

KISA SINAVLAR HAKKINDA BİLGİLENDİRME

BİL141

BİLGİSAYAR PROGRAMLAMA DERSİ

Kısa sınavlara dair özet bilgiler:

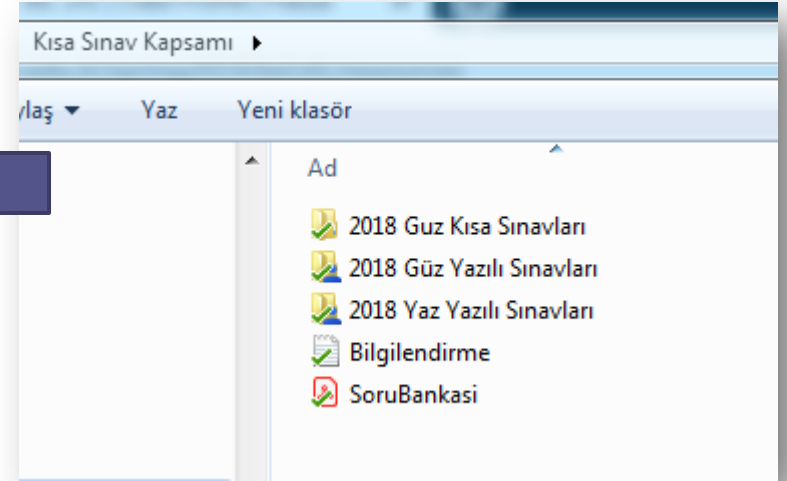
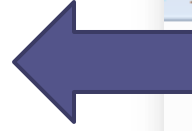
- 3 soru türü vardır:
 - Çoktan seçmeli soru türü
 - Kodda boşluk doldurma soru türü (bkz. Soru Bankası)
 - Ekran çıktısı soru türü (bkz. Soru Bankası)
- Genelde toplam 4 ila 8 sorudan oluşur.
- Genelde süresi **10-15 dakikadır**. Uygulama ders esnasında, uygun görülen bir zaman aralığında yapılır.

Kısa sınavlara dair özet bilgiler:

Kısa Sınavlar Ders Drive Klasörü-> Kısa Sınav Kapsamı'ndan erişebileceğiniz soru havuzundan seçilmiş soruların ufak değişiklikler içeren sorularından oluşur.

Bunun 2 istisnası olabilir:

- İzlemenizi istediğimiz videolardan gelebilecek soru (Kısa Sınav I)
- DevCpp hata paneli soruları (bir örneği için bkz. [2018 Guz Kısa Sınavları->KısaSınavIII_SoruCevaplar](#))
- Ayrıca kısa sınavlardan önce Piazza'dan gelebilen o kısa sınava özel bilgilendirme mesajlarını takip ediniz. Toplam soru adedi, hangi konudan kaç soru çıkacağı gibi bilgiler bu mesajlarda sizlere duyurulabilir.
- **En yüksek puanlı 5 kısa sınav** değerlendirmeye alınır.



Kendi şubesi dışında kısa sınava girmek mümkündür ancak ilgili kısa sınav notunun %90'ı değerlendirme alınır. (2018-2019 Yaz döneminde Şube2 çok yoğun olduğu için, Şube2 öğrencileri Şube1'de derse ve kısa sınava girerlerse onlara puan kaybı uygulanmaz.)

GÖZETMENLER GÖZETMEK İÇİNDİR.

- Gözetmenlerimiz sınavdan habersiz kişilerdir.
 - Onlara soru sormayın.
 - Onların verdikleri cevaplar tahminden ibaret olup **yanıltıcı** olabilir.
- Soru sormanızı gerektirecek durum zaten soru metninde açıklanıyor olabilir.
 - Sorunun yanlış/eksik olduğunu düşünüyorsanız bir varsayımda bulunup, varsayımınızı ve ona göre olan cevabınızı yazınız.



Sınav Güvenliği

- Sınav güvenliği açısından, kısa sınavlarımız her şubeye farklı sorular ve her şubede de en az 2 grup olacak şekilde tasarlıdır.
- Bu önlemlerin yeterliliği konusunda herhangi bir endişeniz olduğunda durumu en kısa sürede bize iletiniz.

Çalışma yöntemi tavsiyesi

- Bugüne kadarki çalışmalarınızın üzerinden geçin. Slaytlara çok iyi hâkim olun.
- Kısa sınav soru havuzundaki soruları birkaç defa çözün.



Tam Cevap Sistemi

- Sınavda tam cevap sistemi uygulanmaktadır çünkü eşik değerler (az yanlış, orta yanlış, çok yanlış gibi) belirlemek ve parçalı puan atamak pratik olmamaktadır.
- Bu nedenle cevap 25 ise 24 yazan öğrenci de 20 yazan öğrenci de o sorudan puan alamaz.

Kısa sınavlarda “En İyisi” Sistemi

- Dönem içinde yapılan kısa sınavların en yüksek puanlı 5 tanesi değerlendirmeye alınır.

Çoktan seçmeli soru türü

- Bir soru ve 4-5 şıktan oluşur.

- Örnek:

Piazza’da paylaşılan “algoritma” belgeselinde hangi sıralama algoritmasına değinilmemiştir?

a) Kabarcık(baloncuk) sıralama

b) ...

Kodda boşluk doldurma soru türü

- Aşağıda bazı kodlar ve ekran çıktıları verilmiştir. Boşlukları doldurunuz.

BU BOŞLUKLARIN HER BİRİNDEN SADECE, DOĞRU CEVABI TAM OLARAK VERMENİZ DURUMUNDA PUAN ALABİLİRSİNİZ. BİR BOŞLUK YANLIŞSA SADECE O BOŞLUKTAN PUAN KAYBEDERSİNİZ.

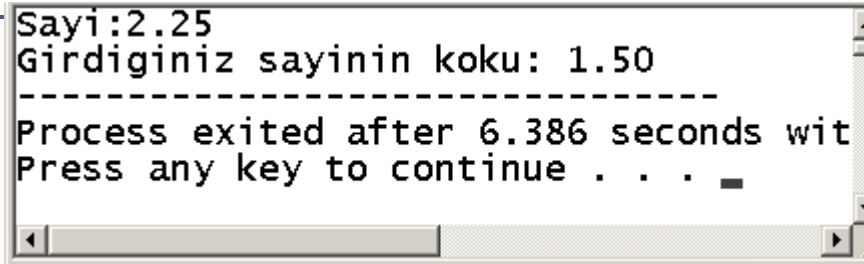
- / ve \ farkı, ' ve " farkı, $\leq$ yerine $<=$ gibi detaylar da dâhil olmak üzere cevabınızın tam olarak doğruluğuna emin olunuz. r yerine R yazmak gibi durumlar, soruda özellikle ölçülenler arasında değilse, puan kaybına yol açmaz.
- Mümkün olduğunca sade komut/ifadeler kullanmanız beklenmektedir.
($x = y$ varken $x = y+1-1$; yazılması puan kaybettirebilir.) Gereksiz komut/ifadeler eklemeyiniz.
- Sadece boşluklara yazdıklarınız değerlendirmeye alınır. Boşluklar içine cevap dışında hiçbir şey yazmayınız.
- Kod çalıştırıldığında derleyici herhangi bir uyarı ya da hata vermeyecek şekilde boşlukları doldurunuz.

Kodda boşluk doldurma soru türü

- Aşağıdaki kod, kullanıcıdan bir sayı alıp kökünü hesaplamakta ve örnek çıktıdaki gibi ekrana yazdırmaktadır. Koddaki boşlukları doldurunuz.

```
#include <stdio.h>
#include <_____>
int main()
{
    double sayi;
    printf("Sayi:");

    _____ ("Girdiginiz sayinin koku: _____", sqrt(sayi));
    _____ 0;
}
```



```
Sayi:2.25
Girdiginiz sayinin koku: 1.50
-----
Process exited after 6.386 seconds with return code 0
Press any key to continue . . .
```

Ekran çıktısı soru türü// Soru metni -1/2

- **BU SORULARDAN SADECE, DOĞRU CEVABI TAM OLARAK VERMENİZ DURUMUNDA PUAN ALABİLİRSİNİZ.**

Örneğin, cevap 14.3 ise 14,3 ya da 14.30 yazan öğrenci o sorudan puan alamaz!

- Kutular içine ekran çıktısını içeren cevabınız dışında hiçbir şey **yazmayınız.**
- Kutulardaki **her hücre ekrandaki bir karakteri** temsil eder. Bir hücreye birden fazla karakter **yazmayınız.**
- Cevabınızın kutuya sığmadığı durumlar olursa, ilgili kısma ek hücreler çizerek cevabınızı **belirtebilirsiniz.**
- Bazı sorularda çıktının bir kısmı cevap kutusunun içine yazdırılmış olabilir, bu tip sorularda çıktının **geriye kalan kısmını** siz tamamlayınız. Bu sorularda kullanıcının klavyeden yaptığı giriş değerleri çıktı kutuları içinde ***koyu italik*** olarak gösterilir.

Ekran çıktısı soru türü// Soru metni -2/2

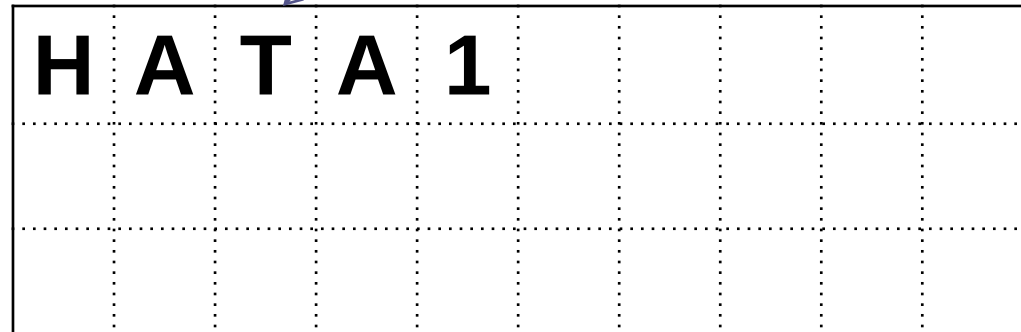
- (Mantık hatası haricinde) Hata içerdiğini düşündüğünüz sorularda ne tür bir hata olduğunu aşağıdaki seçeneklerden seçerek cevap kutusunun içine (örn. “HATA1” şeklinde) **yazınız VE açıklamasını** cevap kutusunun altına ya da soruya yakın başka boş bir alana (örn. “Çünkü koddaki ... eksik.” şeklinde) **belirtiniz**.
 - **HATA1:**Kod hiç derlenemeyecektir.
 - **HATA2:**Kod derlenecek ama çalışması esnasında hata verecektir.
 - **HATA3:**Kod derlenecek ama çalışması esnasında sonsuz döngüye girecektir.
- Yukarıdaki ifadelere uymayan hatalarda (örneğin mantık hatası içerdiğini düşündüğünüz sorularda) hatasız sorularda olduğu gibi sadece ekran çıktısını yazınız, HATA# vs. yazmayınız.
- YUKARIDA BELİRTİLDİĞİ ÜZERE, HATA TESPİTİ DURUMUNDA **HEM HATA NUMARASINI HEM DE NEDENİNİ** YAZMANIZ İSTENMEKTEDİR. PUAN ALABİLMENİZ İÇİN İKİSİNDEN BİRİ EKSİK OLMAMALIDIR.

Sınav kağıdınızda bunun benzeri metin olacaktır. Hata numaralarına ve anlamlarına sınav esnasında bakabileceksiniz.

Dikkat: Hata tespiti durumunda hem numarasını hem de açıklamasını yazmanız istenmektedir.

```
#include <stdio.h>
int main()
{
    Printf("AAA");
    return 0;
}
```

Kutunun içine sadece
hata türü yazılır.



H	A	T	A	1							

Çünkü Printf yerine printf olmalıydı.

Kutunun yakınına
açıklaması yazılır.

```
#include <stdio.h>
int main()
{
    int a;
    printf("AAA:");
    scanf("%d", &a);
    printf("bbb:%d-
%d",a,a);
    return 0;
}
```

Cevap kutusu sınav kağıdına bu şekilde gözükmektedir.

Kalın ve italik kısım kullanıcı girişini temsil eder. Kod çalıştıktan sonra kullanıcı 25 değerini girmiş ve enter'a basmıştır.

A	A	A	:	2	5			
b	b	b	:					

Bu soruda olduğu gibi bazı sorularda çıktının bir kısmı halihazırda cevap kutusunda mevcut olabilir. Bu durumda sizden istenen cevabın geriye kalan kısmını doldurmanızdır.

DOĞRU CEVAP:

A	A	A	:	2	5			
b	b	b	:	2	5	-	2	5

Soruları titizlikle
çözmeli. Cevabımı
dikkatlice kontrol
etmeliyim.



Hatırlatma

- `printf("%lf", 3.5*2);`
 - Ekran çıktısı 7.000000 olur 7 değil!
- Cevaplarda alt satıra inmek ya da inmemek bütün mesele olabilir.
- ***Derslerdeki alıştırmalarda yaptığınız hataları gözden geçirin, bu hatalardan ders alın ve sınavda bu hataları tekrarlamayın!***

Ekran çıktısı soru türü

```
#include <stdio.h>
int main()
{
    int x = 20;
    printf("C:%d", ++x);
    return 0;
}
```

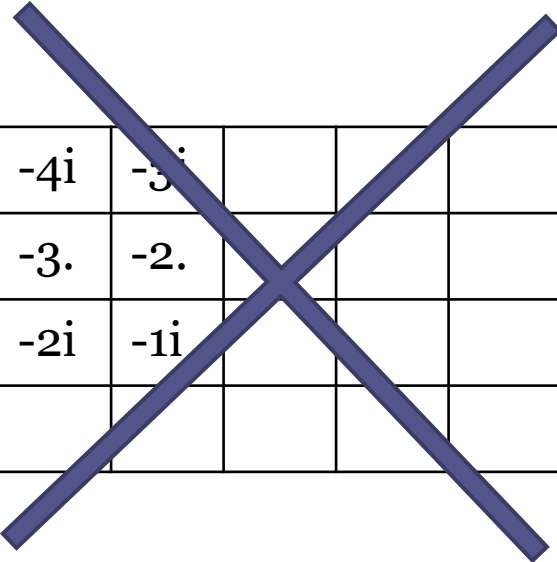
2	1						

C	:	2	1				



Doğru cevap?

Ekran Çıktısı:



-4i	-3i				
-3.	-2.				
-2i	-1i				

Ekran Çıktısı:

-	4	i	-	3	i
-	3	.	-	2	.
-	2	i	-	1	i

Ekran çıktısı soru türü

```
#include <stdio.h>
int main()
{
    int caliskan = 10;
    caliskan/=4;
    caliskan*=caliskan*caliskan;
    printf("%d saat calisti", ++caliskan);
    return 0;
}
```

9		s	a	a	t		c	a	l	i	s	t	i	.

Kutulara
sığmama olursa
ek kutu çizip
cevabınızı
yazınız.