motife koyar.
                    alinan_veri++;
                }
            }
        }
    }
    else
    {
        printf("Satir:");
    }
}
```

```

        scanf("%d", &satir);
        printf("Motif:");
        scanf("%s", motif);
    }
    // printf("<%s><%s>\n", veri, sozcuk ); //DEBUG
    int sutun = 2*satir-1; //alt kenardaki karakter sayisi
    char tablo[satir][sutun];
    // ekrana bastirsan, imleci geri getirmek bir dert
    // once tabloya kaydedelim, sonra bastiririz
    tabloyu_isle(satir, sutun, motif, tablo);
    yazdir(satir, sutun, tablo);
    return 0;
}

int motifMi(char sozcuk[])
{
    // basinda ve sonunda " var mi
    if( sozcuk[0]=='\'' && sozcuk[ strlen(sozcuk)-1 ]== '\'' )
        return 1;
    else
        return 0;
}

void kirp (char sozcuk[], char motif[])
{
    int i, j=0;
    int sol = 1, sag = 1; //sol ve sagdan 1er karakter kirpacak.
    // yanindaki '\0' olana son karakter derler, onu istemiyoruz.
    for(i=sol; sozcuk[i+sag]!='\0'; i++)
    {
        /*
        motif[j]'e kaydet ve peşinden hemen motif'in içindeki
        sembolik imlecim olan j'i sağa kaydır.
        */
        motif[j++] = sozcuk[i];
    }
    motif[j] = '\0';
    // j diye ayri indis olusturup, i üzerinden cambazlık yapmıyorum.
    // motif[i-1] yazsam da olurdu belki ama 4 bayt harcarim, beynimi yormam.
}

void tabloyu_isle(int satir, int sutun, char motif[], char tablo[][sutun] )
{
    int i,j;
    for(i=0; i<satir; i++)
        for(j=0; j<sutun; j++)
            tablo[i][j]=' ';
    int v=0;
    int veri_uzunluk = strlen(motif);
    //sağ kenarı işle
    for(i=0, j=sutun/2; i<satir-1; i++, j++)
        tablo[i][j] = motif[(v++)%veri_uzunluk];
    //alt kenarı işle
    for(j=sutun-1; 0<j; j--)
        tablo[satir-1][j] = motif[(v++)%veri_uzunluk];
    //sol kenarı işle
    for(i=satir-1, j=0; 1<=i; i--, j++)
        tablo[i][j] = motif[(v++)%veri_uzunluk];
}

void yazdir(int satir, int sutun, char tablo[][sutun])
{
    int i,j;
    for(i=0; i<satir; i++)
    {
        for(j=0; j<sutun; j++)
            printf("%c ", tablo[i][j]);
        printf("\n");
    }
}

```

SORU 1 - (TEKNİK ÇÖZÜM) 2 BOYUTLU DİZİYE İŞLENEREK ÇÖZÜM - KODUN SADECE MOTİF İŞLEME KISMI

```
#include <stdio.h>
#include <string.h>
int main()
{
    char motif[10]="abcdef"; //kullanıcıdan alınacak
    int l = strlen(motif);
    int n=6; //kullanıcıdan alınacak
    int i,j;
    int r = 0;
    int b = 3*n-4;
    int sol = 4*n-5;
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=2*n-1; j++)
        {
            if(j-i ==n-1) //sag kenar
            {
                printf("%c ", motif[r%l] );
                r++;
            }
            else if(i+j==n+1) //sol kenar
            {
                printf("%c ", motif[sol%l] );
                sol--;
            }
            else if( i == n) // alt kenar
            {
                printf("%c ", motif[b%l]);
                b--;
            }
            else
                printf(" ");
            // getch();
        }
        printf("\n");
    }
    return 0;
}
```

SORU 1 - (İMİLEÇLİ ÇÖZÜM) İMLECİ EKRANDA DOLAŞTIRAN ÇÖZÜM - KODUN SADECE MOTİF İŞLEME KISMI

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <windows.h>
void imlec_xy(int x, int y);
int main()
{
    char dizgi[] = "abcde";
    int i, n=8, l = strlen(dizgi);
    int yatay=1;
    int dikey=1;
    int x=2*(n-1);
    int y=0; //tüm çözümde ilk iki satir dolu oldugu icin burasi 2 olacak
    for(i=0; i<4*n-4; i++)
    {
        imlec_xy(x,y);
        printf("%c", dizgi[i%l]);
        if(i==n-1 || i==3*n-3)
        {
            yatay*=-1;
            dikey-=1;
        }
        x+=2*yatay;
    }
```

```

        y+=dikey;
        getch();
    }
    imlec_xy(0,n+1);
    return 0;
}
void imlec_xy(int x, int y)
{
    COORD coord;
    coord.X = x;
    coord.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE), coord);
}

```

SORU2 ÇÖZÜM ALGORİTMALARI:

Sorunun çözümü için alternatif algoritmalar üretmeye gerek yoktur. Maksimum bulma algoritması ile dizinin tüm elemanları dolaşılıp maksimum kâr getireninin indislerini tespit etmek yeterlidir. *Şu ana kadar maksimum bildiğimden, daha büyük bir elemana rastlarsam maksimum bildiğimi o eleman olarak güncellerim.* Kabarcık sıralama gibi sıralama algoritmaları bu soru için çözümü gereksiz yere uzatmak anlamına gelir. Eğer 2.maksimuma vs. ihtiyaç olsaydı, sıralama daha cazip olabilirdi.

SORU2 - STANDART ÇÖZÜM

```

#include <stdio.h>
#include <math.h>
#define YUMURTA_BEDEL 50 //TL
#define ELEKTRIK_KM 10
#define EN 8
#define BOY 8

typedef struct
{
    int dogu, guney;
    double yumurta_ucret, yol_masrafi, kar;
} musteri_yapi;

double mesafe (int x, int y);
void musteri_tespit(int [][]EN, musteri_yapi* );

int main()
{
    // File-> Properties -> Actual code lines: 66
    int dizi[BOY][EN] =
    {
        {0, ,0, ,0, ,0, ,0, ,0, },
        {23, ,0, ,0, ,0, ,0, ,0, ,0, },
        {0, ,0, ,0, ,0, ,0, ,0, ,0, },
        {0, ,0, ,0, ,0, ,25, ,0, ,0, ,0, },
        {0, ,0, ,0, ,25, ,0, ,0, ,0, ,0, },
        {0, ,0, ,0, ,0, ,0, ,0, ,0, ,0, },
        {0, ,0, ,0, ,0, ,0, ,0, ,0, ,0, },
        {10, ,0, ,0, ,0, ,0, ,0, ,0, ,26, }
    };
    musteri_yapi musteri;
    musteri_tespit(dizi, &musteri);
    printf("Guney yonunde:\t\t\t\t\t km\nDogu yonunde:\t\t\t\t\t km\n", musteri.guney, musteri.dogu);
    printf("Organik yumurta koli teslim:\t\t\t\t\t adet\n", dizi[musteri.guney][musteri.dogu]);
    printf("Musteri odemesi (KDV dahil):\t\t\t\t\t %21f Turk Lirasi\n", musteri.yumurta_ucret);
    printf("Elektrik masrafi (gidis-gelis):\t\t\t\t\t %21f Turk Lirasi\n", musteri.yol_masrafi);
    printf("Elde edilen kar:\t\t\t\t\t %21f Turk Lirasi", musteri.kar);
    return 0;
}

```

```

double mesafe (int x, int y)
{
    return sqrt( pow(x,2)+pow(y,2) );
}

void muster_i_tespit(int dizi[BOY][EN], muster_i_yapi* maks_ptr)
{
    double yumurta_ucret, yol_masrafi, kar;
    int i,j;
    maks_ptr->kar = 0;
    for(i=0; i<BOY; i++)
        for(j=0; j<EN; j++)
        {
            if(dizi[i][j])
            {
                yumurta_ucret = dizi[i][j]*YUMURTA_BEDEL;
                yol_masrafi = 2*mesafe(i,j)*ELEKTRIK_KM;
                kar = yumurta_ucret - yol_masrafi;
                if( maks_ptr->kar < kar )
                {
                    *maks_ptr = (muster_i_yapi)
                    {
                        //yapi degiskenine deger atama
                        .dogu=j,
                        .guney=i,
                        .kar=kar,
                        .yumurta_ucret=yumurta_ucret,
                        .yol_masrafi=yol_masrafi
                    };
                }
            }
        }
}

```