

Ders 34

```
1 //define _CRT_SECURE_NO_WARNINGS
2
3 // odev
4 #include <stdio.h>
5 #include <string.h>
6 #include "nutility.h"
7
8 int main(void)
9 {
10     const char* const pmons[] = {
11         "Ocak",
12         "Subat",
13         "Mart",
14         "Nisan",
15         "Mayis",
16         "Haziran",
17         "Temmuz",
18         "Agustos",
19         "Eylul",
20         "Ekim",
21         "Kasim",
22         "Aralik"
23     };
24 }
```

Eleman adresi ve kendisi değişmez

Örnek zor girilen ayı kaçınıcı olduğunu ekrana yazsın

```

int main(void)
{
    const char* const pmons[] = {
        "ocak",
        "subat",
        "mart",
        "nisan",
        "mayis",
        "haziran",
        "temmuz",
        "agustos",
        "eylul",
        "ekim",
        "kasim",
        "aralik"
    };

    char entry[SIZE];

    printf("bir ay ismi girin: ");
    scanf("%s", entry); //nisan

    size_t i;

    for (i = 0; i < asize(pmons); ++i) {
        if (!strcmp(entry, pmons[i]))
            break;
    }

    if (i == asize(pmons)) {
        printf("%s gecerli bir ay ismi degil\n", entry);
    }
}

```

Döngü gövdesinden çıkmak için if break yerine if içindekini && ile for içinde yaz

Çok önemli bunu bugün ödevde değiştirmem lazım.

```

for (i = 0; i < asize(pmons) && strcmp(entry, pmons[i]); ++i) {
}

if (i == asize(pmons)) {
    printf("%s gecerli bir ay ismi degil\n", entry);
}
else {
}

```

1 Kaç elemanı vardır.

```
"soner", "su", "suheyla", "suleyman", "sumeyye", "suphi", "taci",  
"tansu", "tarcan", "tarik", "tarkan", "tayfun", "tayyar", "tayyip",  
"teslime", "tevfik", "tijen", "tonguc", "tufan", "tugay", "tugra",  
"turhan", "ufuk", "ugur", "umit", "utku", "yalcin", "yasar", "yase  
"yavuz", "yelda", "yeliz", "yesim", "yilmaz", "yunus", "yurdagul",  
"zahide", "zahit", "zarife", "zekai", "necati", "zeliha", "zerrin"  
  
int main(void)  
{  
    printf("%zu\n", asize(p));  
}
```

Ekrana isimleri yazdıralım

```
"yavuz", "yelda", "yeliz", "yesim", "yilmaz", "yunus", "yurdagul", "yurdakul",  
"zahide", "zahit", "zarife", "zekai", "necati", "zeliha", "zerrin", "ziya", "zu  
  
int main(void)  
{  
    for (size_t i = 0; i < asize(p); ++i)  
        puts(p[i]);  
}
```

Dizinin elemanlarını rasgele karıştıralım bu algoritmaya shoble algoritması deniyor basitçe.

Bu benim için önemli ulak ödevi bunla ilgili.

```
#define SIZE 100
```

```
const char* p[] = { ... }
```

```
int main(void)
{
```

```
    randomize();
```

```
    for (int i = 0; i < 1000; ++i) {
        size_t idx_1 = rand() % asize(p);
        size_t idx_2 = rand() % asize(p);
        if (idx_1 != idx_2) {
            const char* ptemp = p[idx_1];
            p[idx_1] = p[idx_2];
            p[idx_2] = ptemp;
        }
    }
```

```
    for (size_t i = 0; i < asize(p); ++i) {
        puts(p[i]);
    }
```

T

Burda rasgelelik var ama sıralama yok

```
"teslime", "tevfik", "tijen", "tonguc", "tufan", "tugay", "tugra", "t"
"turhan", "ufuk", "ugur", "umit", "utku", "yalcin", "yasar", "yasemin"
"yavuz", "yelda", "yeliz", "yesim", "yilmaz", "yunus", "yurdagul", "y"
"zahide", "zahit", "zarife", "zekai", "necati", "zeliha", "zerrin", "
```

```
int main(void)
{
    randomize();

    for (int i = 0; i < 1000; ++i) {
        size_t idx_1 = rand() % asize(p);
        size_t idx_2 = rand() % asize(p);
        if (idx_1 != idx_2) {
            const char* ptemp = p[idx_1];
            p[idx_1] = p[idx_2];
            p[idx_2] = ptemp;
        }
    }

    for (size_t i = 0; i < asize(p); ++i) {
        if (i && i % 10 == 0)
            printf("\n");

        printf("\t%s", p[i]);
    }
}
```

Rastgele yazdık daha sonra bunları "" arasında yazdırıp karıştırıp bir daha programa aldık.

Soru virgül kaldı sentax hatası olur mu.

```
"yunus", "billur", "tarkan", "recep", "fikret", "sevda", "melih", "necmettin", "hande", "teoman",
"gursel", "aykut", "selenay", "ufuk", "soner", "muzaffer", "temel", "teslime", "enes", "gul",
"naciye", "haldun", "yesim", "eda", "handesu", "binnur", "birhan", "yavuz", "figen", "nazli",
"cetin", "devrim", "haluk", "devlet", "deniz", "mustafa", "sadullah", "celal", "pelin", "mert",
"afacan", "poyraz", "menekse", "yalcin", "kezban", "garo", "tunc", "sefa", "fadime", "su",
"tonguc", "kaan", "cumhur", };
```

Yanlış oluşturamaz virgül kalabilir ilk değer verirken olsada olur olmasada boyutunu etkilemez.

Trailing comma denir.

Yazıyları yazdıralım bir daha

Soru for a ne yazarsam bu isimlerin ilk harfleri yan yana yazar.

Printf(%c , *p[i]) bence

```
int main(void)
{
    for (size_t i = 0; i < asize(p); ++i) {
        printf("%c ", *p[i]);
    }
}
```

Aynısını dedi.

////////////////////////////////////

Buda olurmuydu.

```
int main(void)
{
    for (size_t i = 0; i < asize(p); ++i) {
        printf("%c ", p[i][0]);
    }
}
```

Evet olurdu.

((P+i)+0)

////////////////////////////////////

İsimlerin son harflerini nasıl yazdırırız.

Bence *p[strlen(p[i])] sonuçta içindeki i yazının uzunluğu -1 olmalı

```

int main(void)
{
    for (size_t i = 0; i < asize(p); ++i) {
        printf("%c ", p[i][strlen(p[i]) - 1]);
    }
}

```

Hocada aynısını yaptı

Ancak boş karakterolsaydı tanımsız olacaktı.

////////////////////////////////////

Soru içinde e olanları yazdır. //

For

Char c = 'e'

if(Strchr(p[i],c))

Printf p[i]

```

int main(void)
{
    printf("icinde hangi karakter olanlar: ");
    int c = getchar();

    for (size_t i = 0; i < asize(p); ++i) {
        if (strchr(p[i], c)) {
            printf("%s ", p[i]);
        }
    }
}

```

Benim yazdığım gibi yaptı.

////////////////////////////////////

Alfabadeki bütün karakterleri karşılaştırmasını yapalım kaç tane olduğunda yazalım


```

int main(void)
{
    for (int c = 'a'; c <= 'z'; ++c) {
        printf("icinde (%c) harfi olanlar\n", c);
        int cnt = 0;

        for (size_t i = 0; i < asize(p); ++i) {
            if (strchr(p[i], c)) {
                printf("%s ", p[i]);
                ++cnt;
            }
        }
        printf("\n%d tane bulundu", cnt);
        pdline();
    }
}

```

Cok güzel bir örnek iç içe for döngüsü ile bütün harflere bakabildik.

////////////////////////////////////

Bu kodda tanımsız davranış var mıdır

Bence var strrev ile yazıya mudale ediypruz

```

#define LINE 100
size_t
const char* p[] = { ... }

int main(void)
{
    for (size_t i = 0; i < asize(p); ++i) {
        puts(_strrev(p[i]));
    }
}

```

Tanımsız davranış kesinlikle var string literali ters çevrilemez c++ da sentax hatası.

Char *p dönüşümü uyarısı

////////////////////////////////////

Soru içinde girilen yazı , olanları yazdıralım.

```
const char* p[] = { ... }
```

```
int main(void)
{
    char entry[40];

    printf("bir yazi girin: ");
    scanf("%s", entry);

    for (size_t i = 0; i < asize(p); ++i) {
        if (strstr(p[i], entry)) {
            printf("%s ", p[i]);
        }
    }
}
```

Microsoft Visual Studio Debug Console

```
bir yazi girin: kan
okan furkan hakan atakan tarkan
D:\KURSLAR\NISAN_2022_C\Release\NISAN_2022_C.exe (process 24148) exited with code 0.
Press any key to close this window . . .
```

Diziyi ilk değerle başlattığımız için istediğimiz kadar ekleme yapabiliriz. Boyut önemsiz asize makrosu kullanık.

////////////////////////////////////

Soru içinde belli karakterler aei gibi var mı.

Strpbrk ile yapabilirim.

```
int main(void)
{
    char entry[40];

    printf("bir yazi girin: ");
    scanf("%s", entry);

    for (size_t i = 0; i < asize(p); ++i) {
        if (strpbrk(p[i], entry)) {
            printf("%s ", p[i]);
        }
    }
}
```

Üsteki koda sadece strpbrk ekledik aslında algoritma bu kadar basit ve net

```

bir yazi girin: poug
gunay smeyye cansu durmus rumeysa nasrullah tuncer polat umit turhan perihan pinat tayfun papaty
nter dost engin yurdagul nagehan ferhunde yurdanur tanju pakize osman abdulmuttalip gulsen fugen
ulki muruvvet furkan busra hulusi petek ercument durriye sadegul yusuf pelinsu julide gurbuz nuci
phi muhsin tayyip nuri zubeyde ceyhun onat cengiz bulent burhan tansu huseyin bilge orkun burak
an tugay cuneyt suleyman korhan utku olcay hulya ege bora tufan sarp eylul yurdakul muslum gizem
n dogan kurthan mukerrem gazi nurdan tugra bennur gulden yunus billur recep teoman gursel aykut
desu binnur yavuz figen haluk mustafa sadullah pelin poyraz garo tunc su tonguc cumhur
D:\KURSLAR\NISAN_2022_C\Release\NISAN_2022_C.exe (process 1416) exited with code 0.
Press any key to close this window . . .

```

Çıktısı bu

////////////////////////////////////

Elemanları tekrar alfabetik sıra ile yazınız

Kodda boş kalan yere yazıcaz.

```

int main(void)
{
    char entry[40];

    for (size_t i = 0; i < asize(p); ++i) {
        printf("%s ", p[i]);
    }
    pdline();

    //kod

    for (size_t i = 0; i < asize(p); ++i) {
        printf("%s ", p[i]);
    }
    pdline();
}

```

Bu bir sıralama algoritması

Dizinin elemanlarının adresini değiştirecez dizideki string literallerini değiştirmicez.

Herhangi bir sıralama algoritmasıyla yazabilirim

Buubla sort

Ana algoritma bu ama benim istediğim işi yapmaz neden

```

for (size_t i = 0; i < asize(p); ++i) {
    printf("%s ", p[i]);
}
pdline();

for (size_t i = 0; i < asize(p) - 1; ++i) {
    for (size_t k = 0; k < asize(p) - 1 - i; ++k) {
        if (p[k] > p[k + 1]) {
            const char* ptemp = p[k];
            p[k] = p[k + 1];
            p[k + 1] = ptemp;
        }
    }
}

for (size_t i = 0; i < asize(p); ++i) {
    printf("%s ", p[i]);
}
pdline();

```

Çünkü karşılaştırma bu şekilde değil strcmp ile yapılmalı.

Burada adresleri karşılaştırıyoruz.

```

char entry[40];

for (size_t i = 0; i < asize(p); ++i) {
    printf("%s ", p[i]);
}
pdline();

for (size_t i = 0; i < asize(p) - 1; ++i) {
    for (size_t k = 0; k < asize(p) - 1 - i; ++k) {
        if (strcmp(p[k], p[k + 1]) > 0) {
            const char* ptemp = p[k];
            p[k] = p[k + 1];
            p[k + 1] = ptemp;
        }
    }
}

```

Bu şekilde olurdu.

```

rkan recep fikret sevdâ melih necmettin hanâde teoman gursel aykut selenay ufuk soner muzaffer temel teslime enes gul naciye haldun
yesim eda handesu binnur birhan yavuz figen nazli cetin devrim haluk devlet deniz mustafa sadullah celal pelin mert afacan poyraz
menekşe yalcin kezban garo tunc sefa fadime su tonguc kaan cumhur ilayda aysel
-----
abdi abdullah abdulmuttalip adem adnan afacan agah ahmet akin alev ali alican alparslan anil arda asim askin aslican aslihan ata a
takan atalay atif atil aycan aydan aykut ayla aylin aynur ayse aysel aytac aziz azize azmi baran bekir belgin bennur beril berivan
berk beste beyhan bilal bilge bilgin billur binnaz binnur birhan bora bulent burak burhan busra cahide cahit can canan candan can
er cansu cebraîl celal celik cem cemal cemil cemile cemre cengiz cesim cetin ceyda ceyhan ceyhun ceylan cezmi cihan cihat ciler cu
mhur cuneyt demet demir deniz derin derya devlet devrim diana dilber dilek dogan dost durmus durriye ece eda edip ediz efe efecan
ege egemen emine emihan emre emrehan eren erenhan erenhan erenhan erenhan erenhan erenhan erenhan erenhan erenhan erenhan erenhan

```

////////////////////////////////////

Bu kodda nereyi değiştirsem büyükten küçüğe sıralama yapar.

```

for (size_t i = 0; i < asize(p); ++i) {
    printf("%s ", p[i]);
}
pdline();

for (size_t i = 0; i < asize(p) - 1; ++i) {
    for (size_t k = 0; k < asize(p) - 1 - i; ++k) {
        if (strcmp(p[k], p[k + 1]) < 0) {
            const char* ptemp = p[k];
            p[k] = p[k + 1];
            p[k + 1] = ptemp;
        }
    }
}

```

Strcmp < 0 yapmalıydık bu sefer küçükse büyüğü öne alıcak

////////////////////////////////////

Mülakat sorusu herkes yanlış cevap veriyor.

```
su  ata ece eda gul naz tan kaya mert tunc utku deniz ferda galip metin zekai
```

Kodu öyle bir yazın ki hem alfabetik alsun hemde boyutu küçükler başta olsun.

Burada sıralama algoritması değişmiyor sadece kriter değişiyor yani if içi değişicek

Aslında buldum ama tek iste birleştiremedim

Kendim yazdım

```
su  ata ece eda gul naz tan kaya mert tunc utku deniz ferda galip
```

```
p[k] adresindeki yazı p[k + 1] adresindeki yazıdan daha uzunsa
```

```
uzunlukları aynı fakat6
```

```
p[k] adresindeki yazı p[k + 1] adresindeki yazıdan büyükse
```

```
p[k]          p[k + 1]
```

```
tan           gul
```

```

it main()

const char* p[] = { "murat", "berat", "semre", "gamze", "gaya", "emin", "mustafa", "ala", "eda", "ece",
    "abdullah", "selami", "kursat" };

for (size_t i = 0; i < asize(p)-1; ++i) {
    for (int k = 0; k < asize(p)-1-i; ++k) {
        if ( strlen(p[k]) > strlen(p[k + 1]) || ( strlen(p[k]) == strlen(p[k + 1]) && strcmp(p[k], p[k + 1]) > 0)) {
            const char* ptemp = p[k];
            p[k] = p[k + 1];
            p[k + 1] = ptemp;
        }
    }
}

```

Buda hocanın yazdığı.

```

for (size_t i = 0; i < asize(p); ++i) {
    printf("%s ", p[i]);
}
pdline();

for (size_t i = 0; i < asize(p) - 1; ++i) {
    for (size_t k = 0; k < asize(p) - 1 - i; ++k) {
        size_t len1 = strlen(p[k]);
        size_t len2 = strlen(p[k + 1]);
        if (len1 > len2 || (len1 == len2 && strcmp(p[k], p[k + 1]) > 0)) {
            const char* ptemp = p[k];
            p[k] = p[k + 1];
            p[k + 1] = ptemp;
        }
    }
}

```

```

-----
su ali ata can cem ece eda efe ege gul naz abdi adem agah akin alev ani
ip ediz emre enes esen esin esra fuat garo gazi jade kaan kaya lale mer
uk ugur umit utku ziya adnan ahmet askin aycan aydan aykut aylin aynur
busra cahit canan caner cansu celal celik cemal cemil cemre cesim ceti

```

Buda hocanın yazdığı benle aynı tek fark uzun olmasın diye strlen i değışkende yazdık.

////////////////////////////////////

Bütün örnekler yukarda açıklandı

////////////////////////////////////

Örnek

```
const char* const pmons[] = {  
    "",  
    "Ocak",  
    "Subat",  
    "Mart",  
    "Nisan",  
    "Mayis",  
    "Haziran",  
    "Temmuz",  
    "Agustos",  
    "Eylul",  
    "Ekim",  
    "Kasim",  
    "Aralik"  
};
```

Boş ilk değer vermek index eşitlemek için.

////////////////////////////////////

Örnek dizinin sonu null pointer.

```
"gursel", "aykut", "selenay", "ufuk", "soner", "muzaffer", "temel", "teslime", "enes", "gul",  
"naciye", "haldun", "yesim", "eda", "handesu", "binnur", "birhan", "yavuz", "figen", "nazli",  
"cetin", "devrim", "haluk", "devlet", "deniz", "mustafa", "sadullah", "celal", "pelin", "mert",  
"afacan", "poyraz", "menekse", "yalcin", "kezban", "garo", "tunc", "sefa", "fadime", "su",  
"tonguc", "kaan", "cumhur", "ilayda", "aysel", "tan", NULL};
```

```
int main(void)  
{  
    int i = 0;  
    while (p[i])  
        printf("%s ", p[i++]);  
}
```

Diziyi bu şekilde null a kadar yazdırabilirim.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

Bu kodda programcının başını belaya sokacak hata nedir uyarı ve hata yok.

```
const char* p[100] = {
    "yeliz", "tansel", "gunay", "sumeyye", "nihal", "kerem", "ayla", "berk", "sezer", "samet",
    "cihat", "saniye", "naz", "ata", "cansu", "yasemin", "durmus", "refik", "rumeysa", "kelami",
    "atil", "taci", "akin", "berivan", "semsit", "hilmi", "sefer", "nasrullah", "tuncer", "polat",
    "kaya", "yelda", "asim", "saadet", "esra", "can", "umit", "turhan", "fazilet", "kayahan",
    "perihan", "pinat", "tayfun", "papatya", "turgut", "keriman", "emine", "niyazi", "sadiye", "lale",
    "rupen", "aynur", "kerim", "esin", "edip", "sidre", "melisa", "fuat", "kunter", "lamia",
    "hilal", "melek", "ceylan", "nalan", "jade", "kazim", "dost", "alican", "engin", "canan",
    "yurdagul", "nagehan", "zeliha", "tekin", "caner", "nazife", "ferhunde", "yurdanur", "zahide", "tanju",
    "celik", "pakize", "sadi", "izzet", "aslican", "osman", "abdi", "nisan", "abdulmuttalip", "gulsen",
    "afacan", "poyraz", "menekse", "yalcin", "kezman", "garo", "tunc", "sefa", "fadime", "su",
};

int main(void)
{
}
```

Kodda kazım le dost arasında virgül yok görmesi zor diye çok önemli

Kazımdost yazar birleşir

Boyutu 100 olduğu için son eleman null pointer oldu.

Bunu direference yaparsak tanımsız davranış olur.

O yüzden her zaman asize(p) ile boyutunu kontrol etmemiz gerek.

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

Eski örnek knut 3*size işlem yapan kodu 2. Size yapmak.

```
int main(void)
{
    int a[SIZE];

    ///CODE
    int sval = 35;

    int i;

    for (i = 0; i < SIZE; ++i) {
        if (a[i] == sval)
            break;
    }

    //
}
```

Çözümü

```
#define SIZE 100

int main(void)
{
    int a[SIZE + 1];

    ///CODE
    int sval = 35;

    int i;

    a[SIZE] = sval;

    for (i = 0; a[i] != sval; ++i)
        ;

    if (i == SIZE)
        I
}
```

Diziye aranacak değeri de yazdık ve sonra eşit değilse çıktı. Mantıklı.

////////////////////////////////////

POINTER TO POINTER

```
int main(void)
{
    int x = 20;
    int* ptr = &x;

    printf("&x = %p\n", &x);
    printf("ptr = %p\n", ptr);
    printf("&ptr = %p\n", &ptr);
}
```

Microsoft Visual Studio Debug Console

```
&x = 003EFA08
ptr = 003EFA08
&ptr = 003EFA04
```

D:\KURSLAR\NISAN_2022_C\Release\NISAN_2022_C.exe (process)
Press any key to close this window . . .

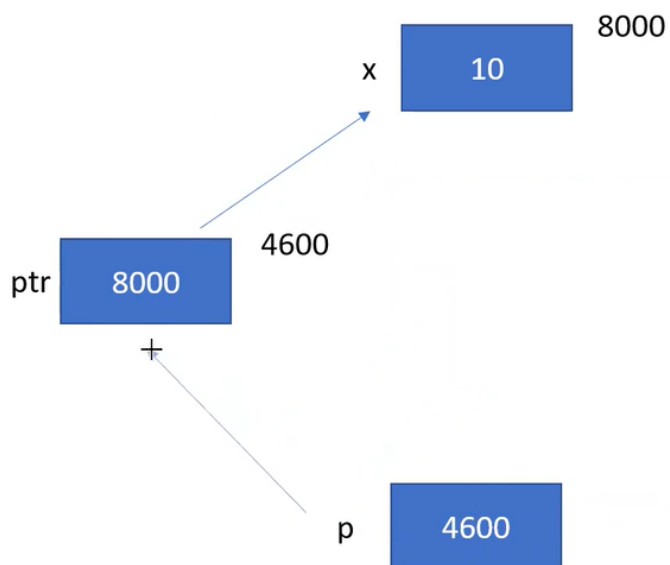
SORU 1

```

int main(void)
{
    int x = 10;
    int y = 45;
    int* ptr = &x; //ptr is a pointer to int
    int** p = &ptr; //p is a pointer to pointer to int

    /*ptr ==> x
    /*p = &y; // ptr = &y

```



SORU

```

int main(void)
{
    int x = 10;
    int y = 45;
    int* ptr = &x; //ptr is a pointer to int
    int** p = &ptr; //p is a pointer to pointer to int

    **p = 999;

    printf("x = %d\n", x);
}

```

X = 999

```

**p = 999;
++** p;

```

X=1000 OLDU.

MÜLAKAT SORUSU BU KOD NE YAPAR !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

int main(void)
{
    int a[] = { 1 ,2, 3 };
    int b[] = { 10 ,20, 30 };
    int c[] = { 100 ,200, 300};

    int* p = a;
    int** ptr = &p;

    ++*ptr;
    ++**ptr;

}

```

Adizisinin 2. Elemanı 3 olacak

1 , 3, 3

////////////////////////////////////

Örnek pswap fonksiyonu

```
void pswap(int** ptr1, int** ptr2)
{
    int* ptemp = *ptr1;
    *ptr1 = *ptr2;
    *ptr2 = ptemp;
}

int main(void)
{
    int x = 56;
    int y = 98;
    int* p1 = &x;
    int* p2 = &y;

    printf("%d %d\n", *p1, *p2);

    pswap(&p1, &p2);

    printf("%d %d\n", *p1, *p2);
}
```

Ptr1 p1 pointer değişkeninin adresi

*ptr1 p1 pointerin değeri

Bu şekilde p1 pointerinin değerini yani gösterdiği nesne adresini değiştiriyorum

Burada x ve y değişkenlerinin değerleri değişmiyor.

String literalinde yaptığımız örnek ile aynı aslında

String değişmiyor onu gösteren pointer dizisindeki pointerler değişiyordu.

Tehlikeli hata

```
void func(int* p, int n)
{
    p = (int*)malloc(n * sizeof(int));
}

int main(void)
{
    int* ptr;

    func(ptr, 100);
}
```

Buradaki hata değeri ptr ye atamıyoruz ptr hala çöp değerde

Call by value aslında bu olması gereken

```
void func(int** p, int n)
{
    *p = (int*)malloc(n * sizeof(int));
}

int main(void)
{
    int* ptr;

    func(&ptr, 100);
}
```

int**p yapmalıydık

içeride *p olmalıydı.

